OPTIMIZING SCHEDULING AND TRAJECTORY PLANNING OF CONNECTED AUTOMATED VEHICLES FOR FREEWAY ON-RAMP MERGING

JIEMING CHEN

PhD

The Hong Kong Polytechnic University

2025

The Hong Kong Polytechnic University

Department of Electrical and Electronic Engineering

Optimizing Scheduling and Trajectory Planning of Connected Automated Vehicles for Freeway On-Ramp Merging

Jieming CHEN

A thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy

August 2024

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

_____ (Signed)

Jieming CHEN (Name of student)

Abstract

Intensive interactions among vehicles at freeway on-ramp merging areas frequently cause congestion and accidents. The collaboration of connected automated vehicles (CAVs) is promising to effectively coordinate these conflicts. However, CAV-based control encounters significant challenges in real-time optimization of vehicle scheduling and trajectory planning, especially in scenarios involving multiple lanes and a large number of vehicles. To tackle these challenges, this dissertation mathematically models the freeway merging problem and develops three algorithms to solve the problem.

The first work proposes a mixed integer nonlinear programming (MINLP) model for the cooperative merging of two traffic streams at a single-mainline freeway on-ramp merging section. The proposed model simultaneously optimizes multiple vehicles' trajectories and their merging sequence to improve traffic efficiency and ensure safety. Unlike conventional treatments, which match one mainline facilitating vehicle with one merging vehicle, the proposed model determines the optimal number of facilitating vehicles and which mainline vehicles should serve as the facilitating vehicles to cooperatively minimize disruption from ramps. The safety and feasibility of the planned vehicle trajectories are guaranteed at any time. To solve the model rapidly, we propose a solution algorithm that incorporates an iterative linear programming method into a novel search process based on a necessary condition for optimality that we identify and prove. The algorithm is highly efficient because it enjoys a significantly reduced search space. The proposed approach, consisting of the MINLP model and the solution algorithm, is evaluated under different traffic demands and mainline-ramp demand ratios and real vehicle arrival patterns from the NGSIM dataset. The performance of the proposed method outperforms benchmark CAV control algorithms, and the computational efficiency is promising for real-time automated merging tasks.

The second work considers a multi-lane freeway on-ramp merging section, focusing on the simultaneous decision-making of lane changes, vehicle sequences, and trajectories. To this end, an integrated MINLP optimization model is proposed to jointly optimize lane change decisions, vehicle sequences, and vehicle trajectories, with the objective of maximizing traffic efficiency and driving comfort. However, such a complicated model cannot be directly solved by existing optimization software. To rapidly obtain solutions, this study develops a Generalized Benders decomposition (GBD)-based solution algorithm to tackle the challenges of multi-vehicle combinatorial optimization and nonlinear trajectory optimization problems. Meanwhile, the property of finite convergence is proved. Numerical experimental results turn out that the traffic performance of the proposed model outperforms benchmark CAV control methods under different traffic demands and mainline-ramp demand ratios, demonstrating significant traffic benefits from jointly regulating lane changes, driving sequences, and utilizing microscopic vehicle information. Also, this study analyses traffic delays and the number of lane changes by the proposed model under varying road lengths, i.e., the lengths of lane-changing and merging areas.

The third work introduces a bi-level approach that nests optimization modelling within deep reinforcement learning to jointly optimize vehicle sequences, lane selection, and trajectories, aiming to provide a rapid, safe, and high-quality solution for the problem of multi-lane freeway merging. In the upper level, we develop an attention-based sequential policy network to sequentially construct driving sequences and lane selections for multiple vehicles. Specifically, we employ an attention mechanism to

learn dynamic inter-dependencies with other vehicles, thus facilitating more informed and adaptive decision-making. In the lower level, we utilize a nonlinear model predictive controller to generate safe trajectories and use total travel delay to guide upper-level learning for optimizing long-term traffic efficiency. Additionally, we introduce a leader-and-lane specific credit assignment mechanism to address global credit assignment for the multi-vehicle merging problem. Computational results demonstrate that our method outperforms rule-based and searching-based methods in terms of solution quality, and the computation efficiency of the proposed approach is promising for real-time automated merging tasks.

Keywords: Connected Automated Vehicles; On-Ramp Merging; Mixed-Integer Nonlinear Programming; Sequencing Planning; Trajectory Planning.

Publications

The thesis is composed of the following three papers:

[A] **Chen, J**., Zhang, Y., Zhou, Y., Guillaume, S., Chung, E., and Wu, Y. (2024) Learning Lane Selection and Driving Orders for Connected Automated Vehicles at Multi-lane Freeway Merging Sections. *IEEE Transactions on Intelligent Transportation Systems (under review).*

[B] **Chen, J**., Wu, Y., Zhou, Y., Chung, E., and Wang, S. (2024) Solving Connected Automated Vehicle Merging Problems: A Generalized Benders Decomposition-based Approach for Mixed-Integer Nonlinear Programming. *Transportation Research Part B (under review).*

[C] **Chen, J**., Zhou, Y., and Chung, E. (2023) An Integrated Approach to Optimal Merging Sequence Generation and Trajectory Planning of Connected Automated Vehicles for Freeway On-Ramp Merging Sections. *IEEE Transactions on Intelligent Transportation Systems*.

Other papers during my Ph.D. study, not included in this thesis, are:

[D] Fan, T., **Chen, J.**, and Chung, E. (2025) CAV lane change protocol with CTH safety guarantee for cooperative driving on dedicated highways. *Communications in Transportation Research (under review)*.

[E] Zhang, Y., **Chen, J.**, Zhou, T., and Guillaume, S. (2025) COIN: collaborative interaction-aware multi-agent reinforcement learning for self-driving systems. *International Conference on Robotics and Automation (ready to submit).*

[F] Fan, X., **Chen, J.** , Wang, Q., and Chung, E. (2024) CAV lane change protocol with CTH safety guarantee for cooperative driving on dedicated highways. *IEEE Transactions on Mobile Computing (under review)*.

[G] Zhou, Y., **Chen, J**., Chung, E., and Ozbay, K. (2023) CAV-Enabled Active Resolving of Temporary Mainline Congestion Caused by Gap Creation for On-Ramp Merging Vehicles. *IEEE Transactions on Intelligent Transportation Systems*.

[H] **Chen, J**., Chen, X., and Liu, S. (2023) Trajectory Planning of Autonomous Mobile Robot using Model Predictive Control in Human-Robot Shared Workspace. *In 2023 IEEE 3rd International Conference on Electronic Technology, Communication and Information (ICETCI)*

[I] **Chen, J.**, Zhou, Y., Chung, E., and Ozbay, K. (2022) CAV-Based Active Congestion Resolving for Improving Mainline Traffic Flow Efficiency of A Freeway On-Ramp Merging Section. *In 2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*

Acknowledgments

I would like to express my deepest gratitude to my supervisor, Prof. Edward CHUNG, for providing me with the opportunity to pursue my Ph.D. studies and for his invaluable guidance over the past three years. During my time at HK PolyU, I am particularly thankful to Prof. Chung for supporting me in exploring research areas of my interest. Even more valuable was his emphasis on the importance of clear communication and his regular meetings, which greatly improved my skills in communication, presentations, and writing. I also deeply appreciate Prof. Chung for organizing hiking trips and supporting my exchange in Singapore, allowing me to experience diverse landscapes, meet new people, and enrich my life experiences.

I would like to extend my sincere thanks to Dr. Yue ZHOU. Under the guidance of Prof. Chung and Dr. Zhou, I completed the submission, revision, and acceptance of my first paper. I sincerely appreciate his willingness to help me, which greatly strengthened my foundation in academic writing.

I am also thankful to Prof. Guillaume SARTORETTI, my supervisor at the National University of Singapore, for giving me the opportunity to join his lab. This experience exposed me to a wider range of research areas and expanded my problem-solving methodologies. I also appreciate him for taking the time to discuss my work and provide valuable suggestions. Additionally, I am very grateful to the friends I met at NUS, including Yifeng ZHANG, Yibing YANG, Jianghong DONG, Jingqiang LIU, Derek TAN, Yizhuo WANG, Peizhuo LI, Weiheng DAI, and everyone in MARMOTLAB.

My thanks also go to the research team and colleagues at PolyU, including Dr. Weihua GU, Dr. Hongbo YE, Dr. Xiaowen BI, Dr. Zhuang XIAO, Dr. Xiao YANG, Dr. Jiaxin WEN, Xiexin ZOU, Meng LONG, Rong ZHAO, Chaoyun WANG, Ruoheng WANG, Chuang XU, Yun LI, Zhixian TANG, Tingting FAN, Li ZHEN, and everyone else in EF113. I am truly grateful to have met you all and to have become friends.

Last but not least, I extend my heartfelt thanks to my parents for their love and support. I am also deeply grateful to my wife, who has always believed in me. Wishing everyone all the best in their endeavours.

Contents

List of Figures

List of Tables

List of Abbreviations

AV	Autonomous Vehicle
ACC	Adaptive Cruise Control
CACC	Cooperative Adaptive Cruise Control
CAV	Connected Automated Vehicle
CF	Car Following
DNN	Deep Neural Networks
DRL	Deep Reinforcement Learning
FIFO	First In First Out
FSM	Finite State Machine
GBD	Generalized Benders Decomposition
GCN	Graph Convolutional Network
IP	Integer Programming
ITS	Intelligent Transportation System
LLM	Large Language Model
LQR	Linear Quadratic Regulator
MARL	Multi-Agent Reinforcement Learning
MCTS	Monte Carlo Tree Search
MDP	Markov Decision Process
MILP	Mixed Integer Linear Programming
MINLP	Mixed Integer Nonlinear Programming
MPC	Model Predictive Control
NLP	Nonlinear Programming
OCP	Optimal Control Problem
PMP	Pontryagin's Minimum Principle
RL	Reinforcement Learning
RSU	Roadside Unit
SA	Simulated Annealing
SCP	Sequential Convex Programming
SQP	Sequential Quadratic Programming
SUMO	Simulation of Urban Mobility
VORLA	Vehicle Ordering and Lane Selection
V2X	Vehicle-to-Everything

Chapter 1

Introduction

1.1 Background

Freeway on-ramp merging areas are recognized as typical bottlenecks where multiple traffic streams compete for limited roadway capacity. Frequent lane-changing manoeuvres of mainline vehicles and merging manoeuvres of on-ramp vehicles disrupt the smooth mainline traffic flow, resulting in traffic congestion, increased fuel consumption, and traffic accidents (Zhang et al., 2011; He et al., 2017; Xiao et al., 2017). Conventional traffic control and management solutions for merging sections typically employ flow-based strategies, such as ramp metering (Papageorgiou and Kotsialos, 2002), variable speed limit (Chen et al., 2014), and the combination of variable speed limits and ramp metering (Carlson et al., 2010). These flow-based methods take a macroscopic perspective and modulate traffic at an aggregate level. However, these methods cannot directly coordinate individual vehicle behaviours, and uncontrollable microscopic vehicle behaviours may still lead to negative macroscopic phenomena, such as traffic paralysis (Yuan et al., 2017; Han and Ahn, 2018).

Significant advances in computation, perception, and control technologies over the last decade have increasingly integrated autonomous vehicles (AVs) into people's lives. AVs make driving decisions based on data from on-board sensors, enabling microscopic control of vehicles' trajectories. Alongside electrification and shared mobility, AVs are considered as one of the three major revolutions in intelligent transportation system (ITS), expected to enhance people's mobility by reducing travel time, lowering fuel consumption, decreasing parking demand, and improving accessibility. (Tengilimoglu et al., 2023). Moreover, AVs are expected to improve road safety by preventing accidents caused by speeding, distraction, and fatigue. However, there are concerns about potential negative impacts, such as increased congestion from higher vehicle miles travelled and more empty trips, as discussed in several studies (Tengilimoglu and Wadud, 2022).

Based on AV and emerging vehicle-to-everything (V2X) communication technologies, connected automated vehicles (CAVs) are enabled to communicate with each other and share information with traffic infrastructures. The direct and fully cooperative control properties of CAVs have introduced a new perspective in microscopic traffic control (van Arem et al., 2006; Talebpour et al., 2016; Wang et al., 2016; Rios-Torres et al., 2017; Rios-Torres et al., 2018; Sun et al., 2018). Its direct control capabilities, such as adaptive car-following control, are highlighted by various studies (e.g., Ni et al., 2016; Zhou et al., 2019b; Li et al., 2021b; Shi et al., 2023a). Additionally, its cooperative driving capabilities enhance the synchronization of vehicle movements, which is particularly beneficial for traffic efficiency, notably in platoon control (e.g., Chen et al., 2018; Li et al., 2018; Zhang et al., 2022; Shen et al., 2022), ramp merging (e.g., Jing et al., 2019; Karimi et al., 2020; Chen et al., 2021), and driving at unsignalized intersections (e.g., Li and Zhou, 2017; Wang et al., 2021).

Merging on-ramp CAV traffic into multilane freeway CAV traffic is an integrated problem that involves decisions on lane changes, vehicle driving sequences, and trajectory planning to optimize traffic performance and driving comfort. Specifically, the lane-changing decision is to determine lane-changing vehicles among mainline vehicles to accommodate on-ramp vehicles or avoid downstream disturbances. Then, the vehicle sequencing decision is to determine the passing order of vehicles targeting the same lane. This is achieved by selecting the target followers of lane-changing and merging vehicles, which are also called facilitating vehicles in this study. Lastly, to implement a scheduling plan that includes lane changes and vehicle sequences, the multi-vehicle trajectory planning is involved to generate collision-free and physically feasible trajectories. Following the planned trajectories, designated facilitating vehicles create adequate gaps, while lane-changing and merging vehicles coordinate their movements to align with these gaps. Additionally, the remaining vehicles adjust their speeds to maintain safe distances from the vehicles ahead. For example, as shown in Figure 1, the mainline vehicles are designated as lane-changing, facilitating, and lane-keeping vehicles. Then, the corresponding trajectories are generated to implement designated lane choices and vehicle sequences.

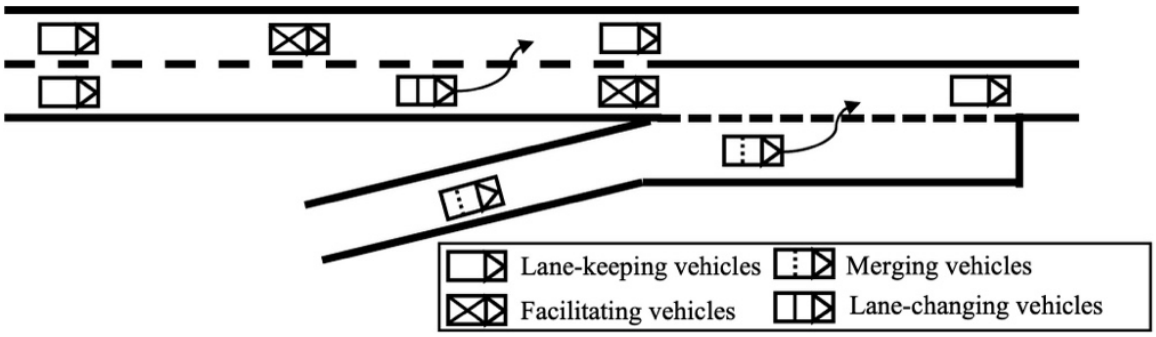


Figure 1.1. Four vehicle types in on-ramp merging scenarios.

1.2 Motivations and Objectives

(1) Advantages by CAV-based Control

The development of CAV-based control has emerged as a promising solution to address the escalating traffic congestion. It offers two key advantages for intelligent transportation control:

1) Precise Control: This approach enables precise control over each vehicle's behaviour, leading to a safer and more efficient ITS. Unlike traditional traffic management methods that focus on controlling traffic flow at a macroscopic level, CAV-based control operates at the individual vehicle level, allowing for more granular adjustments.

2) Optimality: In fully CAV environments, a primary focus is on achieving optimal scheduling decisions to maximize traffic performance. Since the number of vehicles at a moment is finite, it is feasible to employ advanced mathematical techniques, such as integer programming, as the foundation for finding the best strategies. By optimizing key factors such as lane assignment, right-of-way decisions, and vehicle speeds, the system can significantly enhance overall traffic efficiency and safety.

(2) Challenges by the CAV-based Control

However, coordinating multiple CAV presents some challenges due to the following conflicting factors:

1) Stringent Real-Time Requirements: The continuous movement of vehicles imposes extremely stringent real-time requirements, making it difficult to achieve timely and effective control.

2) Exponential Complexity: Coordinating multiple CAVs involves solving an integer programming (IP) problem for scheduling. As the number of CAVs in the system increases, the complexity of coordinating them grows exponentially. This complexity not only demands significant computational resources but also poses challenges in maintaining scalability and responsiveness in large-scale deployments.

3) MINLP Problem: Trajectory planning introduces a nonlinear programming (NLP) problem, along with the scheduling IP problem, resulting in a mixed-integer nonlinear programming (MINLP) problem for CAV-based control, which is NP-hard. This inherent difficulty requires sophisticated optimization techniques and often necessitates trade-offs between optimality and computational feasibility. Moreover, the real-world implementation of these solutions must account for dynamic and unpredictable traffic conditions, further complicating the problem.

(3) Objectives

Based on these advantages and challenges, the objectives of this study are as follows:

- Mathematically model the CAV-based merging problem including scheduling-related and trajectory-related decisions, to enable a comprehensive description and analysis.
- Design efficient solution algorithms based on the characteristics of the problem to rapidly solve this MINLP problem.

1.3 Organization of thesis

Based on the above research backgrounds and objectives, the rest of the thesis is structured as follows:

- Chapter 2 presents a comprehensive literature review of three parts: trajectory planning, vehicle sequence planning, and lane change planning.
- Chapter 3 models the cooperative merging of an on-ramp CAV stream and a mainline CAV stream at a single-lane freeway on-ramp merging section as an MINLP model. It considers passing orders and detailed trajectories of two streams of CAVs, aiming to minimize total travel time. To solve the proposed model, we propose an integrated solution algorithm that includes a sequential search process for rapidly determining the optimal merging sequence and an iterative linear programming method for generating nonlinear trajectories. Numerical experiments are conducted to demonstrate the good performance of the proposed approach, highlighting the benefits of optimizing both merging sequences and trajectories.
- Chapter 4 formulates the problem of merging on-ramp CAV flows into multiple mainline CAV flows as an MINLP model. It considers spatiotemporal relationships between vehicles, vehicle kinematics, and road geometry, with the aim of maximizing traffic efficiency, avoiding unnecessary lane changes, and generating the smoothest vehicle trajectories. To efficiently solve the proposed MINLP model, we develop a decomposition-based solution algorithm. The benefits from considering vehicle sequencing, multilane utilization, and microscopic vehicle information, are highlighted through comparative analyses with three baseline models. Additionally, this chapter

discusses the most time-efficient lengths for cooperation areas.

- Chapter 5 also addresses the multi-lane freeway merging problem and proposes a hybrid bi-level control approach that integrates deep reinforcement learning with optimization modelling. This approach aims to combine the fast inference capabilities of learning-based methods with the explainable safety guarantees of optimization-based methods. The vehicle sequences, lane selections, and trajectories are optimized to minimize total travel delay. Specifically, the upper level serves as the scheduling planner, where we design an attention-based neural network to make decisions on target lanes and right-of-way. The lower level employs a nonlinear model predictive controller to continuously update trajectories, ensuring vehicles reach the designated lanes and follow the planned sequence. Experiments demonstrate that our method quickly achieves superior solution quality compared to other meta-heuristic and rule-based methods. It effectively balances computational time with solution quality while also offering promising scalability across scenarios with varying numbers of vehicles.
- Chapter 6 concludes the thesis by summarizing the ideas of algorithm design, specifying contributions, and discussing future works.

Chapter 2

Literature review

CAV-based merging control involves three key tasks: trajectory planning, vehicle sequence planning, and lane change planning. Trajectory planning generates collision-free paths through merging sections. Vehicle sequence planning optimizes the passing orders (i.e., right-of-way) of CAVs, and lane change planning assigns target mainline lanes for CAVs. This chapter first reviews and categorizes existing methods for these three tasks, and then identifies current challenges and limitations in CAV-based control.

2.1 Trajectory Planning

Trajectory planning is essential for safe cooperative merging process, adjusting the speeds of some mainline and on-ramp vehicles to create adequate spacing. In these studies, the mainline facilitating vehicles are predetermined. This can be categorized into two main strategies: reactive control, and look-ahead control.

(1) Reactive Control

Reactive control generates actions based on the error between the actual and reference states. Commonly used linear feedback controllers, such as adaptive cruise control (ACC) and cooperative adaptive cruise control (CACC), aim to maintain a predefined constant time headway or spacing. Xu et al. (2003) controlled a mainline facilitating vehicle with adaptive cruise control (ACC) and cooperative adaptive cruise control (CACC) to form enough space for a merging vehicle and assessed performance of both control methods. Milanes et al. (2011) designed a fuzzy controller incorporating human procedural knowledge to track a linear reference distance function for a merging vehicle and a trailing mainline vehicle. Chen et al. (2021) enhanced an ACC controller with a feedforward component to ensure smooth merging behaviours and string stability. Due to the simplicity and effectiveness of ACC and CACC, numerous studies explored its broader impacts on traffic dynamics. For instance, Xiao et al. (2018) studied how deactivating CACC affects congestion and flow heterogeneity at traffic bottlenecks. Li et al. (2021) investigated a high-risk ACC pattern characterized by strong braking followed by rapid acceleration.

(2) Look-ahead Control

Look-ahead control incorporates dynamic models to optimize current and near-future decisions. The multi-step planning problem can be regarded as an optimal control problem (OCP), applying Pontryagin's minimum principle (PMP) to derive analytic solutions. Ntousakis et al. (2016) formulated an OCP for a merging vehicle to minimize multi-order derivatives of speed. Zhou et al. (2018) addressed two separate OCPs: one for a mainline vehicle to decelerate and create a gap, and another for an on-ramp vehicle to follow and merge into this gap. Subsequently, Zhou et al. (2019a) further studied this

problem by considering the minimum speed constraints of the facilitating manoeuvre. The advantage of this method is that the problem can be expressed as a linear quadratic controller clearly, and continuous-time optimal trajectories can be derived analytically. However, it is difficult to deal with complicated constraints, such as nonlinear kinematic equations and collision avoidance.

Alternatively, look-ahead control can also be formulated as a nonlinear programming (NLP) optimization model which can address more complex constraints numerically. Nilsson et al. (2016) proposed two loosely coupled convex quadratic programs to obtain longitudinal and lateral motion trajectories over a finite discrete time horizon, respectively. Letter et al. (2017) described trajectories in discrete-time form and optimized vehicle states in each moment. Similarly, Hu and Sun (2019) adopted the discrete-time form and developed two optimization models, one for cooperative lane-changing and another for merging manoeuvres. Karimi et al. (2020) formulated centralized quadratic problems for different triplets of vehicles composed of both conventional vehicles and CAV.

Additionally, model predictive control (MPC) is an advanced optimization-based control approach that has gained significant attention in recent years because its short horizon characteristic makes it well-suited for a real-time vehicle control, such as the scenarios of on-ramp merging. Shen et al. (2022a) and Shen et al. (2022b) studied fully distributed optimization-based control schemes using a platoon-centred MPC approaches, which consider both linear and nonlinear vehicle dynamics. They proposed a sequential convex programming (SCP)-based distributed scheme, based on operator splitting methods. Li et al. (2023) employed a longitudinal distributed MPC with a virtual car-following (CF) concept to expand the initial feasible set and to ensure three key aspects: asymptotic local stability, l_2 norm string stability, and safety. Chen et al. (2023b) adopted the sequential quadratic programming (SQP) method to solve the MPC-based 2D trajectory planning model.

2.2 Vehicle Sequence Planning

Sequencing planning is critical for determining the passing order of vehicles on mainline and on-ramp lanes to enhance traffic efficiency. Related studies mainly focus on merging scenarios involving a single mainline lane and a single on-ramp lane, and these strategies can be categorized into rule-based strategies, tree search-based strategies, integer programming-based strategies, and learning-based strategies.

(1) Rule-based Strategies

Rule-based strategies typically ensure completeness and can be executed within polynomial time. However, they often struggle with global optimality. With the concept of virtual vehicles proposed by Uno et al. (1999), Rios-Torres et al. (2017) adopted the first-in-first-out (FIFO) principle was used to assign merging time instants to all vehicles. Similarly, Chen et al. (2021) proposed a heuristic virtual rotation approach to align vehicles from different lanes into a virtual single line, thereby sorting vehicles based on their initial positions.

(2) Tree Search-based Strategies

Search-based strategies treat different vehicle orders as nodes (or states) and adopt various methods to search for optimal or suboptimal solutions. Pei et al. (2019) adopted dynamic programming to find the optimal solution, but the travel delay for each node was estimated rather than derived from actual trajectories. Chen et al. (2020) formulated an optimization model that incorporates three driving modes. All possible driving modes of multiple vehicles were enumerated and fed into the model to find the best solution. Tang et al. (2022) utilized the Monte Carlo Tree Search (MCTS) to find a suboptimal

sequence within a limited time. Shi et al. (2023b) used a depth-first search approach with heuristic pruning rules to obtain solutions.

(3) Optimization-based Strategies

Optimization-based approaches, which formulate the problem as mixed integer programming, can achieve optimal solutions, but their inherent NP-hard nature hinders real-time applications. Cao et al. (2014) implemented an MPC scheme to optimize the movements of both mainline and on-ramp vehicles, incorporating a nonlinear constraint to maintain a minimum relative distance. This approach also determined vehicle sequencing when generating trajectories but applying this nonlinear constraint to multiple vehicles proved challenging. Similarly, Xie et al. (2017) utilized a mixed-integer linear programming (MILP) model specifying absolute spacing between vehicles to maximize speed over a fixed number of time steps. This model considered all possible vehicle sequences, simplifying the problem by assuming a known duration for merging cooperation. Ye et al. (2019) introduced a bi-level optimization model where the upper level featured an MILP model aimed at minimizing travel delays and determining optimal merging times for each vehicle at the merge point. Subsequent trajectory planning was executed using either a discrete-time or a heuristic continuous-time trajectory planner, based on these optimal merging times. Mu et al. (2021) developed a complex MINLP model for merging two platoons and proposed a heuristic algorithm to solve it. However, this algorithm primarily searched solutions close to the sequence generated by the first-in-first-out rule, which does not guarantee optimality. Chen et al. (2021) proposed a hierarchical controller with tactical and operational layers. The tactical layer used a mixed integer programming model that employed the Helly car-following model and a proportional controller to determine vehicle acceleration, sequence, and terminal time instants. This layer enumerated all sequences and sampled terminal time instants to identify the best solution. The operational layer then utilized an OCP to generate the corresponding trajectories. In a subsequent study, Chen et al. (2022) applied cooperative game theory to model decision-making for the merging sequence.

(4) Learning-based Strategies

Learning-based approaches can learn a responsive approximation solution through trails and errors by constantly interacting with environments, while the safety and feasibility certificates for such methods cannot always be guaranteed due to the complexity of deep neural network structures.

Most studies attempt to directly control the high-level actions (e.g., yield, lane change) or low-level actions (acceleration or velocity value) of each vehicle using deep reinforcement learning (DRL). Hwang et al. (2022) proposed a finite state machine (FSM) containing four phases (ready for safe gap selection, gap approach, negotiation, and lane-change execution), in which the DRL policy is employed to execute lane changes. Chen et al. (2023a) formulated the mixed-traffic highway on-ramp merging problem as a multi-agent reinforcement learning (MARL) problem, where each vehicle makes high-level control decisions, such as turning left, turning right, cruising, speeding up, and slowing down. Hu et al. (2024) employed a Graph Convolutional Network (GCN) with Attention to capture high-dimensional features of CAV states and model their interactions. They designed an action space containing discrete manoeuvres, including acceleration, deceleration, and lane changes for a group of CAVs, and used DRL to learn an optimal policy.

Some studies treat the problem of CAVs navigating through conflicting areas as a combinatorial problem, using a sequence-to-sequence modelling approach combined with DRL to optimize vehicle right-of-way and reduce traffic delays. Zhang et al. (2023) adopted a pointer network structure to

determine the driving order of multiple CAVs crossing an intersection and then used MCTS to further refine the solutions. Jiang et al. (2024) applied the same structure to the single-lane freeway merging scenario.

2.3 Lane Change Planning

Lane changes are a fundamental aspect of multilane freeway merging, with research primarily focusing on both macroscopic and microscopic control strategies. Macroscopic strategies stem from the perspective of traffic flow management, aiming to fully utilize multi-lane capacities and balance the flow across multiple mainline lanes. In contrast, microscopic strategies focus on the detailed actions of individual vehicles, including lane-changing moments, gap acceptance, and vehicle velocity profiles during changing lanes.

(1) Macroscopic Control

Macroscopic Control focuses on managing the total number of lane changes to enhance traffic efficiency and reduce congestion. Hang et al. (2021) formulated the cooperative decision-making problem for CAVs using a coalitional game approach. They analysed lane-changing and lane-keeping decisions involving three to five CAVs based on four types of coalitions. Pan et al. (2021) proposed a control framework that combines ramp metering, variable speed limits, and lane changing controls for mixed freeway traffic, aiming to optimize traffic throughput. Tajdari et al. (2022) simplified the conservation law model into a linear time-invariant traffic flow model, applying a linear quadratic regulator (LQR) as the feedback controller to adjust lateral flows (i.e., lane changes) and the flow from an on-ramp to maximize throughput and prevent congestion.

(2) Microscopic Control

Microscopic Control focuses on the lane-changing decisions of individual vehicles. Li et al. (2021) defined reachable and attainable sets based on domain knowledge to determine whether a lane change should be executed. Zhang et al. (2022) proposed a hybrid model predictive control for platoon-based cooperative lane-changing control, with a machine learning-aided algorithm speeding up solution process. Liu et al. (2022) trained a policy network for lane selection using deep reinforcement learning, which aims to evenly distribute traffic flow across lanes and enhance individual vehicle speeds. Chen et al. (2022) modelled car-following, cruising, and cooperative lane- changing manoeuvres for a group of vehicles and determined the optimal manoeuvres through enumeration. Yang et al. (2023) modelled vehicle interactions as a cooperative game, with lane-changing decisions determined by a game cost matrix. Hence, decisions for a group of vehicles are comprised of a series of decisions between each pair of vehicles. Hu et al. (2023) designed safety-critical control method to conduct lane-changing manoeuvres for platooning CAVs using Control Barrier Functions (CBFs) and Control Lyapunov Functions (CLFs).

2.4 Summary

In summary, different strategies for CAV-based control across the three tasks are categorized as shown in Fig. 2.1. Trajectory planning is classified into reactive control and look-ahead control. Vehicle sequence planning can be approached using four methods: rule-based strategies, tree search-based strategies, integer programming-based strategies, and learning-based strategies. Lane change planning can be divided into macroscopic flow control and individual lane change decisions.

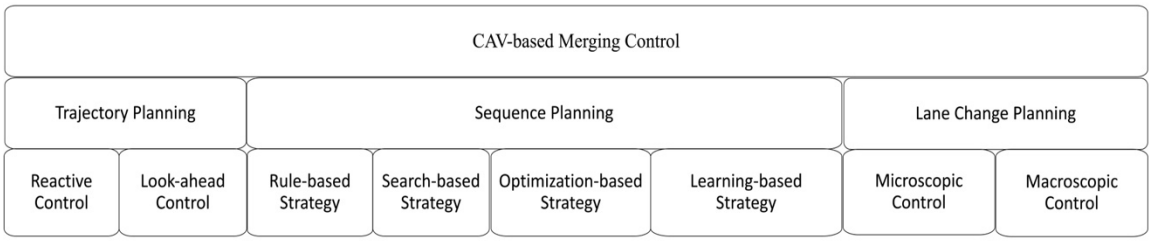


Figure 2.1. Categorization of Strategies for CAV-Based Control Tasks

Despite the variety of methods available, critical challenges still remain:

1) Integration of Multi-Task Strategies: While various strategies exist for each task, there is a lack of comprehensive frameworks that effectively integrate all three tasks. Most approaches treat these tasks independently, which prevents fully leveraging the potential of CAVs. Recent research has increasingly focused on integrating trajectory and sequence planning, but a holistic approach that considers all three tasks together in multi-lane scenarios is still lacking.
2) Scalability and Real-Time Application: Many existing methods struggle with scalability, particularly when applied to large numbers of vehicles, and face difficulties in real-time operation. There is a pressing need for more efficient algorithms capable of handling high-density traffic scenarios without compromising performance.

Motivated by the above gaps, this thesis focuses on designing efficient solution algorithms to address the challenges of the complicated problem by leveraging the inherent structure of the tasks involved. By enhancing scalability for real-time applications, the proposed methods aim to overcome the limitations of existing approaches, enabling more effective and practical CAV-based control.

Chapter 3

Integrated Optimization of Merging Sequences and Trajectories

This chapter models the task of cooperative merging of an on-ramp CAV stream and a mainline CAV stream as a mixed integer nonlinear programming problem that integrally optimizes merging sequences and vehicle trajectories. To solve the proposed model, we propose an integrated solution algorithm that includes a sequential search process for rapidly determining the optimal merge-in gaps and an iterative linear programming method for generating nonlinear trajectories and determining merging positions and moments. This chapter is organized as follows. Section 3.1 provides the preliminaries of considered scenarios, merging sequences and trajectories; Section 3.2 describes the modelling of cooperative merging of a mainline and an on-ramp traffic streams at a freeway on-ramp merging section as a MINLP problem; Section 3.3 develops a computationally efficient algorithm to solve the model; Section 3.4 validates the proposed method by numerical experiments; Section 3.5 concludes this chapter.

3.1 Preliminaries

We first introduce the merging scenario considered in this study, and then define two key notions that will be used in the formulation of the optimization problem, namely the merging sequence and the mathematical representation of trajectories.

(1) The Single-mainline Freeway Merging Scenario Considered in This Chapter

This chapter is oriented toward merging two streams of vehicles at a freeway on-ramp section. Fig.3.1 shows a typical freeway on-ramp merging section consisting of one mainline lane and one on-ramp lane connected with one acceleration lane. Initially, vehicles are assumed to be in a car-following mode. A roadside unit (RSU) is placed upstream of the intersection of the mainline and on-ramp lanes. RSU receives nearby CAVs' information, executes the proposed model, and then sends acceleration/deceleration commands to CAVs. In addition, a trigger point (TP), enabled by, for example, a loop detector, is placed at a suitable location of the on-ramp. Every time an on-ramp vehicle arrives at the TP, RSU starts a new control cycle and regards nearby mainline and on-ramp vehicles as a batch of planned vehicles. It means that, instead of driving in a car-following mode, these planned vehicles follow the instructions of the proposed model. After completing merging tasks, these vehicles revert to the car-following mode again. Fig. 3.1 shows two batches of planned vehicles. The formation of planned vehicles includes grouping on-ramp vehicles and mainline vehicles. In addition to the on-ramp vehicle that drives through the TP, its following on-ramp vehicles will be grouped if their time headway is less than a specified threshold. Then, the first and last on-ramp vehicles are mapped onto the mainline lane

to determine the planned mainline vehicles in this control cycle.

In this chapter, we only consider the longitudinal motion of merging vehicles, in line with many previous studies. In addition, the model proposed in this paper is not limited to single-mainline merging scenarios because in generalized multi-mainline ramp merging scenarios, lane-changing and on-ramp merging manoeuvres are usually expected to be performed on different segments of roads (Gao et al., 2022; Sharma et al., 2022). This means that we assume that the lane-changing behaviours of the mainline vehicles has already taken place before the cooperative merge starts. Therefore, to focus on the merging behaviours, we consider the scenario with one mainline lane and one on-ramp lane.

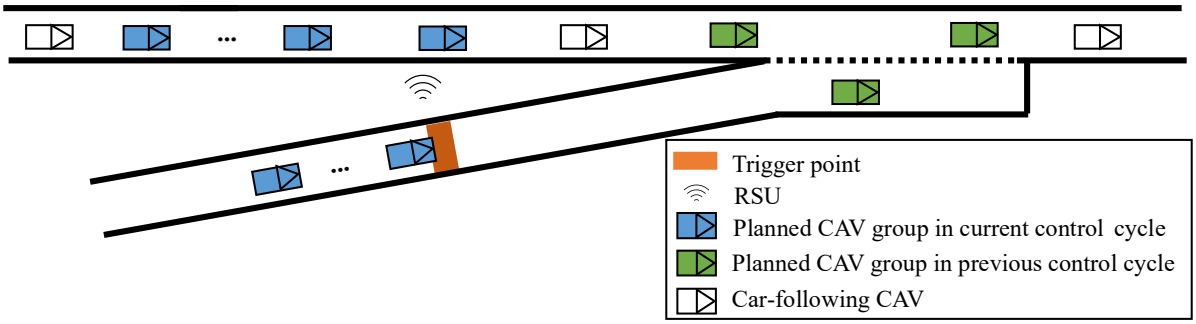


Figure 3.1. The on-ramp merging scenario.

As aforementioned, merging two streams of vehicles consists of two tasks: merging sequence scheduling and merging trajectory planning. Merging sequence scheduling determines the order of the controlled vehicles in the merged stream, i.e., after the merging is completed. Merging trajectory planning is responsible for generating trajectories for the mainline facilitating vehicles to create desired gaps and for the on-ramp vehicles to coordinate with these gaps adhering to the determined merging sequence. Furthermore, these two tasks are coupled with each other in the sense that the outcome of one of them can influence the outcome of the other. That is, the merging sequence influences the design of the trajectories, and the cost associated with these trajectories will in turn influence the determination of the merging sequence.

(2) Merging Sequence

The merging sequence refers to the order of the controlled vehicles right after the merging of the two streams of vehicles. It reflects the result of merge-in gap selections of on-ramp vehicles. In this paper, a merge-in gap refers to the gap between two mainline vehicles into which one or more on-ramp vehicles can merge. It ought to be noted that under our method, it is not necessary that each candidate merge-in gap will be selected. As shown in Fig. 3.2, let $L \coloneqq \{m, r\}$ denote the set of lanes, in which m and r refer to the mainline lane and on-ramp lane, respectively. Then, I_m and I_r represent a set of planned vehicles in the mainline lane and on-ramp lane, respectively, which means there are $|I_m|$ planned mainline vehicles and $|I_r|$ planned on-ramp vehicles in this control cycle. Besides, there may be a leading vehicle for these planned vehicles. Note that the leading vehicle can be either a mainline or on-ramp vehicle. It tracks designed trajectories from the previous control cycle, thereby not being involved in this control cycle. Then, let $I_m^+ \coloneqq \{1, \dots, |I_m| + 1\}$ denote the set of these merge-in gaps on the mainline lane. Both indexes of planned vehicles in each lane and merge-in gaps increase from one against the direction of traffic travel.

For on-ramp vehicle i, we define a binary vector $\boldsymbol{\gamma}_i \coloneqq [\gamma_{i,1}, \dots, \gamma_{i,k}, \dots, \gamma_{i,|I_m|+1}]^T$ where each element $\gamma_{i,k}$ represents whether on-ramp vehicle i, $i \in I_r$, chooses the merge-in gap k, $k \in I_m^+$, to merge. Thus, the existence of binary variables leads to the gap selection problem as an integer

programming problem.

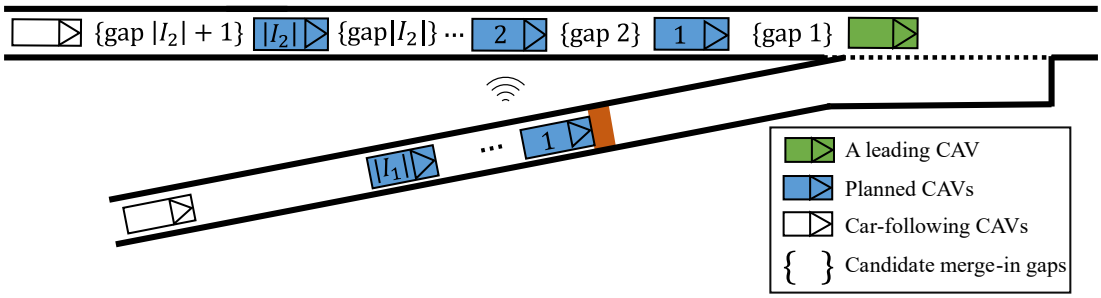


Figure 3.2. Merging sequence description.

The choice of merge-in gaps determines terminal positions of all vehicles. Specifically, if the i^{th} on-ramp vehicle merges into the k^{th} merge-in gap, then on-ramp vehicle i must keep a safe distance from both the $k-1^{th}$ and k^{th} mainline vehicles at the terminal time. Thus, $\boldsymbol{\gamma}_i$ is closely coupled with vehicles' trajectories.

(3) Mathematical Representation of Vehicle Trajectories

We consider a second-order model as the vehicles' kinematic model:

$$\dot{x}_{l,i}(t) = v_{l,i}(t), \tag{3.1}$$

$$\dot{v}_{l,i}(t) = a_{l,i}(t), \tag{3.2}$$

where l, $l \in L$, refers to lanes; i, $i \in I_l$, refers to the vehicle index; t refers to time; $a_{l,i}(t)$ is the acceleration; $v_{l,i}(t)$ is the velocity; $x_{l,i}(t)$ is the position. The nonholonomic property of vehicles is omitted since we focus on the longitudinal motion. The coordinate system adopted in this paper is the Frenet coordinate system with the road center line as the reference line (Werling et al., 2010). The longitudinal distance refers to the distance along the road center line.

Polynomial functions up to the third order are adopted to prescribe the solution space of the model (3.1)–(3.2):

$$\begin{bmatrix} x_{l,i}(t) \\ v_{l,i}(t) \\ a_{l,i}(t) \end{bmatrix} = \begin{bmatrix} t^3 & t^2 & t & 1 \\ 3t^2 & 2t & 1 & 0 \\ 6t & 2 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta_{l,i,3} \\ \theta_{l,i,2} \\ \theta_{l,i,1} \\ \theta_{l,i,0} \end{bmatrix}, \tag{3.3}$$

where $\boldsymbol{\theta}_{l,i} := [\theta_{l,i,3}, \theta_{l,i,2}, \theta_{l,i,1}, \theta_{l,i,0}]^T$ is the parameter vector of the polynomial curve of vehicle i on lane l. The employment of polynomials allows a reduced search space. However, the polynomial basis has difficulty in handling some constraints, such as velocity constraints and minimum distance between the positions of two adjacent vehicles at any time along the entire position trajectories. In order to incorporate these constraints into the polynomial representations of the trajectories, we introduce the Bézier curves.

A Bézier curve is defined by a set of control points. The control points include a start point, a terminal point, and shape-defined points. The first and last control points are always the endpoints of the curve. The order of a Bézier curve depends on the number of control points. A Bézier curve $P(s)$ of order n, obtained from $n+1$ control points, is written as:

$$P(s) = \sum_{j=0}^{n} B_j^n(s) P_j \quad s \in [0,1], \tag{3.4}$$

where $B_j^n(s)$ is the jth Bernstein polynomial of degree n, which can be defined by:

$$B_j^n(s) = \binom{n}{j} s^j (1-s)^{n-j} = \frac{n!}{j!\,(n-j)!} s^j (1-s)^{n-j}. \tag{3.5}$$

From (3.4), $P(s)$ is a linear combination of n Bernstein polynomial bases, and P_j is the coefficient of each $B_j^n(s)$. P_j can regulate the magnitude of a basis. Hence, the shape of a curve $P(s)$ can be manipulated by the control points.

A Bernstein polynomial has a convex hull property, which means that a Bernstein polynomial is confined within the convex hull of its control points (Cichella et al., 2018). Therefore, we can enforce continuous trajectory functions to be within a specified feasible set by setting the control points. Then, polynomial basis (3.3) can be equivalently rewritten in terms of the Bernstein basis, and the control points of $x_{l,i}(t)$, denoted by $\boldsymbol{P}_{l,i}^x$, can be represented by $\boldsymbol{\theta}_{l,i}$:

$$\begin{bmatrix} P_{l,i,3}^x \\ P_{l,i,2}^x \\ P_{l,i,1}^x \\ P_{l,i,0}^x \end{bmatrix} = \begin{bmatrix} t_f^3 & t_f^2 & t_f & 1 \\ 0 & t_f^2/3 & 2t_f/3 & 1 \\ 0 & 0 & t_f/3 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \theta_{l,i,3} \\ \theta_{l,i,2} \\ \theta_{l,i,1} \\ \theta_{l,i,0} \end{bmatrix}, \tag{3.6}$$

where t_f is terminal time of the trajectories. Similarly, the control points of $v_{l,i}(t)$, denoted by $\boldsymbol{P}_{l,i}^v$, are written as:

$$\begin{bmatrix} P_{l,i,2}^v \\ P_{l,i,1}^v \\ P_{l,i,0}^v \end{bmatrix} = \begin{bmatrix} 3t_f^2 & 2t_f & 1 & 0 \\ 0 & t_f & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \theta_{l,i,3} \\ \theta_{l,i,2} \\ \theta_{l,i,1} \\ \theta_{l,i,0} \end{bmatrix}. \tag{3.7}$$

The detailed derivation of (3.6) and (3.7) is given in Appendix B.

In our problem, t_f and the terminal states of vehicles are unknown, which causes (3.3, 3.6, 3.7) to be nonlinear equations and results in a nonlinear programming problem.

3.2 Model Formulation

A MINLP is formulated in this subsection to model the problem of integrated optimization of merging sequence generation and trajectory planning for cooperative on-ramp merging. The notation used in this paper is listed as follows.

Indices and Sets

L	set of lanes, $l \in L$, $l = \{m, r\}$, where m and r refer to the mainline lane and on-ramp lane.
I_m	set of planned mainline vehicles, $i\ (or\ j) \in I_m$.
I_r	set of planned on-ramp vehicles, $i\ (or\ j) \in I_r$.
I_m^+	set of merge-in gaps on the mainline lane, $k \in I_m^+$.

Input Parameters

$\tilde{v}$	desired terminal speed, e.g., the free-flow speed or initial speed of CAVs.
τ	minimum allowable time gap between vehicles.

s_0 buffer distance.
l_v length of a vehicle.
l_{zone} length of the cooperation zone.
w_m weighting factor for the travel delay of mainline vehicles.
w_r weighting factor for the travel delay of on-ramp vehicles.
$\bar{b}, \bar{a}$ maximum deceleration and acceleration.
$\underline{v}, \bar{v}$ minimum and maximum speeds.
$x_{l,i}^{init}$ initial position of vehicle i on lane l.
$v_{l,i}^{init}$ initial velocity of vehicle i on lane l.

Decision Variables

t_f terminal time when merging cooperation is finished. continuous variable.
$\boldsymbol{\theta}_{l,i}$ vector of trajectory parameters of vehicle i on lane l. $\boldsymbol{\theta}_{l,i} = \left[\theta_{l,i,3}, \theta_{l,i,2}, \theta_{l,i,1}, \theta_{l,i,0}\right]^T$. continuous variables.
$\boldsymbol{\gamma}_i$ vector indicating merge-in gap selection of on-ramp vehicle i, $i \in I_r$. $\boldsymbol{\gamma}_i = [\gamma_{i,1}, \dots, \gamma_{i,|I_m|+1}]^T$ in which $\gamma_{i,k}$, $i \in I_r$, $k \in I_m^+$, equals 1, if and only if the k^{th} candidate merge-in gap is chosen by the i^{th} on-ramp vehicle; 0, otherwise. binary variables.

Our objective is to have the two streams of vehicles merge safely and meanwhile minimize their total delay, through planning the trajectories for all these vehicles. As introduced earlier, we are aiming at a joint optimization task that can yield the optimal merging sequence and trajectories simultaneously. Therefore, our decision variables include $\boldsymbol{\theta}_{l,i}$, $l \in L$, $i \in I_l$; $\boldsymbol{\gamma}_i$, $i \in I_r$; and t_f. The model is detailed as follows.

$$\min_{\boldsymbol{\gamma}_i, \boldsymbol{\theta}_{l,i}, t_f} w_m \sum_{i \in I_m} \int_0^{t_f} \left(\tilde{v} - v_{m,i}(t)\right) dt \; + w_r \sum_{i \in I_r} \int_0^{t_f} \left(\tilde{v} - v_{r,i}(t)\right) dt \tag{3.8}$$

subject to:

$$x_{l,i}(t_f) \geq x_{l,i+1}(t_f) + \tau \cdot v_{l,i+1}(t_f) + l_v + s_0 \qquad \forall l \in L, i \in I_l \backslash \{|I_l|\} \tag{3.9}$$

$$x_{lead}(t_f) \geq x_{m,1}(t_f) + \tau \cdot v_{m,1}(t_f) + l_v + s_0 \tag{3.10}$$

$$x_{r,1}(t_f) \leq x_{r,1}(0) + l_{zone} \tag{3.11}$$

$$\sum_{k \in I_m^+} \gamma_{i,k} = 1 \qquad \forall i \in I_r \tag{3.12}$$

$$\gamma_{i+1,k} \leq \sum_{n=0}^{k} \gamma_{i,n} \qquad \forall i \in I_r \backslash \{|I_r|\}, k \in I_m^+ \tag{3.13}$$

$$x_{r,i}(t_f) \geq \boldsymbol{x}_m^T \tilde{\boldsymbol{\gamma}}_i + \tau \cdot v_{r,i}(t_f) + l_v + s_0 \qquad \forall i \in I_r \tag{3.14}$$

$$x_{r,i}(t_f) \leq \begin{bmatrix} x_{lead}(t_f) \\ \boldsymbol{x}_m \end{bmatrix}^T \boldsymbol{\gamma}_i - \tau \cdot v_{r,i}(t_f) - l_v - s_0 \qquad \forall i \in I_r \tag{3.15}$$

$$x_{l,i}(t) - x_{l,i+1}(t) \geq l_v + s_0 \qquad \forall t \in [0, t_f), l \in L, i \in I_l \backslash \{|I_l|\} \tag{3.16}$$

$$\bar{b} \leq a_{l,i}(t) \leq \bar{a} \qquad \forall t \in [0, t_f], l \in L, i \in I_l \tag{3.17}$$

$$\underline{v} \leq v_{l,i}(t) \leq \bar{v} \qquad \forall t \in [0, t_f], l \in L, i \in I_l \tag{3.18}$$

$$x_{l,i}(0) - x_{l,i}^{init} = 0; \; \forall l \in L, i \in I_l \tag{3.19}$$

$$v_{l,i}(0) - v_{l,i}^{init} = 0; \; \forall l \in L, i \in I_l. \tag{3.20}$$

1) Cost Function: To improve traffic efficiency, the objective function (3.8) is to minimize the difference between all vehicles' speeds and the desired speed, cumulated over the entire control horizon.

Moreover, consecutive tiny speed drops at the merging point can incur traffic breakdown [37]. Therefore, the cost function helps avoid breakdown.

2) Terminal Constraints: Constraints (3.9) ensure that terminal spacings between vehicles within the same lane must be greater than or equal to the distance corresponding to the minimum allowable time gap plus a vehicle length and buffer distance. Similarly, constraint (3.10) considers that the leading vehicle restricts the terminal position of the first mainline vehicle following behind it. In (3.10), $x_{lead}(t_f)$ refers to the terminal position of the leading vehicle. Note that if the leading vehicle is planned in the last control cycle, its trajectory is known; otherwise, it can be predicted by rolling out the car-following model. Additionally, constraint (3.11) enforces that the merging positions of on-ramp vehicles must be within the range of the cooperation zone, l_{zone}, due to the limited acceleration lane.

3) Constraints for Gap Selection: Obviously, each on-ramp vehicle can only merge into one merge-in gap, as constraints (3.12). Moreover, because overtaking is not allowed, constraints (3.13) state that an on-ramp vehicle can only choose a gap from the merge-in gap chosen by its preceding on-ramp vehicle and the following upstream merge-in gaps. Note that adjacent on-ramp vehicles can choose the same merge-in gap, i.e., it is permissible for a mainline vehicle to facilitate several on-ramp vehicles. Furthermore, when a merge-in gap is selected, the on-ramp vehicle and two mainline vehicles ahead and behind the selected gap are linked spatially. Correspondingly, constraints (3.14) and (3.15) impose that at the terminal time, if the k^{th} gap is chosen by the i^{th} on-ramp vehicle, both the $k-1^{th}$ and k^{th} mainline vehicles need to spatially form a minimum distance from the i^{th} on-ramp vehicle, respectively. This minimum distance corresponds to the minimum allowable time gap plus a vehicle length and buffer distance. Constraints (3.14) are for the spatial relation between on-ramp vehicles and mainline vehicles behind merge-in gaps. In (3.14), the vector $\boldsymbol{x}_m \coloneqq [x_{m,1}(t_f), \dots, x_{m,|l_m|}(t_f)]^T$ contains terminal positions of all mainline vehicles; $\widetilde{\boldsymbol{\gamma}}_i \coloneqq [\gamma_{i,1}, \dots, \gamma_{i,|l_m|}]^T$ contains all merge-in gap options except the last one, $\gamma_{i,|l_m|+1}$. Likewise, constraints (3.15) are for the spatial relation between on-ramp vehicles and mainline vehicles ahead of merge-in gaps. The physical implication of keeping at least one minimum allowable time gap is to smoothly switch to the car-following strategy (constant time-gap CACC or ACC) when the merging process is completed.

4) Constraints of Collision Avoidance: During the cooperation, constraints (3.16) guarantee that collision will not happen between vehicles within the same lane. To ensure safety at any time, rather than only at discrete time nodes, the spacing between two trajectories, $x_{l,i}(t) - x_{l,i+1}(t)$, is represented by the Bézier basis. Thanks to its convex hull property, constraints (3.16) can be concisely expressed as the control points of $x_{l,i}(t) - x_{l,i+1}(t)$ must be greater than or equal to the vehicle length plus the minimum spacing. Corresponding inequality equations are written as:

$$P^x_{l,i,1} - P^x_{l,i+1,1} \geq l_v + s_0, \tag{3.21}$$

$$P^x_{l,i,2} - P^x_{l,i+1,2} \geq l_v + s_0, \tag{3.22}$$

where control points are expressed as (3.6).

5) Constraints on Vehicle Speeds and Accelerations: Constraints (3.17) and (3.18) bound the acceleration and velocity of all vehicles at any time. Because the acceleration of a vehicle is linear in time, referring to constraint (3.3), acceleration constraints only need to be imposed at the initial time and terminal time. Therefore, constraints (3.17) can be equivalently re-written as:

$$\bar{b} \leq a_{l,i}(0) \leq \bar{a}, \tag{3.23}$$

$$\bar{b} \le a_{l,i}(t_f) \le \bar{a}. \tag{3.24}$$

To ensure velocity constraints at any time, control points of velocity functions are restricted to the lower and upper bounds of velocity, and constraints (3.18) are rewritten as:

$$\underline{v} \le P^{v}_{l,i,1} \le \bar{v}, \tag{3.25}$$

$$\underline{v} \le P^{v}_{l,i,2} \le \bar{v}, \tag{3.26}$$

where control points are expressed as (3.7).

6) Constraints of Initial Conditions: The starting position and speed of each vehicle are specified in constraints (3.19) and (3.20).

In sum, constraints (3.8–3.20) represent the proposed mathematical problem formulation. Substantial nonlinearities exist in the cost function and the constraints. Furthermore, constraints (3.14) and (3.15) introduce binary vectors. These render the optimization problem difficult to solve. Hence, we propose a solution algorithm to solve it efficiently.

3.3 An Integrated Solution Algorithm

In the above, we have formulated the task of merging sequence optimization and the task of vehicle trajectory optimization into one single mathematical model, i.e. the proposed MINLP, where $\boldsymbol{\theta}_{l,i}$, $l \in L$, $i \in I_l$, and t_f are continuous decision variables defining trajectories, and $\boldsymbol{\gamma}_i$, $i \in I_r$, are binary decision variables defining merging sequence. When solving the proposed MINLP, the trajectory variables and the merging sequence variables must be solved jointly, in the sense that neither of them can be determined independently of the other or can be determined earlier than the other. Only if this can be achieved, can we say that the proposed method is an integrated approach to optimal merging sequence generation and trajectory planning.

(1) Overview

We present an overview of the proposed algorithm, how we simultaneously solve the merging sequence optimization and trajectory optimization tasks in an integrated way, rather than a heuristic two-stage approach that first deals with merging sequence optimization and then trajectory optimization. First, for a group of on-ramp vehicles, the proposed algorithm determines their optimal merge-in gaps one by one, starting from the first on-ramp vehicle (i.e., the most downstream one) and moving upstream. Second, for a certain on-ramp vehicle, its optimal merge-in gap is determined from all the candidate merge-in gaps for this on-ramp vehicle. Last, for each candidate merge-in gap, we optimize the trajectory parameters of all mainline vehicles, this on-ramp vehicle and all the on-ramp vehicles ahead of this on-ramp vehicle. The incurred cost is computed and recorded. Therefore, the optimal merge-in gap can be selected as the one that has incurred the lowest cost among all the candidate gaps.

The above approach implies: 1) the results of trajectory optimization have influences on the selection of optimal merge-in gaps, and vice versa; 2) if and only if the optimal merge-in gap for the last on-ramp vehicle in this CAV group is determined, both the optimal merging sequence and corresponding trajectories of all involved vehicles in this CAV group are determined jointly and simultaneously. Neither of them can be determined independently of the other or can be determined earlier than the other. It is in such a way that the proposed solution algorithm integrates the task of merging sequence optimization and the task of trajectory optimization into one unified process.

The algorithm is formally presented in TABLE 3.1, where the embedded Algorithm 2 optimizes vehicle trajectories, which will be introduced in Section III. E.

In Section III. B., we describe in detail by a simple case how the proposed algorithm determines the optimal merge-in gaps for a group of on-ramp vehicles one by one. In Section III. C., we explain the advantage of such a sequential determination of on-ramp vehicles' optimal merge-in gaps – the size of search space can be reduced. In Section III. D., we prove that such a sequential way will indeed generate the same optimal results as one that considers all the on-ramp vehicles simultaneously, so have we verified by experiment in Section IV. H. Finally, in Section III. E., we present the iterative linear programming method, which is responsible for the trajectory generation, i.e., Algorithm 2 embedded in the integrated solution algorithm.

Table 3.1. Proposed integrated solution algorithm.

Algorithm 1. Integrated Solution Algorithm for Joint Optimization of Merging Sequence and Vehicle Trajectories

Input: $|I_m|$ mainline CAVs and $|I_r|$ on-ramp CAVs

Output: Optimal merging sequence $\{\boldsymbol{\gamma}_1^*, \boldsymbol{\gamma}_1^*, \dots, \boldsymbol{\gamma}_{|I_r|}^*\}$

1 **Algorithm 2** returns the cost and the trajectories for the first on-ramp CAV and the $|I_m|$ mainline CAVs under different $\boldsymbol{\gamma}_1$, to obtain $\boldsymbol{\gamma}_1^*$.
2 **for** $i := 2$ to $|I_r|$ **do**
3 **Algorithm 2** returns the cost and the trajectories for i (from the 1^{st} to i^{th}) on-ramp CAVs and the $|I_m|$ mainline CAVs under different $\boldsymbol{\gamma}_i$, with $\boldsymbol{\gamma}_1^*, \dots, \boldsymbol{\gamma}_{i-1}^*$ as conditions, selecting $\boldsymbol{\gamma}_i^*$ that yields the lowest cost.
4 **end for**
5 return $\{\boldsymbol{\gamma}_1^*, \boldsymbol{\gamma}_2^*, \dots, \boldsymbol{\gamma}_{|I_r|}^*\}$

(2) Determining Optimal Merge-in Gaps for On-Ramp Vehicles One by One

Mixed integer problems can be solved by the classic branch-and-bound (B&B) method. However, the size of the B&B's solution space in this problem is $2^{(|I_m|+1)|I_r|}$, where $|I_m|$ and $|I_r|$ are the number of mainline and on-ramp vehicles respectively. This implies that the size of the search space grows exponentially as the number of involved vehicles increases. To overcome this issue and exploit the characteristics of the merging scenario, the proposed process determines the optimal merge-in gap for each on-ramp vehicle one by one, starting from the first (i.e., the most downstream) on-ramp vehicle and moving upstream.

Specifically, in the first step, all the combinations of $\boldsymbol{\gamma}_1$, i.e., all possible merge-in gap selections for the first on-ramp vehicle, are enumerated, and the trajectories of the first on-ramp vehicle and all mainline vehicles are planned under each combination to determine $\boldsymbol{\gamma}_1^*$. Then, the first and the second on-ramp vehicles and all the mainline vehicles are planned conditioning on $\boldsymbol{\gamma}_1^*$ to determine $\boldsymbol{\gamma}_2^*$. Analogously, the first to the i^{th} on-ramp vehicles and all the mainline vehicles are planned conditioning on $\boldsymbol{\gamma}_1^*, \dots, \boldsymbol{\gamma}_{i-1}^*$ to determine $\boldsymbol{\gamma}_i^*$. This process terminates until the best merge-in gap selection of the last on-ramp vehicle, i.e., $\boldsymbol{\gamma}_{|I_r|}^*$, is decided.

Take the example of there being two mainline and two on-ramp vehicles in a control cycle. The corresponding searching process for this example is illustrated in Fig. 3.3. Obviously, there exist three merge-in gaps for merging vehicles because there are two mainline vehicles. Fig. 3.3(a) depicts that all

possible merge-in gaps of the first merging vehicle will be enumerated to find $\boldsymbol{\gamma}_1^*$. Fig. 3.3(b) shows that all the three possible gaps need to be searched again for the second merging vehicle if the first merging vehicle chooses the first gap, i.e., $\boldsymbol{\gamma}_1^* = [1, 0, 0]^T$. Similarly, Fig. 3.3(c) illustrates that the search space of the second merging vehicle narrowed to two options when the first merging vehicle chooses the second gap. Lastly, Fig. 3.3(d) shows that the second on-ramp vehicle has only one merging order option, i.e., $[0, 0, 1]^T$, if its leading on-ramp vehicle chooses the last merge-in gap. In Step 3 of Algorithm 1, when determining the merging sequence of the remaining vehicles, the merging sequence of the preceding vehicles has already been fixed, but their trajectories would be replanned to obtain a unified t_f and optimal trajectories for all vehicles. Moreover, under Algorithm 1, thanks to the sequential determination of optimal merge-in gaps for the on-ramp vehicles described above, the original MINLP is reduced to nonlinear program (NLP) to optimize vehicle trajectories, as in Step 1 and Step 3 of Algorithm 1.

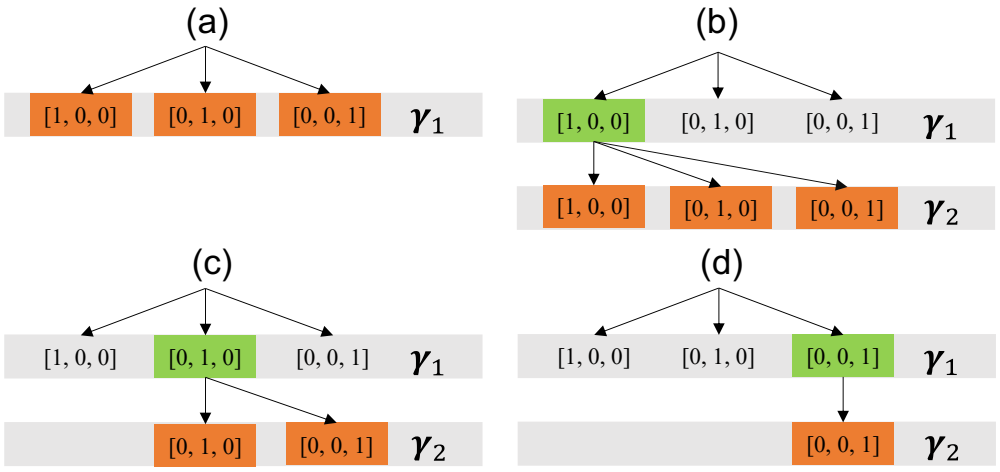


Figure 3.3. The search tree of merging sequences.

(a) the search space for the first on-ramp vehicle; (b) (c) (d) three possible search cases for the second on-ramp vehicle

(3) Reduced Search

The proposed optimal sequential searching process can efficiently reduce the search space due to the following three reasons. First, the number of combinations of $\boldsymbol{\gamma}_i$, $i \in I_r$, is $|I_m| + 1$ since (3.12) imposes that only one element in $[\gamma_{i,1}, \gamma_{i,2}, \dots, \gamma_{i,|I_m|+1}]^T$ can be one. This greatly reduces the solution space from $2^{|I_m|+1}$ to $|I_m| + 1$ for on-ramp vehicle i. Hence, for all $|I_r|$ on-ramp vehicles, the search space is reduced to $(|I_m| + 1)^{|I_r|}$, instead of $2^{(|I_m|+1)|I_r|}$. Second, (3.13) indicates that an on-ramp vehicle can only select the same mainline gap chosen by its preceding on-ramp vehicle or the following gaps. In other words, an on-ramp vehicle cannot choose a mainline gap downstream of the mainline gap chosen by its preceding on-ramp vehicle. Third, as presented in Algorithm 1, the merging sequence is sequentially solved from $\boldsymbol{\gamma}_1$ to $\boldsymbol{\gamma}_{|I_r|}$ so that a tree-structured search space is constituted, and the tree's depth is $|I_r|$. As an illustration, Fig. 3.3 shows a tree with depth two when there are two mainline and two on-ramp vehicles.

(4) Optimality

Now we prove that the proposed optimal sequential searching process can indeed generate the

optimal merging sequence. To this end, rather than work with the original objective function, (3.8), we instead work with an equivalent objective function:

$$\{\boldsymbol{\gamma}_1^*, \boldsymbol{\gamma}_2^*, \dots, \boldsymbol{\gamma}_{|I_r|}^*\} = \underset{\boldsymbol{\gamma}_1,\dots,\boldsymbol{\gamma}_{|I_r|}}{\operatorname{argmin}} \sum_{i\in I_m} t_{m,i} + \sum_{j\in I_r} t_{r,j}\,, \tag{3.27}$$

where $t_{m,i}, t_{r,j}$ are the travel time of the i^{th} mainline vehicle and the j^{th} on-ramp vehicle, respectively.

To see the equivalency between (3.8) and (3.27), first we note that (3.8) is actually to maximize the speeds of all the vehicles over the whole cooperation zone. This is because in (3.8), although the time to complete the cooperative merging maneuver, t_f, is not pre-determined, but before this time, all the vehicles will try to stay close to the maximum speed, $\tilde{v}$, and after this time, all the vehicles will traverse the remained part of the cooperation zone at the maximum speed, $\tilde{v}$. On the other hand, (3.27) minimizes the total travel time of all the vehicles over the whole cooperation zone, which is equivalent to maximizing the speeds of all the vehicles over the whole cooperation zone.

Note that (3.27) does not explicitly involve the continuous decision variables, t_f and $\boldsymbol{\theta}_{l,i}$, but only the binary variables $\boldsymbol{\gamma}_i$, $i \in I_r$, which will greatly simplify the following analysis. This is justified because, per the proposed optimal sequential search algorithm, to determine $\boldsymbol{\gamma}_i^*$, $i \in I_r$, for every enumerated $\boldsymbol{\gamma}_i$, $i \in I_r$, t_f and $\boldsymbol{\theta}_{l,i}$ will be solved from a nonlinear program to be explained in Section III.B., so the associated trajectory cost can be computed; $\boldsymbol{\gamma}_i^*$ will then be determined after all the enumerations.

Now we utilize the characteristics of single-lane merging problem to derive a necessary condition for the optimality of (3.27). In below, we claim this necessary condition and then prove it.

Proposition 1: The merge-in gap selection of an on-ramp vehicle can be optimal only if the merge-in gap selections of all the preceding on-ramp vehicles are optimal. That is, $\boldsymbol{\gamma}_1^*, \dots, \boldsymbol{\gamma}_{p-1}^*$ is a necessary condition for $\boldsymbol{\gamma}_p^*$, $\forall p \in I_r$.

Proof: It is straightforward that after merging, mainline and on-ramp vehicles form a platoon, which means that at this time, the leading vehicle of an on-ramp vehicle is either its initial preceding on-ramp vehicle or a mainline vehicle, and the same goes for the leading vehicle of a mainline vehicle. Therefore, the relation between the travel time of two consecutive vehicles after merging is given by (3.28) and (3.29).

$$t_{r,j+1} = t_{r,j} + min\{\tau, \tau_0\} \; or \; t_{m,i} + min\{\tau, \tau_0\} \tag{3.28}$$

$$t_{m,i} = t_{m,i-1} + min\{\tau, \tau_0\} \; or \; t_{r,j} + min\{\tau, \tau_0\}\,, \tag{3.29}$$

where τ is the minimum allowable time gap; τ_0 is initial time gap between these two vehicles. (3.28) and (3.29) indicate that the travel time of a vehicle is dependent on its leading vehicle. Moreover, (3.28) consists of two cases. In the first case, the j^{th} on-ramp vehicle is the leading vehicle of the $j+1^{th}$ on-ramp vehicle, and therefore, the optimal $\mathrm{t}_{r,j}$, $\mathrm{t}_{r,j}^*$, is the necessary condition for $t_{r,j+1}^*$. In the second case, the leading vehicle is the i^{th} mainline vehicle. Likewise, $t_{r,j+1}^*$ requires $\mathrm{t}_{m,i}^*$. Then, $\mathrm{t}_{m,i}^*$ implies $\mathrm{t}_{r,j}^*$ because the j^{th} on-ramp vehicle must be downstream of the i^{th} mainline vehicle, as reflected by (3.29). Therefore, in both cases, $t_{r,j}^*$ is a necessary condition for $t_{r,j+1}^*$. Similarly, a necessary condition for $t_{m,i+1}^*$ is $t_{m,i}^*$.

Based on the necessary condition and the assumption that $\boldsymbol{\gamma}_1^*, \dots, \boldsymbol{\gamma}_{p-1}^*$ corresponds to minimum

$\sum_{i=1}^{|I_m|} t_{m,i} + \sum_{j=1}^{p-1} t_{r,j}$, we deduce that the $\boldsymbol{\gamma}_1^*, \ldots, \boldsymbol{\gamma}_{p-1}^*$ is a necessary condition for $\boldsymbol{\gamma}_p^*$. Therefore, Proposition 1 holds. □

The above necessary condition leads to the proposed sequential searching algorithm, which iteratively solves problem (3.30) for each on-ramp vehicle, from the most downstream on-ramp vehicle to the most upstream on-ramp vehicle, to determine the optimal merging sequence and the trajectories.

$$\tilde{\boldsymbol{\gamma}}_1^*, \ldots, \tilde{\boldsymbol{\gamma}}_p^* = \underset{\boldsymbol{\gamma}_1,\ldots,\boldsymbol{\gamma}_p}{\operatorname{argmin}} \sum_{i=1}^{|I_m|} t_{m,i} + \sum_{j=1}^{p} t_{r,j} \tag{3.30}$$

$$\text{subject to: } \boldsymbol{\gamma}_1 = \tilde{\boldsymbol{\gamma}}_1^*, \ \ldots, \boldsymbol{\gamma}_{p-1} = \tilde{\boldsymbol{\gamma}}_{p-1}^*; \forall p \in I_r$$

Problem (3.30) can be considered as to determine the optimal merge-in gap of the p^{th} on-ramp vehicle, $\tilde{\boldsymbol{\gamma}}_p^*$, conditioning on fixed $\tilde{\boldsymbol{\gamma}}_1^*, \ldots \tilde{\boldsymbol{\gamma}}_{p-1}^*$. We have added "~" over $\boldsymbol{\gamma}_i^*$, $i \in I_r$ to be rigorous, i.e., to honour the subtle fact that although problem (3.30) is the consequence of a necessary optimality condition of problem (3.27), but they are two different problems. In Appendix A, we offer an alternative way to show Proposition 1.

(5) An Iterative LP Method for Solving the NLP Trajectory Planning Model

Integer variables are searched by above sequential searching algorithm, but each subproblem in the search tree is still nonlinear w.r.t. the continuous variables $\boldsymbol{\theta}_{l,i}$ and t_f. It can be observed that the cost function (3.8) and constraints (3.9–3.20) are linear combinations of nonlinear (3.3, 3.6, 3.7). However, (3.3, 3.6, 3.7) can be treated as linear functions with respect to $\boldsymbol{\theta}_{l,i}$ if t_f is known. Based on this point, each node in the search space can be seen as a LP problem when a feasible t_f is given. Therefore, the gradient descent method is adopted to iteratively update t_f. Concretely, the LP subproblem associated to the t_f^k in kth iteration is first solved to obtain the optimal $\boldsymbol{\theta}_{l,i}^k$, and then the resulting objective value can provide gradient direction for the subsequent t_f^{k+1} update. The detailed procedure to solve each subproblem is summarized in Table 3.2. We can see that the gradient of objective function is calculated numerically, and the step size is determined with the Armijo rule.

Table 3.2. The iterative LP method

Algorithm 2. An Iterative LP Method for Trajectory Optimization

Input: A node $\tilde{o}$ in the search tree

Output: Optimal time t_f^*; trajectories $\boldsymbol{\theta}_{l,i}^*$; cost value J^* for node $\tilde{o}$

$k \leftarrow 0$

Given a feasible t_f^0, solve $\tilde{o}$ to obtain J^0.

Repeat

  Given t_f^k, solve $\tilde{o}$ to obtain the $\boldsymbol{\theta}_{l,i}^k$ and the cost value $J(t_f^k)$

  Given $t_f^k + \epsilon$, solve $\tilde{o}$ to obtain the cost value $J\left(t_f^k + \epsilon\right)$

  $d^k \leftarrow -(J\left(t_f^k + \epsilon\right) - J(t_f^k))/\epsilon$

  Pick step size α^k according to the Armijo rule.

  $t_f^{k+1} \leftarrow t_f^k + \alpha^k d^k$

  Given t_f^{k+1}, solve $\tilde{o}$ to obtain J^{k+1}

  $k \leftarrow k + 1$

until $\left|J^k - J^{k-1}\right| > \xi$

return $t_f^* \leftarrow t_f^k$, $\boldsymbol{\theta}_{l,i}^* \leftarrow \boldsymbol{\theta}_{l,i}^k$, $J^* \leftarrow J^k$

(6) Recursive Implementation for Feedback

In practical implementation, it is crucial to deal with various disturbances that may occur, such as variations in vehicle speed or unexpected events such as vehicles abruptly changing lanes from other main lanes. Therefore, a feedback loop is necessary to deal with these disturbances.

To this end, the trajectories of CAV groups that have passed the trigger point (TP) are updated periodically. Specifically, within a predefined update interval, denoted as t_{up}, the proposed approach replans trajectories based on the updated surrounding vehicles information, including their positions and speeds. This process continues until the cooperative merge task is successfully completed.

3.4 Numerical Experiments

To evaluate the proposed methodology, the computational experiments are performed on a workstation (16 cores of CPUs; 3.2GHz). The mathematical models and algorithms proposed in this study are implemented in Python (CasADi).

(1) Experimental Settings

Experiments are simulated in SUMO1.14, a microscopic traffic simulation tool. The SUMO internal model, the Wiedemann 99 model, is applied to simulate the car-following behaviour. The adjusted driving behaviour and vehicles' parameter values are listed in Table 3.3. All other parameters assume default values.

Table 3.3. Parameter setting

Parameter	Value	Unit
v_f	120	km/h
$\bar{a}$	2.75	$\mathrm{m/s^2}$
$\bar{b}$	2.75	$\mathrm{m/s^2}$
s_0	1.5	m
l_v	4.37	m
τ	1.5	s

As shown in Fig. 3.4, the simulated freeway segment consists of one mainline lane and one on-ramp. The mainline is 4.1 kilometres long, which contains a 2-kilometer 'warm-up' segment. The cooperation zone, which includes the merging and preparation sections, is 800 meters long. The so-called preparation section is from the on-ramp node up to the position in the mainline as a mapping of the trigger point in the on-ramp. The downstream segment is 1.5 kilometres, so that affected road sections are fully covered.

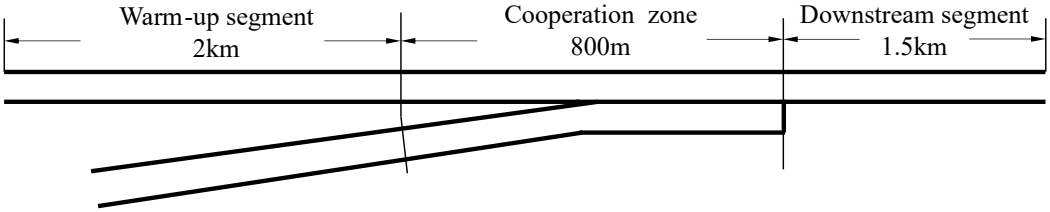


Figure 3.4. Road setting.

To evaluate the effectiveness of the proposed method, we test different combinations of mainline/on-ramp demand ratio and the total demand. The three tested mainline/on-ramp demand ratios are 4/1, 3/1, and 2/1. The four tested total demand values (1932, 2082, 2147, 2190) are respectively 90%, 97%, 100%, and 102% of the theoretical capacity. Therefore, in total there are 12 scenarios to be tested. The experiment for each scenario runs for one hour in SUMO.

The proposed method is compared with three baseline methods to evaluate its performance:

- VROCP: the virtual rotation optimal control problem. First, the merging sequence is determined by a heuristic first-in-first-out way, and then the optimal control is applied to generate optimal vehicle trajectories (Zhou et al. 2018).
- FTOCP: the fixed-time optimal control problem. First, a vehicle sequence is determined using the FIFO method, and then trajectories are planned over a fixed merge time.
- MCTS-DA: Monte Carlo Tree Search-based decomposition algorithm to find the optimal vehicle sequence. The time-optimal merge sequence is obtained by repeatedly solving a mixed integer programming model using the MCTS method (Tang et al. 2018).

(2) Cooperative Merging in the Presence of a Metering Signal

Ramp metering signal control (one-car-per-green) can make on-ramp vehicles arrive uniformly at the event trigger point to avoid inputting large disturbances to the mainline. Both the proposed approach and the benchmark method, VROCP, are tested assuming the presence of such a ramp metering signal control. The results are in Table 3.4. The third and fourth columns present travel delays of the two methods, respectively. The last column shows the percentage of reduction of delay.

Table 3.4 shows that the proposed approach has similar travel delay to VROCP when the mainline/on-ramp demand split ratio is 4:1. On the other hand, the proposed approach has significant improvement when the split ratio is 3:1 and 2:1.

When the ratio is 4:1, on-ramp vehicles are sparsely distributed. There is little mutual influence between on-ramp vehicles, and adjacent on-ramp vehicles do not affect the choice of each other's facilitating mainline vehicles. The proposed approach chooses the same merging sequence, i.e., FIFO as VROCP does. Therefore, both methods yield similar delays.

When the on-ramp demand is high, with a ratio of 2:1, intense interactions occur. This means that the arrival of the following on-ramp vehicles coincides with the ongoing merging and deceleration process of the preceding on-ramp and mainline vehicles. As a result, more severe slowdowns and a significant increase in delay occurs.

In this intensive interaction scenario, the delay improvement ($D.I.$) becomes much more evident, due to the multi-vehicle cooperation advantage of the proposed model. More specifically, the proposed model considers cooperation between two streams of vehicles, rather than just one on-ramp vehicle merging into two mainline vehicles. For example, a facilitating mainline vehicle can create a large gap to accommodate multiple on-ramp merging vehicles as a group to minimize the overall delay.

Table 3.4. Total travel delay comparison with ramp metering

Mainline/ On-Ramp Demand Ratio	Total Demand (veh/hour)	Proposed $D_1\ (s)$	VROCP $D_2\ (s)$	$D.I.$
4:1	1932	828.4	832.5	0.5%
4:1	2082	886.1	909.8	2.6%
4:1	2147	915.2	922.9	0.8%
4:1	2190	938.0	948.5	1.1%
3:1	1932	912.1	918.7	0.7%
3:1	2082	1115.3	1244.3	10.4%
3:1	2147	1182.8	1310.8	9.8%
3:1	2190	1226.3	1413.3	13.2%
2:1	1932	956.0	956.57	0%
2:1	2082	1182.8	2432.3	**51.4%**
2:1	2147	2981.8	4048.2	**26.3%**
2:1	2190	3362.4	4212.7	**20.2%**

Note: (1) D_1 and D_2 refer to total travel delay by the proposed approach and VROCP, respectively; (2) D.I. refers to delay improvement and is defined as $D.I. := \frac{D_2 - D_1}{D_2}$.

(3) Cooperative Merging Without a Metering Signal

Without metering control, on-ramp vehicles arrive at the on-ramp randomly, so they may form a big disturbance to the mainline traffic. In this case, we also compare the proposed approach with VROCP under the same traffic conditions.

Table 3.5. Total travel delay comparison without ramp metering

Mainline/ On-Ramp Demand Ratio	Total Demand (veh/hour)	Proposed $D_1\ (s)$	VROCP $D_2\ (s)$	$D.I.$
4:1	1932	1658.1	1958.1	15.3%
4:1	2082	1866.7	2203.1	15.3%
4:1	2147	1916.9	2242.6	14.5%
4:1	2190	2008.3	2366.3	15.1%
3:1	1932	2061.3	2355.4	12.5%
3:1	2082	2704.1	3065.1	11.8%
3:1	2147	2976.2	3484.7	14.6%
3:1	2190	3150.1	3590.7	12.3%
2:1	1932	1989.5	Failure	—
2:1	2082	3199.7	Failure	—
2:1	2147	4235.4	Failure	—
2:1	2190	4962.9	Failure	—

Note: (1) D_1, D_2 refer to total travel delay; (2) $D.I. := \frac{D_2 - D_1}{D_2}$.

From Table 3.5, we see that travel delay increases both as the inflow demand increases and as the number of on-ramp vehicles increases. Also, the proposed approach has at least an 11% improvement over the baseline under any condition. Moreover, when the split ratio is 2:1, VROCP would even lead to collisions because it only considers the leading mainline vehicle and ignores the leading on-ramp vehicle when planning trajectories.

In addition to delay reduction, the proposed approach also reduces the variation of speed (VS). High VS is found to be a precursor of collision accidents (Aashto (2001); Lee et al. (2003); Lee et al. (2005)). To illustrate the improvement in VS achieved by the proposed approach, we compare the speed

contour achieved by the proposed approach with the speed contour achieved by the SUMO default control. The evaluation time-space region is specified as [1800, 2500] sec × [1500, 3500] m.

Fig. 3.5(a) is the speed contour of the SUMO default control. The intersection of mainline and ramp is at 2.0 km, and we can observe that speed fluctuation starts from here and extends to the downstream, with significant velocity drops at 1900 and 2350 seconds. Fig. 3.5(b) presents the speed contour plot of the proposed approach. We see that deceleration occurs at 1.5 km since the event trigger point is set there. Obviously, the proposed approach can mitigate speed variance. As a result, VS and travel delays are all reduced.

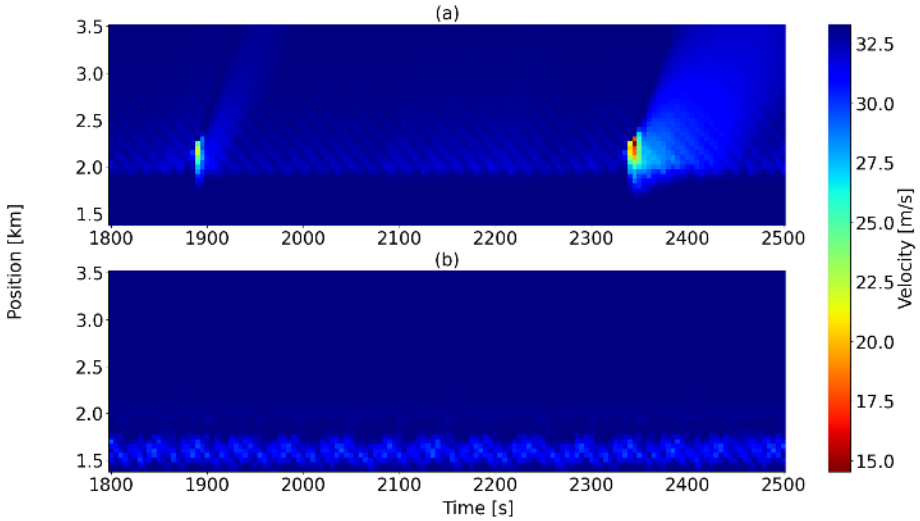


Figure 3.5. Speed contour in the mainline

(a) with SUMO default control; (b) with the proposed approach.

In addition to traffic efficiency, we validate that the proposed approach guarantees CTH rules since safety is an essential consideration. Fig. 3.6 shows the time gap statistics for all CAVs at multiple locations on the mainline in a one-hour simulation period. Fig. 3.6(a) illustrates time gap distributions under the SUMO default behaviour, and we can observe that time gaps downstream of the 2.0 km location are much less than the desired time gap (1.5s) and even close to 0, which is risky and unreasonable. On the contrary, under the control of the proposed approach, time gaps can always be no less than the desired gap as shown in Fig. 3.6(b). Fig. 3.6(b) also indicates that before vehicles drive through the trigger point (at 1.5 km location), the initial time gaps are concentrated around 2.2 seconds. Over the coordination zone, i.e., from 1.5 km to 2 km, some of the time gaps increase to between 3 and 4 seconds, which corresponds to the slowing down of the facilitating mainline vehicles to generate gaps for the on-ramp traffic; downstream of the 2.0 km location, the gaps reduce to around 1.5 seconds because the on-ramp merging vehicles have merged into the mainline.

(4) Analyses on Vehicles' Trajectories

One advantage of the proposed model is the ability to allow multiple on-ramp vehicles to merge into the gap between two mainline vehicles rather than to allow only one on-ramp vehicle, which can be clearly demonstrated by plotting individual trajectories.

As shown in Fig. 3.7 (a), the second and third on-ramp vehicles choose to merge in between the second and the third mainline vehicles when the initial positions of these two on-ramp vehicles are exactly between these two mainline vehicles, and there is not much difference in all vehicles' velocities. The third mainline vehicle carries out a larger slowdown to generate a larger gap. However, as illustrated

by Fig. 8 (a), the third on-ramp vehicle merges behind the third mainline vehicle when the initial speed of the third on-ramp vehicle is low. This is reasonable because when the initial speed of a merging vehicle is close to mainline vehicles' speed, if a far upstream mainline vehicle is selected as a facilitating vehicle, the on-ramp merging vehicle will have to wage an extra deceleration and then acceleration in order to wait for the arrival of the mainline facilitating vehicle. Thus, nearby mainline vehicles should be chosen. Conversely, when the initial speed of a merging vehicle is low, choosing a nearby mainline vehicle to generate a gap would cause significant speed drop of facilitating and its following vehicles. Consequently, a relatively more upstream mainline vehicle should be selected, and the on-ramp vehicle could accelerate first to mitigate this negative effect. Chen et al. (2021)'s results drew similar conclusions: When the on-ramp vehicle's speed is low, it would merge to the back of mainline platoon. When the on-ramp vehicle's speed gets close to mainline vehicles' speeds, the mainline vehicle close to the initial position of the merging vehicle is selected. Speed profiles, shown in Fig. 7 (b) and 8 (b), are parabolic or linear, because we describe the trajectory by cubic polynomial functions. Similarly, acceleration trajectories, shown in Fig. 7 (c) and 8 (c), are linear or constant. The upper and lower limits of the state variable velocity and the control input acceleration are always obeyed at any time instant.

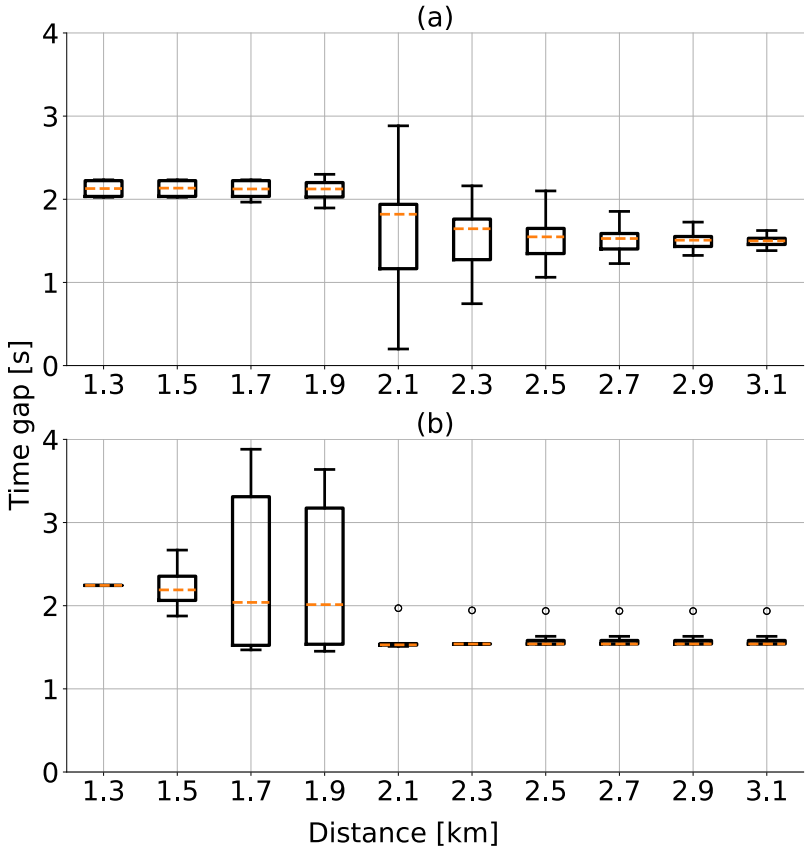


Figure 3.6. Time gap distribution in the mainline

(a) with SUMO default control; (b) with the proposed approach.

(5) The effect of the Length of the Cooperation Zone

We are interested in the impact of different lengths of the cooperative zone on ramp merging. Thus, we test the safety and traffic efficiency of our proposed model and the baseline model under a 350-meter-long, 325-meter-long, and 300-meter-long zone (including the acceleration lane), respectively. Table 3.6 summarizes the average time gap between merging and facilitating vehicles, and the total travel delay, resulting from the proposed approach and VROCP, under different lengths of the

cooperation zone.

From Table 3.6, we see that the baseline method, i.e., VROCP, achieves desired time gap (no less

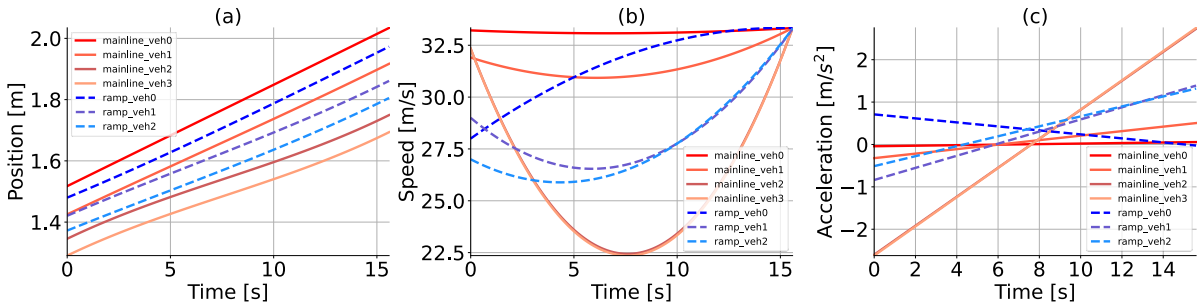


Figure 3.7. Vehicle trajectories for the high on-ramp vehicle's speed scenario

(a) positions; (b) speeds; (c) acceleration.

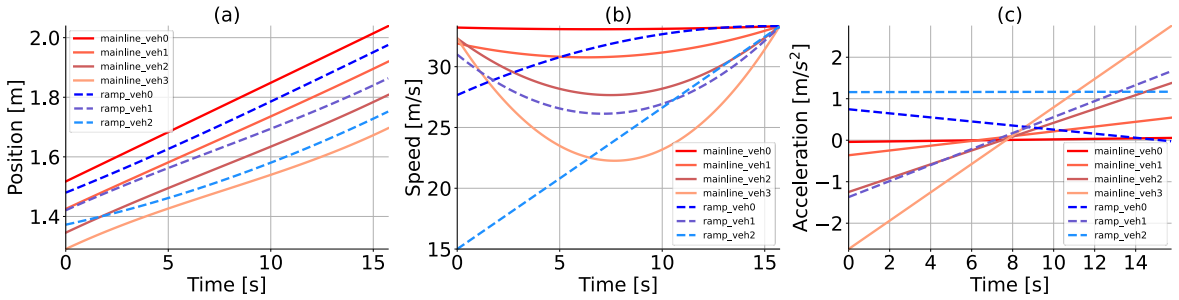


Figure 3.8. Vehicle trajectories for the low on-ramp vehicle's speed scenario

(a) positions; (b) speeds; (c) acceleration.

than 1.5 seconds) when the cooperation zone is sufficiently long (350m). However, when the zone reduces to 325m, VROCP cannot guarantee the safe time gap anymore and significantly increases travel delay. Moreover, VROCP would lead to accidents when the zone further reduces to 300m. In contrast, the proposed approach still ensures the target time gap even when the zone is so short that the baseline fails. Inevitably, travel delay would increase as zone shortens.

Table 3.6. Safety and delay comparison with different cooperation ranges

Cooperation Zone [m]	Proposed		VROCP	
	$\bar{\tau}[s]$	$D[s]$	$\bar{\tau}[s]$	$D[s]$
350	1.50	1186.95	1.66	1311.78
325	1.50	1292.37	1.47	1714.62
300	1.50	1998.20	Failure	Failure

Note: (1) $\bar{\tau}$ refers to mean time gap; (2) D refers to total travel delay.

The reason why the proposed approach can ensure safety in the limited ramp length case is that we formulate the constant time headway rule for terminal spacing between vehicles as hard constraints, while the baseline method puts the terminal spacing between on-ramp vehicles and facilitating vehicles in the objective function, which leads to unsafe solution in restricted scenarios. Furthermore, when the cooperation zone is short, the cooperation pair chosen by FIFO cannot generate enough space for merging, but in contrast, the proposed approach can choose an upstream mainline vehicle to generate a feasible gap in advance albeit at the cost of on-ramp vehicles' delay.

(6) The Effect of Weight Factors

The effect of weighting factors, i.e., w_m and w_r, on travel delay is investigated. Five combinations of (w_m, w_r) are tested: (0.1, 0.9), (0.3, 0.7), (0.5, 0.5), (0.7, 0.3) and (0.9, 0.1). A total of 300 mainline and 150 on-ramp CAVs enter the simulation within 15 minutes.

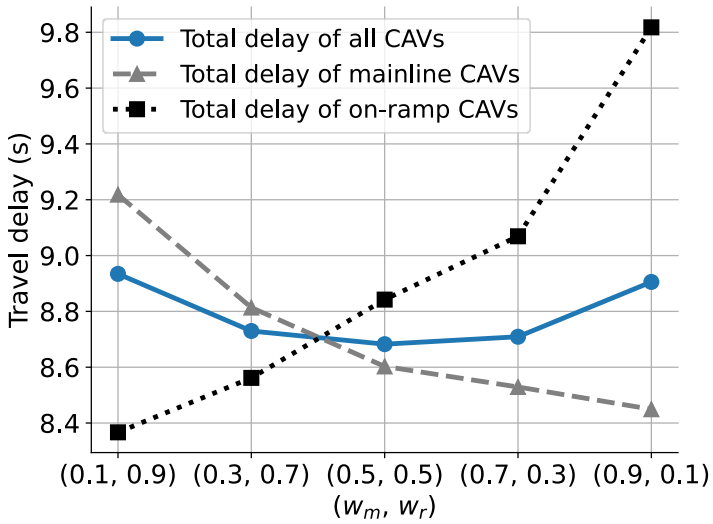


Figure 3.9. Travel delay under different combinations of (w_m, w_r)

As shown in Fig. 3.9, increasing w_m can reduce the delay of mainline CAVs. Similarly, increasing w_r reduces the delay of on-ramp CAVs. In addition, when w_m is equal to w_r, the objective is to minimize the delay of all CAVs, and therefore the minimum total delay of all CAVs is obtained under the case.

(7) Computational Efficiency

The computation time of different numbers of mainline vehicles and on-ramp vehicles is tested. The tested numbers of mainline vehicles are 1, 5, 10, 15, and 20, and the tested numbers of on-ramp vehicles are 1, 2, and 3.

As a result, Fig. 3.10 shows that with 20 mainline vehicles and 1 on-ramp vehicle, the computation time is 0.63 seconds. With 20 mainline vehicles and 2 on-ramp vehicles, the computation time increases to 1.1 seconds. Similarly, with 20 mainline vehicles and 3 on-ramp vehicles, the computation time increases further to 1.8 seconds.

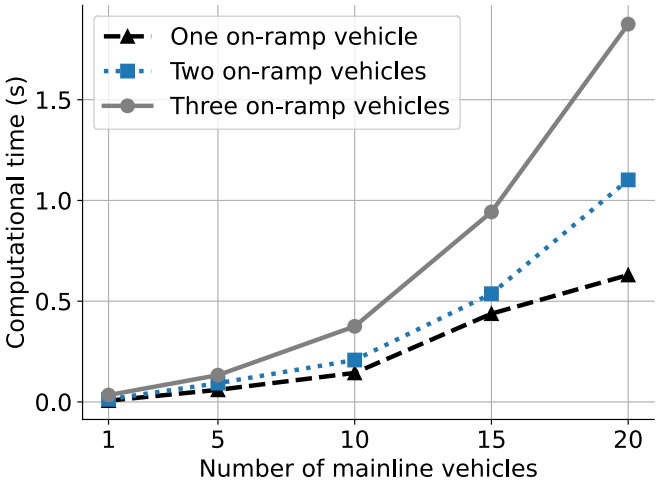


Figure 3.10. Computation time of various mainline and on-ramp CAVs number

We can see that the proposed method can handle a large number of mainline vehicles within 1 second, but the computation time increases quickly as the number of on-ramp vehicles increases. This result is consistent with the analysis of the search space of the vehicle sequences in Section III.B. That is, the search space is $(|I_m| + 1)^{|I_r|}$. As the number of on-ramp vehicles increases, the size of the search space grows exponentially. This means that increasing the number of vehicles on the ramp will have a greater impact on the computation time. Nevertheless, in most practical ramp merge scenarios, the number of ramp vehicles is usually less than the number of mainline vehicles.

(8) Real-world Data Validation

The proposed method is validated using real-world data from the Next Generation Simulation (NGSIM) Open Data, specifically the Interstate 80 (I-80) Freeway Dataset (U.S. (2016)) , which contains ramp merge scenarios. We extract both on-ramp vehicles and the rightmost mainline vehicles from the I-80 dataset, resulting in a total of 850 vehicles observed over a 45-minute period. These vehicles enter the simulation with the initial position and speed provided by the dataset and then are guided through the cooperative merging zone under the control of the following different methods: the proposed approach and the three alternative methods for comparison, namely, MCTS-DA, VROCP, and FTOCP. We compare the performance of the proposed approach with these alternative methods in terms of travel delay and total computation time.

Table 3.7. Performance comparison on real dataset.

	Proposed	Enumeration	FIFO-OCP	FT-OCP
$D\ (s)$	469	469	562	666
$D_m\ (s)$	285	285	411	456
$C\ (s)$	13.9	441	5.9	3.9

Note: (1) D refers to total delay of all vehicles; (2) D_m refers to total delay of mainline vehicles; (3) C refers to total computation time.

As shown in Table 3.7, the proposed approach outperforms the three alternatives. Specifically, the proposed approach achieves the same total travel delay as MCTS-DA, which can also obtain the time-optimal vehicle sequence by enumerating all vehicle sequences and repeatedly solving a MILP model

to obtain the travel time of vehicles. However, the proposed approach requires way less computation time than MCTS-DA thanks to the reduced search space of the proposed method. The other two alternatives, VROCP and FTOCP, spend less total computation time, but their travel delays are larger because they do not allow for flexible merging sequence and merging time, respectively. In addition, the lower total delay of mainline vehicles shows that the better merging sequence mitigates the impact of on-ramp vehicles on mainline traffic. In the 45-minute traffic simulation, the accumulated total computation time differences between the proposed method and these two alternative methods are 8 seconds and 10 seconds respectively, but during this time, each method is triggered and solved for many times, and the single computation time difference is only at the magnitude of 0.1 seconds on average.

(9) Impact of Human-driven Vehicles on CAVs

We use a real lane-changing trajectory of a human-driven vehicle (HDV), followed by two mainline CAVs and an on-ramp CAV. Every 0.5 seconds, the model would re-plan these three CAVs based on the updated information of the human-driven vehicle. At each replanning step, the future speed of the human-driven vehicle is assumed to remain its current speed, a typical treatment in relevant studies. The below results show that the proposed approach successfully fulfilled the merging task under the influence of an uncontrolled, perturbing HDV.

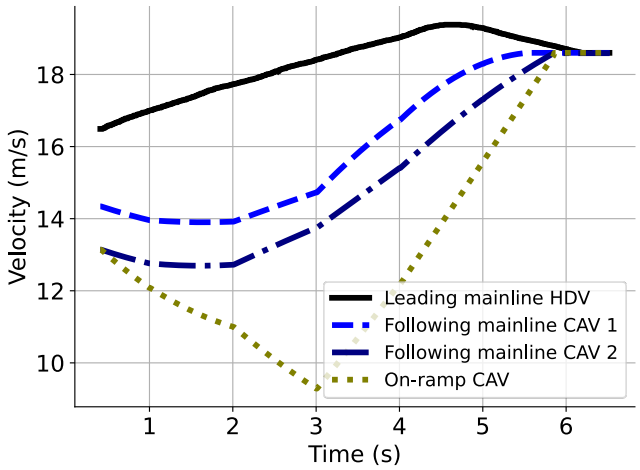


Figure 3.11. Merging speed profile for the unexpected lane change

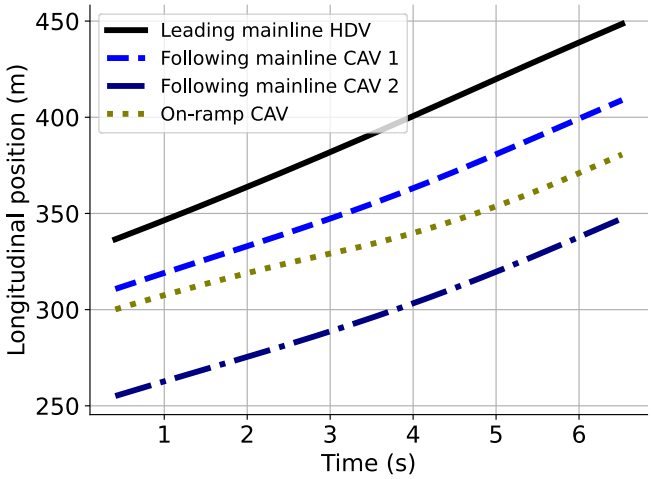


Figure 3.12. Merging position profile for the unexpected lane change

As depicted in Fig. 3.11, the human driver initiates the lane change at a higher speed than the vehicles in the target lane and then gradually reduces the speed back to normal. The following mainline CAV first reduces speed slightly and then gradually accelerates to achieve the desired spacing with the leading human-driven vehicle. Similarly, the merging CAV also undergoes a noticeable deceleration to achieve sufficient merging distances (33m) to the leading and following mainline CAVs, respectively. Eventually, the speeds of all the CAVs reach the steady-state speed of the leading human-driven vehicle.

3.5 Summary

In this chapter, we model the task of cooperative merging of an on-ramp CAV stream and a mainline CAV stream at a freeway on-ramp merging section as a mixed integer nonlinear programming problem that guarantees safety and enjoys high-quality vehicle trajectories. The total travel delay is minimized by simultaneously optimizing the merging sequence and the continuous-time vehicle trajectories. Such a treatment can avoid generating a merging sequence that would result in infeasible or low-quality trajectories. Moreover, the merging positions and time are outcomes of the optimization model rather than heuristically pre-defined. In addition, trajectories are described in continuous-time form so that safety is guaranteed at any time.

To efficiently solve the proposed MINLP model, on-ramp merging vehicles' optimal merge-in gaps are determined one by one. This sequential search process was built based on a necessary condition of optimality of the proposed MINLP model which we identified and proved. Therefore, the sequential search process can generate the true optimal merging sequence, that is, the one that is obtained by considering all the on-ramp merging vehicles together. Thanks to the sequential feature, the search space is significantly reduced. Subproblems are NLP and are efficiently solved by the iterative LP method to generate planned trajectories.

The traffic efficiency, safety and computational efficiency of the proposed approach are demonstrated under different traffic conditions and compared with three alternative methods on the NGSIM dataset. In addition, the impact of the length of the cooperative zone and the weighting factors on the traffic efficiency and safety is investigated. The computational efficiency is also evaluated. Furthermore, the proposed method is implemented in a feedback loop to complete the cooperative merging task under a real HDV trajectory.

Chapter 4

Decomposition of Vehicle Scheduling and Trajectory Planning

This chapter formulates the problem of merging on-ramp CAV flows into multiple mainline CAV flows as an MINLP model, which considers spatiotemporal relationships between vehicles, vehicle kinematics, and road geometry, with the aim of maximizing traffic efficiency, avoiding unnecessary lane changes, and generating the smoothest vehicle trajectories. there is a notable absence of fast solution algorithms for the joint decision-making concerning lane changes, vehicle sequences, and trajectories. The computational challenges posed by complex mathematical models, coupled with stringent real-time requirements, lead many studies to rely on heuristic rules or simplified surrogate motion planning for lane-changing and sequencing decisions. These approaches often yield suboptimal solutions (e.g., Hu and Sun, 2019). Additionally, enumeration methods, though commonly employed in small-scale multi-vehicle games, inevitably lead to significant computational delays (e.g., 5Hang et al., 2021; Yang et al., 2023).To efficiently solve the proposed MINLP model, we develop a decomposition-based solution algorithm to decouple the integer programming model and nonlinear programming model. This chapter is organized as follows. Section 4.1 formulates the problem of cooperative lane changes and merging of mainline and an on-ramp traffic streams at a multi-lane freeway on-ramp merging section as an MINLP model. Sections 4.2 develops a computationally efficient decomposition-based algorithm to solve the model. Section 4.3 presents the simulation setup and validates the proposed method through numerical experiments. Section 4.4 concludes this study.

4.1 Model Formulation

Section 4.1 first introduces the problem background, i.e., the merging scenario. Before formulating a mixed-integer nonlinear programming (MINLP) model for the cooperative merging problem at multilane freeway merging sections in Section 4.4, Sections 4.2, and 4.3 give mathematical descriptions for vehicle scheduling, and vehicle trajectories used in the model formulation, respectively.

(1) Multilane Freeway Merging Scenario

Figure 4.1 illustrates a typical freeway on-ramp merging section that consists of two mainline lanes (an inside lane and an outside lane) and an on-ramp lane connected with an acceleration lane. Multiple vehicles drive on the two mainline lanes, and several on-ramp vehicles intend to merge into the freeway. As shown in Figure 2, the merging section also includes a roadside unit (RSU) and a trigger point. Specifically, the trigger point is located on the on-ramp lane to trigger the controller (i.e., the proposed optimization model) when an on-ramp vehicle arrives; the RSU is placed upstream of the intersection between the outside mainline lane and the on-ramp lane to collect vehicle information, execute specific controllers, and send commands to vehicles. Before any vehicle passes the trigger point, all vehicles operate in a car-following mode. When an on-ramp vehicle passes the trigger point, the RSU initiates a

control cycle. Within this control cycle, the on-ramp vehicle that passes the trigger point, along with the following on-ramp vehicles and nearby mainline vehicles are grouped together to form the current planned connected automated vehicle (CAV) group. These CAVs in the current control cycle no longer operate in the car-following mode but follow the recommended controller instructions to execute the lane-changing and merging manoeuvres and planned trajectories. After completing commands, these CAVs revert to the car-following mode. Here, note that to avoid the risk associated with both lane-changing and merging manoeuvres occurring in the same area, these two types of manoeuvres are executed separately in their respective lane-changing and merging area. In order for CAVs to obey this separate execution rule, the corresponding mathematical expression is incorporated into our proposed model.

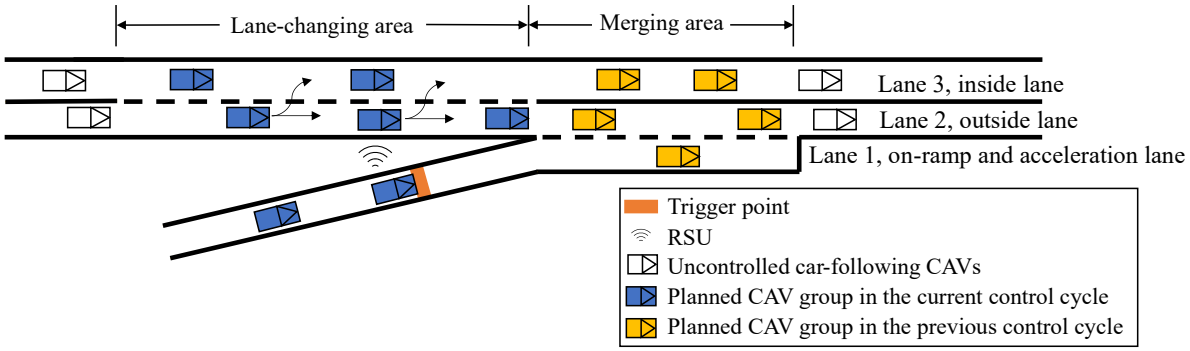


Figure 4.1. The multi-lane freeway on-ramp merging scenario.

Two assumptions used in this study are summarized as follows. The first assumption is that this study allows for a fully CAV environment, which is widely used in the existing literature (e.g., Yang et al., 2022; Chen et al., 2023). In this case, all CAVs within a group share kinematic information and are fully controllable. The second assumption is that the aforementioned lane-changing area adopts a one-side lane-changing prohibition rule, which allows CAVs in the outside lane to change lanes to the inside lane once, while prohibiting CAVs in the inside lane from changing lanes to the outside lane. This assumption is also widely used in literature, such as Hu and Sun (2019), and can help increase the safety of the entire system.

(2) Mathematical Description for Vehicle Sequences and Lane Changes

As depicted in Figure 4.2, let $L := \{1, 2, 3\}$ denote the set of lanes, where indices 1, 2, and 3 correspond to the on-ramp, outside lane, and inside lane, respectively. Let I_l represent the set of planned vehicles initially in lane l ($l \in L$) within a control cycle, which implies that lane l initially has $|I_l|$ planned vehicles within the control cycle. In addition, two dummy vehicles are conceptually positioned after the last planned vehicles in outside and inside lanes, respectively, indexed as $|I_2| + 1$ and $|I_3| + 1$. It should be noted that dummy vehicles are just for model formulation. We therefore have an extended set $I_l^+ := I_l \cup \{|I_l| + 1\}, l \in \{2, 3\}$.

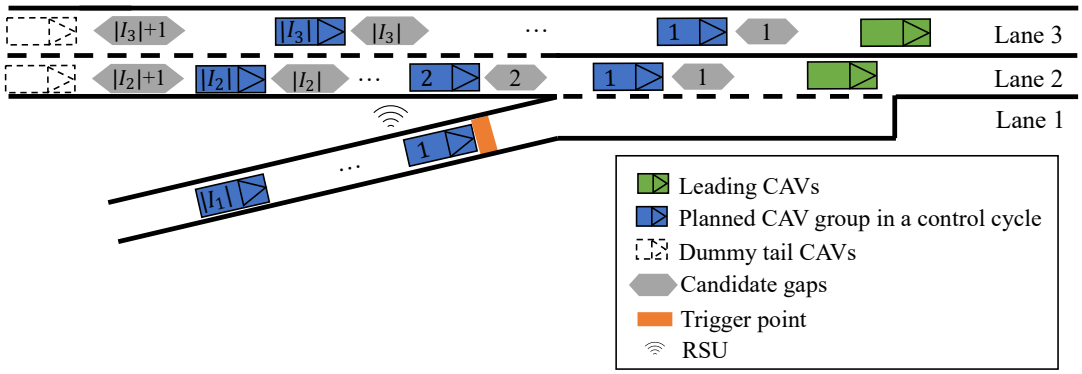


Figure 4.2. Schematic illustration of vehicles and candidate gaps in a control cycle.

Lane l, $l \in \{2, 3\}$, has $|I_l|$ planned vehicles and a dummy vehicle, resulting in a total of $|I_l| + 1$ candidate gaps in front of $|I_l| + 1$ vehicles. Hence, selecting a gap from candidate gaps to merge or change lanes can be regarded as selecting a facilitating vehicle from $|I_l| + 1$ vehicles. Finally, the current control cycle also includes two leading vehicles whose trajectories are planned in the last control cycle, each located in front of the first planned vehicles in their respective lanes. Since these two leading vehicles are already planned in the last control cycle, they are not planned again in the current control cycle.

To establish vehicle sequences driving through the end of the freeway merging area, each on-ramp vehicle must select one outside-lane vehicle (including the dummy vehicle) acting as its facilitating vehicle. Similarly, outside-lane lane-changing vehicles must also select inside-lane vehicles (including dummy one) as facilitating vehicles. Here, note that choosing the dummy vehicle is intended to follow the last outside-lane (or inside-lane) vehicles. Also, a facilitating vehicle can help multiple vehicles to cut in. Let $\alpha_{i,j}$ represent a binary variable, which equals 1 if and only if on-ramp vehicle i, $i \in I_1$, chooses the j^{th} outside-lane vehicle, $j \in I_2^+$, as its facilitating vehicle for merging, and equals 0 otherwise. Similarly, $\beta_{i.j}$ is defined as a binary variable, which equals 1 if and only if outside-lane vehicle i, $i \in I_2$, chooses the j^{th} inside-lane vehicle, $j \in I_3^+$, as its facilitating vehicle for a lane change. In addition, γ_i is defined as a binary variable, which equals 1 if and only if outside-lane vehicle i, $i \in I_2$, changes to the inside lane, and equals 0 otherwise. That is, γ_i represents the lane change decision for outside-lane vehicle i, $i \in I_2$. It should be noted that an outside-lane vehicle can simultaneously act as a facilitating vehicle and a lane-changing vehicle, which means that the corresponding $\alpha_{i,j}$ and $\beta_{i.j}$ can equal 1 at the same time. Lastly, let $\tau_{l,i}$ denote the terminal time when the vehicle i, $i \in I_l$, initially in lane l, $l \in L$, passes the end of the merging area. Terminal time ($\tau_{l,i}$) implies the passing sequence of each vehicle. Terminal time difference between two consecutive vehicles in the same lane must keep at least a minimum time headway denoted by h seconds whose value can be calculated by:

$$h = t_{\text{gap}} + \frac{l_{\text{veh}} + s_0}{v_{\text{f}}}, \tag{4.1}$$

where t_{gap}, l_{veh}, s_0, and v_{f} represent a specified minimum time gap, a vehicle length, a standstill distance, and a free-flow speed, respectively.

(3) Mathematical description for vehicle trajectories

Let s_{lc}, s_{m}, and S represent the end of cooperative lane-changing area, the end of cooperative merging area, and the set of discrete points along mainline lanes ($S := \{1, \dots, s_{\text{lc}}, \dots, s_{\text{m}}\}$), respectively.

The interval distance between two consecutive discrete points is denoted by Δ. Following the idea of planning in the space domain (Thomas and Boyd, 2014; Karlsson et al., 2020), vehicle trajectories are typically described with respect to discrete point $s \in S$ (i.e., s is regarded as an independent variable). Therefore, in this study, for vehicle i initially in lane l ($l \in L, i \in I_l$), its trajectory consists of the control input variable and state variables. Specifically, the control input variable is the change of acceleration in space coordinates, represented by $b_{l,i}(s)$; the state variables record the profiles of time, kinetic energy per unit mass, and acceleration at different positions, defined as state variables $t_{l,i}(s)$, $E_{l,i}(s)$, and $a_{l,i}(s)$, respectively. Then, the longitudinal kinematic equations are formulated as Equations (4.2)–(4.4), whose detailed derivation can be found in Hamednia et al. (2022). In the following sections of this study, we use $b_{l,i,s}$, $a_{l,i,s}$, $E_{l,i,s}$, and $t_{l,i,s}$ ($l \in L, i \in I_l, s \in S$) to represent the jerk, acceleration, kinematic energy, and time profiles of vehicle i initially in lane l at position point s.

$$\frac{dt_{l,i}(s)}{ds} = \frac{1}{v_{l,i}(s)} = \frac{1}{\sqrt{2E_{l,i}(s)}} \tag{4.2}$$

$$\frac{dE_{l,i}(s)}{ds} = a_{l,i}(s) \tag{4.3}$$

$$\frac{da_{l,i}(s)}{ds} = b_{l,i}(s). \tag{4.4}$$

This study adopts the finite difference equations of discrete Equations (4.2), (4.3), and (4.4). This treatment would lead numeric error. However, this way helps transform an intractable functional optimization problem to a tractable numeric optimization problem. Moreover, Equation (4.2) requires that the velocity must be greater than zero, so the proposed model cannot be used for full stop cases.

(4) Mathematical Model

Before introducing the proposed MINLP model, notation used in this study is first summarized as follows.

Sets and indices:

L set of all lanes, $l \in L = \{1,2,3\}$, where indices 1, 2, and 3 refer to the on-ramp lane, outside lane, and inside lane, respectively.

I_l set of planned vehicles initially in lane l ($l \in L$), i (j) $\in I_l$.

I_l^+ extended set of planned and dummy vehicles initially in lane l ($l \in L\backslash\{1\}$), $i \in I_l^+$.

S set of discrete points along mainline lanes, $s \in S = \{1, \cdots, s_{\mathrm{lc}}, \cdots, s_{\mathrm{m}}\}$.

Parameters:

$a_{l,i}^{\mathrm{init}}$ acceleration of vehicle i initially in lane l at its initial position.

$\underline{a}$, $\overline{a}$ minimum and maximum acceleration values.

$\underline{b}$, $\overline{b}$ minimum and maximum jerk values.

$\underline{E}$, $\overline{E}$ minimum and maximum kinetic energy values; $\overline{E} = \frac{1}{2}v_{\mathrm{f}}^2$.

h minimum time headway between two consecutive vehicles in a lane.

M big-M constant parameter.

$p_{l,i}$ initial position of vehicle i initially in lane l; $p_{l,i} \in S\backslash\{s_{\mathrm{lc}}, \cdots, s_{\mathrm{m}}\}$.

s_{m} end of the merging section.

s_{lc} end of the lane-changing section.

$\hat{t}_l$ time that a leading vehicle of the planned CAV group in mainline lane l passes the end of merging area, $l \in L\backslash\{1\}$; a leading vehicle is referred to as the vehicle preceding the most downstream vehicle in current planned CAV group.

$\underline{t}_{l,i}, \bar{t}_{l,i}$ lower and upper bounds of the time that vehicle i in lane l passes the end of merging area, $l \in L, i \in I_l$.

v_{f} free flow speed.

$v_{l,i}^{\mathrm{init}}$ velocity of vehicle i initially in lane l at its initial position.

w_{lc} weighting factor for lane changes.

$w_{\mathrm{a}}, w_{\mathrm{b}}$ weighting factors for the cost of acceleration and jerk.

w_{E} weighting factor for difference from the maximum kinetic energy value.

Δ distance interval between two discrete consecutive points.

Decision variables:

Sequencing and lane-changing related decision variables:

$\tau_{l,i}$ continuous, terminal time that vehicle i in lane l passes the end of the merging section, $l \in L, i \in I_l$.

$\alpha_{i,j}$ binary, equals 1 if and only if on-ramp vehicle i chooses outside-lane vehicle j as its facilitating vehicle for merging, 0 otherwise; $i \in I_1, j \in I_2^+$.

$\beta_{i,j}$ binary, equals 1 if and only if outside-lane vehicle i chooses inside-lane vehicle j as its facilitating vehicle for a lane change, 0 otherwise; $i \in I_2, j \in I_3^+$.

γ_i binary, equals 1 if and only if outside-lane vehicle i decide to change to the inside lane, 0 otherwise; $i \in I_2$.

Ω binary, equals 1 if and only if all outside-lane vehicles in the control cycle decide to change to the inside lane; 0 otherwise.

Trajectory related decision variables:

$a_{l,i,s}$ continuous, acceleration that vehicle i in lane l passes the s^{th} point, $l \in L, i \in I_l, s \in S$.

$b_{l,i,s}$ continuous, jerk that vehicle i in lane l passes the s^{th} point. $l \in L, i \in I_l, s \in S$.

$E_{l,i,s}$ continuous, kinetic energy per unit mass that vehicle i in lane l passes the s^{th} point, $l \in L, i \in I_l, s \in S$.

$t_{l,i,s}$ continuous, time that vehicle i in lane l passes the s^{th} point, $l \in L, i \in I_l, s \in S$.

According to the notation introduced, an MINLP model [MVO] is formulated, which incorporates both vehicle scheduling and trajectory planning to achieve optimality in traffic efficiency and trajectory metrics at a multi-lane merging section.

[MVO]

$$\min \sum_{l\in L}\sum_{i\in I_l}\tau_{l,i} + w_{\mathrm{lc}}\sum_{i\in I_2}\gamma_i + \sum_{l\in L}\sum_{i\in I_l}\sum_{s\in\{p_{l,i},\cdots,s_{\mathrm{m}}\}}\left[w_{\mathrm{a}}a_{l,i,s}^2 + w_{\mathrm{b}}b_{l,i,s}^2 + w_{\mathrm{E}}\left(\bar{E}-E_{l,i,s}\right)^2\right] \quad (4.5)$$

subject to

$$\underline{t}_{l,i} \le \tau_{l,i} \le \bar{t}_{l,i} \qquad \forall l\in L, i\in I_l \quad (4.6)$$

$$\tau_{l,i+1} - \tau_{l,i} - h \ge 0 \qquad \forall l\in\{1,3\}, i\in I_l\backslash\{|I_l|\} \quad (4.7)$$

$$\left|\gamma_i-\gamma_j\right|M + \left(\tau_{2,i}-\tau_{2,j}-h\right) \ge 0 \qquad \forall i\in I_2\backslash\{1\}, j\in\{1,\cdots,i-1\} \quad (4.8)$$

$$\tau_{1,1} - \hat{t}_2 - h \ge 0 \quad (4.9)$$

$$\tau_{3,1} - \hat{t}_3 - h \ge 0 \quad (4.10)$$

$$\tau_{2,i} - [(1-\gamma_i)\hat{t}_2 + \gamma_i\hat{t}_3\,] - h \ge 0 \qquad \forall i\in I_2 \quad (4.11)$$

$$\gamma_i M + \tau_{2,i} - \sum_{j=2}^{|I_3|+1}\beta_{i,j}\tau_{3,j-1} - h \ge 0 \qquad \forall i\in I_2 \quad (4.12)$$

$$\gamma_i M + \left(\sum_{j\in I_3}\beta_{i,j}\tau_{3,j} - \tau_{2,i} - h\right)|I_3| \ge 0 \qquad \forall i\in I_2 \quad (4.13)$$

$$\sum_{j\in I_3^+}\beta_{i,j} = \gamma_i \qquad \forall i\in I_2 \quad (4.14)$$

$$\sum_{m=1}^{j}\alpha_{i,m} \ge \alpha_{i+1,j} \qquad \forall i\in I_1\backslash\{|I_1|\}, j\in I_2^+ \quad (4.15)$$

$$\left(\alpha_{i,j}-\alpha_{i,j+1}\right)\gamma_j = 0 \qquad \forall i\in I_1, j\in I_2\backslash\{|I_2|\} \quad (4.16)$$

$$\sum_{j\in I_2^+}\alpha_{i,j} = 1 + \sum_{j\in I_2}\gamma_j\alpha_{i,j} \qquad \forall i\in I_1 \quad (4.17)$$

$$\tau_{1,i} - \sum_{j=2}^{|I_2|+1}\alpha_{i,j}\tau_{2,j-1}\left(1-\gamma_{j-1}\right) - h \ge 0 \qquad \forall i\in I_1 \quad (4.18)$$

$$\sum_{j\in I_2}\alpha_{i,j}\tau_{2,j}\left(1-\gamma_j\right) - \tau_{1,i} - h + \Omega M \ge 0 \qquad \forall i\in I_1 \quad (4.19)$$

$$\Omega = \left\lfloor\frac{\sum_{j\in I_2}\gamma_j}{|I_2|}\right\rfloor \quad (4.20)$$

$$a_{l,i,s+1} = a_{l,i,s} + b_{l,i,s}\cdot\Delta \qquad \forall l\in L, i\in I_l, s\in\{p_{l,i},\cdots,s_{\mathrm{m}}-1\} \quad (4.21)$$

$$E_{l,i,s+1} = E_{l,i,s} + a_{l,i,s}\cdot\Delta \qquad \forall l\in L, i\in I_l, s\in\{p_{l,i},\cdots,s_{\mathrm{m}}-1\} \quad (4.22)$$

$$t_{l,i,s+1} = t_{l,i,s} + \frac{1}{\sqrt{2E_{l,i,s}}}\cdot\Delta \qquad \forall l\in L, i\in I_l, s\in\{p_{l,i},\cdots,s_{\mathrm{m}}-1\} \quad (4.23)$$

$$a_{l,i,p_{l,i}} = a_{l,i}^{\mathrm{init}} \qquad \forall l\in L, i\in I_l \quad (4.24)$$

$$E_{l,i,p_{l,i}} = \frac{1}{2}(v_{l,i}^{\mathrm{init}})^2 \qquad \forall l\in L, i\in I_l \quad (4.25)$$

$$t_{l,i,p_{l,i}} = 0 \qquad \forall l\in L, i\in I_l \quad (4.26)$$

$$t_{l,i,s_{\mathrm{m}}} = \tau_{l,i} \qquad \forall l\in L, i\in I_l \quad (4.27)$$

$$t_{3,j-1,s} - t_{2,i,s} - h + M\cdot(1-\beta_{i,j}) \ge 0 \qquad \forall i\in I_2, j\in I_3\backslash\{1\}, s\in\{s_{\mathrm{lc}},\cdots,s_{\mathrm{m}}\} \quad (4.28)$$

$$t_{2,i,s} - t_{3,j,s} - h + M\cdot(1-\beta_{i,j}) \ge 0 \qquad \forall\, i\in I_2, j\in I_3, s\in\{s_{\mathrm{lc}},\cdots,s_{\mathrm{m}}\} \quad (4.29)$$

$$t_{l,i,s} - t_{l,i-1,s} - h \ge 0 \qquad \forall l\in\{1,3\}, i\in I_l\backslash\{1\}, s\in\{p_{l,i},\dots,s_m\} \quad (4.30)$$

$$t_{2,i,s} - t_{2,i-1,s} - h \ge 0 \qquad \forall i\in I_2\backslash\{1\}, s\in\{p_{2,i},\cdots,s_{\mathrm{lc}}\} \quad (4.31)$$

$$|\gamma_i-\gamma_j|M + t_{2,i,s} - t_{2,j,s} - h \ge 0 \qquad \forall i\in I_2\backslash\{1\}, j\in\{1,\cdots,i-1\}\ s\in\{s_{\mathrm{lc}},\cdots,s_{\mathrm{m}}\} \quad (4.32)$$

$$\underline{b} \le b_{l,i,s} \le \overline{b} \qquad \forall l \in L, i \in I_l, s \in \{p_{l,i}, \cdots, s_m\} \tag{4.33}$$

$$\underline{a} \le a_{l,i,s} \le \overline{a} \qquad \forall l \in L, i \in I_l\,, s \in \{p_{l,i}, \cdots, s_m\} \tag{4.34}$$

$$\underline{E} \le E_{l,i,s} \le \bar{E} \qquad \forall l \in L, i \in I_l, s \in \{p_{l,i}, \cdots, s_m\} \tag{4.35}$$

$$t_{l,i,s} \ge 0 \qquad \forall l \in L, i \in I_l, s \in \{p_{l,i}, \cdots, s_m\} \tag{4.36}$$

$$\tau_{l,i} \ge 0 \qquad \forall l \in L, i \in I_l \tag{4.37}$$

$$\alpha_{i,j} \in \{0,1\} \qquad \forall\, i \in I_1, j \in I_2 \tag{4.38}$$

$$\beta_{i,j} \in \{0,1\} \qquad \forall\, i \in I_2, j \in I_3 \tag{4.39}$$

$$\gamma_i \in \{0,1\} \qquad \forall i \in I_2 \tag{4.40}$$

$$\Omega \in \{0,1\}. \tag{4.41}$$

Objective: The objective function (4.5) aims to optimize both traffic performance and vehicle trajectory metrics by minimizing the sum of three terms. The first two terms (i.e., $\sum_{l\in L}\sum_{i\in I_l}\tau_{l,i}$ and $\sum_{i\in I_2}\gamma_i$) are traffic performance metrics including the total travel time and the number of lane-changing manoeuvres. The purpose of this setup is to minimize the total travel delay of CAVs and to avoid unnecessary lane changes. The third term contains two trajectory metrics, namely, the smoothness of acceleration and jerk, i.e., $\sum_{l\in L}\sum_{i\in I_l}\sum_{s\in S}(w_{\mathrm{a}}a_{l,i,s}^2 + w_{\mathrm{b}}b_{l,i,s}^2)$, and the kinetic energy difference relative to the maximum kinetic energy $\bar{E}$ at each position, i.e., $\sum_{l\in L}\sum_{i\in I_l}\sum_{s\in S} w_{\mathrm{E}}\left(\bar{E} - E_{l,i,s}\right)^2$. This kinetic energy-related objective helps make vehicle speeds close to the free flow speed.

Sequencing and lane-changing related constraints: To ensure the feasibility of vehicle sequences, lane *changes*, and terminal time of vehicles), constraints (4.6)–(4.20) and (4.37)–(4.41) incorporates several factors, including terminal time limits, terminal time relationships between vehicles, selection rules of facilitating vehicles, and connection between lane-changing and facilitating decisions.

(1) Terminal time constrained by velocity and acceleration limits: Constraints (4.6) set feasible terminal time limits, i.e., the minimum and maximum time required for each vehicle to pass the end of the merging section, considering *the* given speed, acceleration, and deceleration limits. The lower and upper bounds of $\tau_{l,i}$ can be calculated by Equations (4.42)–(4.43).

$$\underline{t}_{l,i} = \begin{cases} \dfrac{\sqrt{v_{l,i,0}^2 + 2\bar{a}d_{l,i,0}} - v_{l,i,0}}{\bar{a}}, & \text{if } \dfrac{v_{\mathrm{f}}^2 - v_{l,i,0}^2}{2\bar{a}} > d_{l,i,0} \\ \dfrac{v_{\mathrm{f}} - v_{l,i,0}}{\bar{a}} + \dfrac{d_{l,i,0} - \frac{v_{\mathrm{f}}^2 - v_{l,i,0}^2}{2\bar{a}}}{v_{\mathrm{f}}}, & \text{otherwise} \end{cases} \tag{4.42}$$

$$\bar{t}_{l,i} = \begin{cases} \dfrac{\sqrt{{v_{l,i,0}}^2 + 2\underline{a}d_{l,i,0}} - v_{l,i,0}}{\underline{a}}, & \text{if } \dfrac{\underline{v}^2 - v_{l,i,0}^2}{2\underline{a}} > d_{l,i,0} \\ \dfrac{\underline{v} - v_{l,i,0}}{\underline{a}} + \dfrac{d_{l,i,0} - \frac{\underline{v}^2 - {v_{l,i,0}}^2}{2\underline{a}}}{\underline{v}}, & \text{otherwise,} \end{cases} \tag{4.43}$$

where $\underline{a}$ and $\bar{a}$ represent the maximum deceleration and acceleration; $\underline{v}$ and $\mathrm{v_f}$ represent the minimum speed and maximum free flow speed; $v_{l,i,0}$ and $d_{l,i,0}$ represent the initial speed and initial distance of vehicle i in lane l to the end of merging section. The shortest time is calculated based on the vehicle accelerating to its maximum speed with the maximum acceleration and maintaining this speed throughout the cooperative section. Conversely, the longest time is determined by the vehicle decelerating to its minimum speed with the maximum deceleration and maintaining this speed

802 throughout the cooperative section.

803 *(2) Terminal time constraints related to CAVs in the same lane:* Constraints (4.7) state that two
804 consecutive on-ramp or inside-lane vehicles must maintain their initial vehicle sequences and keep at
805 least a minimum time headway between their terminal time. In other word, downstream on-ramp and
806 inside-lane vehicles must reach the end point of the merging section before upstream on-ramp and
807 inside-lane vehicles. However, since outside-lane vehicles can change lanes to overtake their preceding
808 vehicles, constraints (4.8) state that only if any two outside-lane vehicles choose the same lane do they
809 have to satisfy the requirements of initial sequences and minimum time headway. Moreover, constraints
810 (4.9) and (4.10) guarantee that the terminal time of on-ramp and inside-lane vehicles is respectively
811 limited by their mainline leading vehicles. In addition, which mainline leading vehicle limits the
812 terminal time of outside-lane vehicles depends on their lane-changing decisions, as stated in constraints
813 (4.11).

814 *(3) Terminal time constraints related to lane-changing and facilitating CAVs*: Any lane-changing
815 vehicle may be between two inside-lane vehicles after changing lanes, i.e., between a specified
816 facilitating vehicle and a preceding vehicle in the inside lane. This permutation determines their terminal
817 time relationship. Specifically, constraints (4.12) and (4.13) impose that the terminal time of a lane-
818 changing vehicle must maintain at least a minimum time headway from that of corresponding
819 facilitating and preceding vehicles, respectively. Here, note that the left side of constraints (4.13) needs
820 $|I_3|$. This is because if the total number of planned vehicles initially in lane 3 is zero, constraints (4.13)
821 need not to be considered, as in the case where there are no vehicles initially in lane 3, decision variable
822 τ related to lane 3 does not exist.

823 *(4) Constraints related to facilitating CAV selection for lane-changing CAVs:* Constraints (4.14)
824 ensure that when an outside-lane vehicle is assigned to change lanes, only one inside-lane vehicle can
825 be chosen as its facilitating vehicle. Conversely, if this outside-lane vehicle is not a lane-changing
826 vehicle, no inside-lane vehicle will be its facilitating vehicle.

827 *(5) Constraints related to facilitating CAV selection for merging CAVs:* Each on-ramp vehicle
828 needs to choose a facilitating vehicle in the outside lane. Constraints (4.15) state that an on-ramp vehicle
829 can only choose its facilitating vehicle from the group of outside-lane vehicles starting from the one
830 selected by the previous on-ramp vehicle and extending through all following outside-lane vehicles. It
831 should be noted that adjacent on-ramp vehicles can choose the same facilitating vehicle, i.e., it is
832 possible for a mainline vehicle to facilitate several on-ramp vehicles.

833 *(6) Connection between lane-changing and merging decisions:* Constraints (4.16) indicate that
834 when an outside-lane vehicle is both a facilitating vehicle and a lane-changing vehicle, the following
835 outside-lane vehicle also becomes a facilitating vehicle for on-ramp vehicles, thereby replacing the
836 previous facilitator. Therefore, the number of facilitating vehicles required for a merging vehicle
837 depends on whether the selected facilitating vehicles change lanes, as stated in constraints (4.17).

838 (*7) Terminal time constraints related to merging and facilitating CAVs:* Constraints (4.18) and
839 (4.19) define the minimum time headway requirement for each on-ramp vehicle with the two related
840 vehicles in the outside lane (facilitating and preceding vehicles). Moreover, these constraints account
841 for a particular case, i.e., if the related outside-lane vehicles are assigned to change lanes, the terminal
842 time constraints are no longer needed because they are in different lanes. Furthermore, the term ΩM in
843 constraints (4.19) is added to ensure that when all outside-lane vehicles change lanes, the constraints
844 still hold. The binary variable Ω indicates whether all outside-lane vehicles change lanes as constraint

(4.20).

(8) Range of variables: Constraints (4.37)–(4.41) state the range of schedule-related variables.

Trajectory-related constraints: To generate feasible vehicle trajectories (i.e., jerk, acceleration, kinematic energy, and time curves along position points), constraints (4.21)–(4.36) takes several factors into account, including the kinematic equations of vehicles, the time relationships between vehicles, the safety requirement for lane changes, the initial state of vehicles, and limits of vehicle capabilities.

(1) Constraints related to vehicle kinematics: Constraints (4.21)–(4.23) describe the discretised equations of vehicle kinematics, linking the state of vehicles (i.e., acceleration, kinematic energy, and time) at every position.

(2) Constraints related to initial and terminal conditions: Constraints (4.24)–(4.26) set the initial values of acceleration, kinematic energy, and time for each vehicle at its starting position, respectively. Constraints (4.27) state that the final point on the time curve and the terminal time are the same because both represent the moment a vehicle passes the end of the merging section.

(3) Constraints related to safe lane changes: Lane-changing maneuvers are executed at the end of the lane-changing section, namely, point s_{lc}. Minimum headway constraints are established to ensure safe lane changes. Specifically, constraints (4.28) and (4.29) impose a minimum time headway that lane-changing vehicles must maintain with their facilitating and preceding vehicles in the inside lane after lane changes.

(4) Constraints related to minimum time headway: Constraints (4.30) state that any two consecutive on-ramp or inside-lane vehicles must maintain at least a minimum time headway at every discrete point along their paths. Similarly, within the cooperative lane-changing area, all outside-lane vehicles must comply with the minimum headway rule, as stated in constraints (4.31). However, within the cooperative merging area, those outside-lane vehicles are distributed across two mainline lanes after changing lanes. Only those in the same lane must obey the minimum headway rule at each point.

(5) Bounded physical constraints: Constraints (4.33)–(4.36) restrict jerk, acceleration, energy, and time for all vehicles at each discrete position point, respectively, ensuring that these physical quantities remain within acceptable ranges throughout the motion of vehicles.

In summary, the objective function of the model [MVO] contains the total travel delay, the number of lane changes, and the smoothness of vehicle trajectories. Sequencing and lane-changing related constraints link the vehicles' terminal time with lane selection and sequence decisions. Trajectory-related constraints are also necessary to obey kinematic equations, bounded physical speed, and minimum time headway requirements between vehicles during whole movements.

4.2 Solution Algorithm

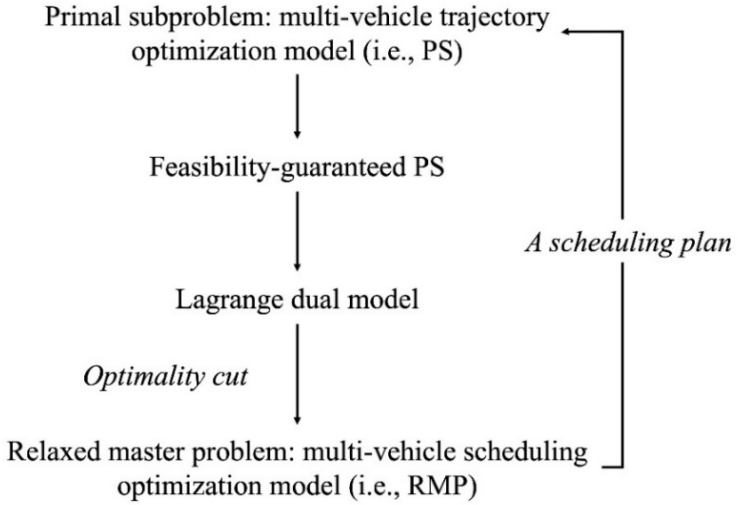
This study designs an efficient decomposition algorithm to deal with the MINLP nature of the proposed model [MVO]. Section 4.3.1 introduces the framework of the proposed Generalized Benders Decomposition-based solution algorithm. Sections 4.3.2, 4.3.3, and 4.3.4 introduce the decomposed primal subproblem, its feasible-guaranteed version, and the cuts yielded from the primal subproblem, respectively. Then, sections 4.3.5 and 4.3.6 introduce the relaxed master problem and its linearization, respectively. Finally, section 4.3.7 proves finite convergence property of the algorithm.

884 **(1) GBD-Based Solution Algorithm**

885 Generalized Benders Decomposition (GBD) is a partitioning framework for large-scale MINLP
886 problems (Geoffrion, 1972; Cai et al., 2001; Gao et al., 2023), which decomposes an MINLP model
887 into a relaxed master problem (RMP) and a primal subproblem (PS). An important step in GBD is to
888 select appropriate decision variables as complicating variables whose values are first obtained in RMP,
889 and the values of complicating variables are then passed into PS as given parameters for the PS solving
890 process. The RMP and PS are solved iteratively until the upper and lower bounds of the original problem
891 converge.

892 Utilizing the concept of GBD, the proposed MINLP model in this study can be divided into an
893 MILP multi-vehicle scheduling optimization model (serving as RMP) and a nonlinear programming
894 (NLP) multi-vehicle trajectory optimization model (serving as PS). Specifically, constraints in RMP are
895 sequencing and lane-changing related constraints, i.e. constraints (4.6)–(4.20) and (4.37)–(4.41). Some
896 of such constraints are nonlinear and thus linearized in Section 4.5. RMP includes all variables related
897 to sequencing and lane-changing ($\boldsymbol{\tau}$, $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, $\boldsymbol{\gamma}$, and $\boldsymbol{\Omega}$), with $\boldsymbol{\tau}$, $\boldsymbol{\gamma}$, and $\boldsymbol{\beta}$ specifically selected as
898 complicating variables that are also presented in PS. When the complicating variables in PS are
899 presented as fixed parameters, PS becomes a structure comprising linear constraints (i.e., constraints
900 (4.21)–(4.22), (4.24)–(4.36)) and non-convex constraints (4.23). All variables related to trajectories ($\boldsymbol{a}$,
901 $\boldsymbol{b}$, $\boldsymbol{E}$, and $\boldsymbol{t}$) are contained in PS.

902 RMP provides a scheduling plan (i.e., lane-changing decisions, facilitating vehicle choices, and
903 terminal time), and PS aims to find trajectories to fulfill the fixed scheduling plan. The feasibility-
904 guaranteed PS is further derived by introducing slack variables, which is introduced in Section 4.3.3.
905 Figure 4.3 illustrates the connection between RMP and PS. Specifically, PS is first transformed to the
906 feasibility-guaranteed PS, and then the so-called optimality cuts are formed through the Lagrange
907 function which is detailed in Section 4.3.4. In essence, the Lagrange function comprises the objective
908 function and the product of optimal Lagrange multipliers with the constraints, under the fixed
909 scheduling plan.



910

911 Figure 4.3. Schematic illustration of the connection between RMP and PS.

912 The procedure of the GBD-based solution algorithm is provided in Algorithm 1. The algorithm
913 iteratively solves RMP and PS until the lower and upper bounds of the objective function in the original
914 problem converge. Specifically, under a given scheduling plan, the optimality cuts from the feasibility-

guaranteed PS are added to RMP at each iteration. If the scheduling plan is feasible, the result of PS provides an upper bound for the original minimization model. The objective function value of RMP provides a lower bound for the original problem.

Table 4.1. GBD-based solution algorithm.

Algorithm 1. GBD-based solution algorithm

Set the lower and upper bounds, $\mathrm{LB}^{(0)} \leftarrow -\infty, \mathrm{UB}^{(0)} \leftarrow \infty$.
Set two tolerance thresholds, $\epsilon \leftarrow 1 \times 10^{-5}$, $\epsilon' \leftarrow 1 \times 10^{-4}$.
Obtain initial values of complicating variables in RMP according to heuristic rules, denoted by $\boldsymbol{y}^{(0)} := (\tau_{l,i}^{(0)}, \gamma_i^{(0)}, \beta_{l,i}^{(0)}),\ \ l \in L, i \in I_l, s \in S$ // $\boldsymbol{y}^{(0)}$ denotes the value set of complicating variables at beginning.
Set iteration index, $k \leftarrow 0$.
While $\mathrm{UB}^{(k)} - \mathrm{LB}^{(k)} \geq \epsilon$ **do**
  $\overline{\boldsymbol{y}} := \boldsymbol{y}^{(k)}$
  Solve the feasibility-guaranteed PS model under fixed complicating variables $\overline{\boldsymbol{y}}$.
  Obtain the value of slack variables, denoted by $\boldsymbol{\xi}^{(k)} \leftarrow (\dot{\xi}_{l,i}^{(k)}, \ddot{\xi}_{i,j,s}^{(k)}, \ddot{\xi}_{i,j,s}^{(k)}, \acute{\xi}_{i,j,s}^{(k)}), l \in L, i \in I_l, s \in S.$
  Obtain the trajectory solution, $\boldsymbol{x}^{(k)} \leftarrow (b_{l,i,s}^{(k)}, a_{l,i,s}^{(k)}, E_{l,i,s}^{(k)}, t_{l,i,s}^{(k)}), l \in L, i \in I_l,\ s \in S$ // for simplicity, $\boldsymbol{x}$ denotes $(\boldsymbol{b}, \boldsymbol{a}, \boldsymbol{E}, \boldsymbol{t})$.
  Obtain the Lagrange multipliers $\boldsymbol{\mu}^{(k)}$ and Lagrange function $\mathcal{L}(\boldsymbol{x}^{(k)}, \overline{\boldsymbol{y}}, \boldsymbol{\xi}^{(k)}, \boldsymbol{\mu}^{(k)})$.
  Generate the optimality cut, represented by $C^{(k)}$.
  If $\boldsymbol{\xi}^{(k)} \leq \epsilon'$ **then** // check the feasibility of PS.
    $\mathrm{UB}^{(k+1)} \leftarrow \min\{\mathrm{UB}^{(k)}, OBJ_{\mathrm{PS}}(\boldsymbol{x}^{(k)})\}$ // OBJ_{PS} records the objective value of PS under solution $\boldsymbol{x}^{(k)}$.
  End if
  Add generated cut $C^{(k)}$ into RMP.
  Solve RMP to update the values of complicating variables $\boldsymbol{y}^{(k+1)} \leftarrow (\tau_{l,i}^{(k+1)}, \gamma_{2,i}^{(k+1)}, \beta_{l,i}^{(k+1)})$.
  $\mathrm{LB}^{(k+1)} \leftarrow OBJ_{\mathrm{RMP}}(\boldsymbol{y}^{(k+1)})$.
  $k \leftarrow k + 1$.
End while
The algorithm terminates as the optimal solution is found.

(2) Primal Subproblem: Multi-Vehicle Trajectory Optimization Model

Given the fixed values of complicating variables (i.e., terminal time $\boldsymbol{\tau}$, lane selection variable $\boldsymbol{\gamma}$, and facilitating vehicle selection $\boldsymbol{\beta}$), the resulting PS model is a convex NLP model that determines trajectories, including longitudinal jerk $\boldsymbol{b}$, acceleration $\boldsymbol{a}$, kinetic energy $\boldsymbol{E}$, and time $\boldsymbol{t}$. For convenience, we define a set of trajectory-related variables as $\boldsymbol{x} := (b_{l,i,s}, a_{l,i,s}, E_{l,i,s}, t_{l,i,s} | l \in L, i \in I_l, s \in S)$ and a set of complicating variables as $\boldsymbol{y} := (\tau_{l,i}, \gamma_i, \beta_{l,i} | l \in L, i \in I_l)$.

The PS model given the fixed value of scheduling-related variables, i.e., $\overline{\boldsymbol{y}} = (\bar{\tau}_{l,i}, \bar{\gamma}_i, \bar{\beta}_{l,i} | l \in L, i \in I_l)$, can be formulated as follows:

$$[\mathrm{PS}] f_{\mathrm{PS}}(\boldsymbol{x}, \overline{\boldsymbol{y}}) = \min \sum_{l \in L} \sum_{i \in I_l} \bar{\tau}_{l,i} + w_{\mathrm{lc}} \sum_{i \in I_2} \bar{\gamma}_i + \sum_{l \in L} \sum_{i \in I_l} \sum_{s \in \{p_{l,i}, \cdots, s_{\mathrm{m}}\}} [w_{\mathrm{a}} a_{l,i,s}^2 + w_{\mathrm{b}} b_{l,i,s}^2 + w_E (\bar{E} - E_{l,i,s})^2] \quad (4.44)$$

subject to

constraints (4.21)–(4.26), (4.30)–(4.31), (4.33)–(4.36),

$$t_{l,i,s_m} = \bar{\tau}_{l,i} \qquad \forall l \in L, i \in I_l \tag{4.45}$$

$$t_{3,j-1,s} - t_{2,i,s} - h + M \cdot (1 - \bar{\beta}_{i,j}) \geq 0 \qquad \forall i \in I_2, j \in I_3 \backslash \{1\}, s \in \{s_{\mathrm{lc}}, \cdots, s_{\mathrm{m}}\} \tag{4.46}$$

$$t_{2,i,s} - t_{3,j,s} - h + M \cdot (1 - \bar{\beta}_{i,j}) \geq 0 \qquad \forall\, i \in I_2, j \in I_3, s \in \{s_{\mathrm{lc}}, \cdots, s_{\mathrm{m}}\} \tag{4.47}$$

$$|\bar{\gamma}_i - \bar{\gamma}_j| M + \; t_{2,i,s} - t_{2,j,s} - h \geq 0 \qquad \forall i \in I_2 \backslash \{1\}, j \in \{1, \cdots, i-1\}\, s \in \{s_{\mathrm{lc}}, \cdots, s_{\mathrm{m}}\}. \tag{4.48}$$

Given the fixed value of complicating variables, the model [PS] solves the corresponding trajectories. Objective function (4.44) optimizes smoothness of vehicle acceleration as well as jerk profiles, while striving to keep vehicle speeds close to the free-flow speed. Constraints (4.45)–(4.48), called coupling constraints, update constraints (4.27), (4.28), (4.29) and (4.32), respectively.

(3) Acceleration through the Feasibility-Guaranteed Primal Subproblem

For some scheduling solutions of RMP, such as some vehicle sequences, terminal time, and lane changes decisions, PS fails to find feasible trajectories, due to the constraints of limited cooperation areas, bounded acceleration and deceleration, and minimum time headway requirement between vehicles. To avoid such infeasibilities, we relax subproblem's coupling constraints by introducing elastic slack variables and penalize these slacks in the objective function. In a final solution, these slacks are sufficiently close to zero.

By introducing four non-negative elastic slack variables ($\boldsymbol{\xi} = (\dot{\xi}_{l,i}, \ddot{\xi}_{i,j,s}, \dddot{\xi}_{i,j,s}, \acute{\xi}_{i,j,s})$), coupling constraints (4.45)–(4.48) are transformed into constraints (4.49)–(4.52), respectively, along with constraints (4.53)–(4.56). Also, the objective function of PS is transformed into objective (4.57). Consequently, the GBD procedure only requires optimality cuts, simplifying the process significantly. This significantly improves the efficiency of the algorithm by eliminating the need for feasibility cuts and ensuring that each iteration directly contributes to converging to the optimal solution.

$$-\dot{\xi}_{l,i} \leq t_{l,i,s_m} - \bar{\tau}_{l,i} \leq \dot{\xi}_{l,i} \qquad \forall l \in L, i \in I_l \tag{4.49}$$

$$t_{3,j-1,s} - t_{2,i,s} - h + M \cdot (1 - \bar{\beta}_{i,j}) + \ddot{\xi}_{i,j,s} \geq 0 \qquad \forall i \in I_2, j \in I_3 \backslash \{1\}, s \in \{s_{\mathrm{lc}}, \cdots, s_{\mathrm{m}}\} \tag{4.50}$$

$$t_{2,i,s} - t_{3,j,s} - h + M \cdot (1 - \bar{\beta}_{i,j}) + \dddot{\xi}_{i,j,s} \geq 0 \qquad \forall\, i \in I_2, j \in I_3, s \in \{s_{\mathrm{lc}}, \cdots, s_{\mathrm{m}}\} \tag{4.51}$$

$$|\bar{\gamma}_i - \bar{\gamma}_j| M + \; t_{2,i,s} - t_{2,j,s} - h + \dddot{\xi}_{i,j,s} \geq 0 \qquad \forall i \in I_2 \backslash \{1\}, j \in \{1, \cdots, i-1\}\, s \in \{s_{\mathrm{lc}}, \cdots, s_{\mathrm{m}}\} \tag{4.52}$$

$$\dot{\xi}_{l,i} \geq 0 \qquad \forall l \in L, i \in I_l \tag{4.53}$$

$$\ddot{\xi}_{i,j,s} \geq 0 \qquad \forall i \in I_2, j \in I_3 \backslash \{1\}, s \in \{s_{\mathrm{lc}}, \cdots, s_{\mathrm{m}}\} \tag{4.54}$$

$$\dddot{\xi}_{i,j,s} \geq 0 \qquad \forall\, i \in I_2, j \in I_3, s \in \{s_{\mathrm{lc}}, \cdots, s_{\mathrm{m}}\} \tag{4.55}$$

$$\acute{\xi}_{i,j,s} \geq 0 \qquad \forall i \in I_2 \backslash \{1\}, j \in \{1, \cdots, i-1\}\, s \in \{s_{\mathrm{lc}}, \cdots, s_{\mathrm{m}}\} \tag{4.56}$$

$$f_{\mathrm{PS}}^{\mathrm{feasibility}}(\boldsymbol{x}, \boldsymbol{\xi}, \bar{\boldsymbol{y}}) = \min \textstyle\sum_{l \in L} \sum_{i \in I_l} \bar{\tau}_{l,i} + w_{\mathrm{lc}} \sum_{i \in I_2} \bar{\gamma}_i + \sum_{l \in L} \sum_{i \in I_l} \sum_{s \in \{p_{l,i}, \cdots, s_{\mathrm{m}}\}} [w_{\mathrm{a}} a_{l,i,s}^2 + w_{\mathrm{b}} b_{l,i,s}^2 +$$
$$w_{\mathrm{E}} \left(\bar{E} - E_{l,i,s}\right)^2] + M[\textstyle\sum_{l \in L} \sum_{i \in I_l} \dot{\xi}_{l,i} + \sum_{i \in I_2} \sum_{j \in I_3 \backslash \{1\}} \sum_{s \in \{s_{\mathrm{lc}}, \cdots, s_{\mathrm{m}}\}} \ddot{\xi}_{i,j,s} +$$
$$\textstyle\sum_{i \in I_2} \sum_{j \in I_3} \sum_{s \in \{s_{\mathrm{lc}}, \cdots, s_{\mathrm{m}}\}} \dddot{\xi}_{i,j,s} + \sum_{i \in I_2 \backslash \{1\}} \sum_{j \in \{1, \cdots, i-1\}} \sum_{s \in \{s_{\mathrm{lc}}, \cdots, s_{\mathrm{m}}\}} \acute{\xi}_{i,j,s}]. \tag{4.57}$$

(4) GBD Cuts from the Feasibility-Guaranteed PS Model

At k^{th} algorithm iteration, $\boldsymbol{x}^{(k)}$ records the values of trajectory-related variables, its optimality cut can be written as follows:

$$\eta \geq \mathcal{L}\left(\boldsymbol{x}^{(k)}, \overline{\boldsymbol{y}}, \boldsymbol{\xi}^{(k)}, \boldsymbol{\mu}^{(k)}\right) + \nabla_{\mathbf{y}}^{T}\mathcal{L}\left(\boldsymbol{x}^{(k)}, \overline{\boldsymbol{y}}, \boldsymbol{\xi}^{(k)}, \boldsymbol{\mu}^{(k)}\right) \cdot (\boldsymbol{y} - \overline{\boldsymbol{y}}), \tag{4.58}$$

where η and $\boldsymbol{y}$ are variables in the model [RMP] which is introduced in Section 4.5; $\overline{\boldsymbol{y}}$ is the given values of complicating variables from the k^{th} iteration, i.e., $\overline{\boldsymbol{y}} := \boldsymbol{y}^{(k)}$; $\boldsymbol{\mu}^{(k)}$ records the values of optimal Lagrange multipliers obtained in the k^{th} iteration; $\mathcal{L}\left(\boldsymbol{x}^{(k)}, \overline{\boldsymbol{y}}, \boldsymbol{\xi}^{(k)}, \boldsymbol{\mu}^{(k)}\right)$ denotes the Lagrange function of the feasibility-guaranteed PS and is calculated as follows:

$$\mathcal{L}\left(\boldsymbol{x}^{(k)}, \overline{\boldsymbol{y}}, \boldsymbol{\xi}^{(k)}, \boldsymbol{\mu}^{(k)}\right) = f_{\text{PS}}^{\text{feasibility}}\left(\boldsymbol{x}^{(k)}, \overline{\boldsymbol{y}}, \boldsymbol{\xi}^{(k)}\right) + (\boldsymbol{\mu}^{(k)})^{T}\boldsymbol{G}\left(\boldsymbol{x}^{(k)}, \overline{\boldsymbol{y}}, \boldsymbol{\xi}^{(k)}\right), \tag{4.59}$$

where $f_{\text{PS}}^{\text{feasibility}}$ represents the objective function (4.57); $\boldsymbol{G}$ denotes the vector of coupling constraints (4.49)–(4.52) under $\boldsymbol{x}^{(k)}$, $\overline{\boldsymbol{y}}$, and $\boldsymbol{\xi}^{(k)}$. $\nabla_{\boldsymbol{y}}\mathcal{L}$ denotes the Jacobian of the Lagrange function with respect to the set of complicating variables $\boldsymbol{y}$.

(5) Relaxed Master Problem: Multi-Vehicle Scheduling Optimization Model

RMP needs to determine the terminal time ($\boldsymbol{\tau}$), facilitating vehicle selection for merging and lane-changing vehicles ($\boldsymbol{\alpha}, \boldsymbol{\beta}$), lane-changing decision ($\boldsymbol{\gamma}, \boldsymbol{\Omega}$), and an auxiliary variable (η). Recall that ($\boldsymbol{\tau}$, $\boldsymbol{\gamma}$, $\boldsymbol{\beta}$) are selected as complicating variables $\boldsymbol{y}$. The model [RMP] is therefore formulated as follows:

$$\text{[RMP]} \quad f(\boldsymbol{y}, \boldsymbol{\alpha}, \eta) = \min \eta \tag{4.60}$$

subject to constraints (4.6)–(4.20), (4.37)–(4.41), optimality cut (4.58).

(6) Linearization of the Relaxed Master Problem

The model [RMP] contains nonlinear constraints (4.8), (4.12)–(4.13), (4.16)–(4.20), such as the absolute function, the rounding down function, the multiplication of binary and continuous variables, as well as the multiplication of binary variables. To address these nonlinear parts, we apply specific linearization techniques to transform the model [RMP] into an MILP model. Before linearizing the model [RMP], newly needed parameters and variables are defined as follows.

Newly defined parameters

M_m big-M for linearization; $m \in \{1, \dots, 3\}$.

r number infinitesimally close to 1 but still less than 1.

Newly defined variables

π_{ij} binary, equal to the value of $|\gamma_i - \gamma_j|$; $i \in I_2 \backslash \{1\}, j \in \{1, \dots, i-1\}$.

ϵ_{ijm} binary, equal to 1 if and only if both α_{ij} and γ_m, 0 otherwise; $i \in I_1, j \in I_2, m \in I_2$.

θ_{ijm} continuous, equal to $\tau_{3,m}$ if and only if $\beta_{i,j}$ is 1, 0 otherwise; $i \in I_2, j \in I_3 \cup \{|I_3| + 1\}, m \in I_3$.

σ_{ijm} continuous, equal to $\tau_{2,m}$ if and only if α_{ij} is 1, 0 otherwise; $i \in I_1, j \in I_2^+, m \in I_2$.

δ_{imj} continuous, equal to $\tau_{2,j}$ if and only if $\epsilon_{i,m,j}$ is 1, 0 otherwise; $i \in I_1, m \in I_2^+, j \in I_2$.

(1): Linearization of the absolute function: To linearize constraints (4.8) containing the absolute function, a new binary variable π_{ij} is first defined to replace $\left|\gamma_i - \gamma_j\right|$ with the help of constraints (4.61)–(4.63).

$$\pi_{ij} \geq \gamma_i - \gamma_j \qquad \forall i \in I_2\backslash\{1\}, j \in \{1, \dots, i-1\} \tag{4.61}$$

$$\pi_{ij} \geq \gamma_j - \gamma_i \qquad \forall i \in I_2\backslash\{1\}, j \in \{1, \dots, i-1\} \tag{4.62}$$

$$\pi_{ij} \in \{0,1\} \qquad \forall i \in I_2\backslash\{1\}, j \in \{1, \dots, i-1\}. \tag{4.63}$$

Then, constraints (4.8) can be transformed into linear constraints (4.64).

$$\pi_{ij} M + \left(\tau_{2,i} - \tau_{2,j} - h\right) \geq 0 \qquad \forall i \in I_2\backslash\{1\}, j \in \{1, \dots, i-1\}. \tag{4.64}$$

(2) Linearization of the product of two binary variables: Constraints (4.16) and (4.17) contain the product of binary $\boldsymbol{\gamma}$ and binary $\boldsymbol{\alpha}$. A new binary variable ϵ_{ijm} is defined to replace this type of nonlinear term, and newly defined constraints are summarized as constraints (4.65)–(4.68).

$$\epsilon_{ijm} \leq \gamma_m \qquad \forall i \in I_1, j \in I_2, m \in I_2 \tag{4.65}$$

$$\epsilon_{ijm} \leq \alpha_{ij} \qquad \forall i \in I_1, j \in I_2, m \in I_2 \tag{4.66}$$

$$\epsilon_{ijm} \geq \gamma_m + \alpha_{ij} - 1 \qquad \forall i \in I_1, j \in I_2, m \in I_2 \tag{4.67}$$

$$\epsilon_{ijm} \in \{0,1\} \qquad \forall i \in I_1, j \in I_2, m \in I_2. \tag{4.68}$$

Therefore, constraints (4.16) and (4.17) are replaced with (4.69) and (4.70), respectively.

$$\epsilon_{ijj} - \epsilon_{i,(j+1),j} = 0 \qquad \forall i \in I_1, j \in I_2\backslash\{|I_2|\} \tag{4.69}$$

$$\sum_{j \in I_2^+} \alpha_{i,j} = 1 + \sum_{j \in I_2} \epsilon_{ijj} \qquad \forall i \in I_1. \tag{4.70}$$

(3) Linearization of the product of a continuous variable and a binary variable: To linearize constraints (4.12)–(4.13) containing the product of a binary variable and a continuous variable, a new continuous variable θ_{ijm} is defined, and constraints (4.71)–(4.73) are needed.

$$\theta_{ijm} \geq \tau_{3,m} + (\beta_{i,j} - 1)M_1 \qquad \forall i \in I_2, j \in I_3 \cup \{|I_3| + 1\}, m \in I_3 \tag{4.71}$$

$$\theta_{ijm} \leq \tau_{3,m} \qquad \forall i \in I_2, j \in I_3 \cup \{|I_3| + 1\}, m \in I_3 \tag{4.72}$$

$$\theta_{ijm} \geq 0 \qquad \forall i \in I_2, j \in I_3 \cup \{|I_3| + 1\}, m \in I_3. \tag{4.73}$$

Hence, constraints (4.12)–(4.13) are transformed into constraints (4.74)–(4.75).

$$\gamma_i M + \tau_{2,i} - \sum_{j=2}^{|I_3|+1} \theta_{i,j,j-1} - h \geq 0 \qquad \forall i \in I_2 \tag{4.74}$$

$$\gamma_i M + \left(\sum_{j \in I_3} \theta_{i,j,j} - \tau_{2,i} - h\right)|I_3| \geq 0 \qquad \forall i \in I_2. \tag{4.75}$$

(4) Linearization of the product of a continuous variable and two binary variables: To linearize a multiplication of three variables. We first deal with the product of two binary variables and then deal with the multiplication of a binary variable and a continuous variable. Constraints (4.18) and (4.19) are linearized sequentially. First, constraints (4.18)–(4.19) can be rewritten as constraints (4.76)–(4.77) with the help of ϵ_{ijm}. Continuous variables σ_{ijm} and δ_{imj} are then defined to replace $\alpha_{i,j}\tau_{2,j-1}$ and $\epsilon_{i,j,(j-1)}\tau_{2,j-1}$, respectively, and related constraints (4.78)–(4.83) are added. Constraints (4.18)–(4.19) are therefore replaced with constraints (4.84)–(4.85), respectively:

$$\tau_{1,i} - h \geq \sum_{j=2}^{|I_2|+1}(\alpha_{i,j}\tau_{2,j-1} - \epsilon_{i,j,j-1}\tau_{2,j-1}) \qquad \forall i \in I_1 \tag{4.76}$$

$$\sum_{j\in I_2}(\alpha_{i,j}\tau_{2,j} - \epsilon_{i,j,j}\tau_{2,j}) - \tau_{1,i} - h + \Omega M \geq 0 \qquad \forall i \in I_1 \tag{4.77}$$

$$\sigma_{ijm} \geq \tau_{2,m} + (\alpha_{ij} - 1)M_2 \qquad \forall i \in I_1, j \in I_2^+, m \in I_2 \tag{4.78}$$

$$\sigma_{ijm} \leq \tau_{2,m} \qquad \forall i \in I_1, j \in I_2^+, m \in I_2 \tag{4.79}$$

$$\sigma_{ijm} \geq 0 \qquad \forall i \in I_2, j \in I_2^+, m \in I_2 \tag{4.80}$$

$$\delta_{imj} \geq \tau_{2,j} + \left(\epsilon_{i,m,j} - 1\right)M_3 \qquad \forall i \in I_1, m \in I_2^+, j \in I_2 \tag{4.81}$$

$$\delta_{imj} \leq \tau_{2,j} \qquad \forall i \in I_1, m \in I_2^+, j \in I_2 \tag{4.82}$$

$$\delta_{imj} \geq 0 \qquad \forall i \in I_1, m \in I_2^+, j \in I_2 \tag{4.83}$$

$$\tau_{1,i} - h \geq \sum_{j\in I_2}\left(\sigma_{i,(j+1),j} - \delta_{i,(j+1),j}\right). \qquad \forall i \in I_1 \tag{4.84}$$

$$\sum_{j\in I_2}(\sigma_{i,j,j} - \delta_{i,j,j}) - \tau_{1,i} - h + \Omega M \geq 0 \qquad \forall i \in I_1. \tag{4.85}$$

Finally, constraints (4.16) are first rewritten with ϵ_{ijm} and σ_{ijm} as constraints (4.86). Then, since the existing of η_{imj}, constraints (4.86) can be further translated into linear constraints (4.87).

$$\sum_{j\in I_2}\sigma_{ijj} - \sum_{j\in I_2}\epsilon_{ijj}\tau_{2,j} + \Omega M \geq \tau_{1,i} + h \,\forall i \in I_1 \tag{4.86}$$

$$\sum_{j\in I_2}\sigma_{ijj} - \sum_{j\in I_2}\eta_{ijj} + \Omega M \geq \tau_{1,i} + h \qquad \forall i \in I_1. \tag{4.87}$$

(5) Linearization of the floor function: constraint (4.20) contains the floor function and can be rewritten as follows:

$$\frac{\sum_{j\in I_2}\gamma_j}{|I_2|} - r \leq \Omega \leq \frac{\sum_{j\in I_2}\gamma_j}{|I_2|}. \tag{4.88}$$

In summary, the linear model of RMP is summarized as follows:

Objective (4.60)

subject to constraints (4.6), (4.7), (4.9)–(4.11), (4.14)–(4.15), (4.37)–(4.41), (4.58), (4.61)–(4.75), (4.78)–(4.85), (4.87)–(4.88).

(7) Finite Convergence Guarantee

Proposition 1: The finite convergence of the GBD algorithm applied in this problem is guaranteed.

Proof: To guarantee the finite convergence of the GBD algorithm, the following issues need to be checked.

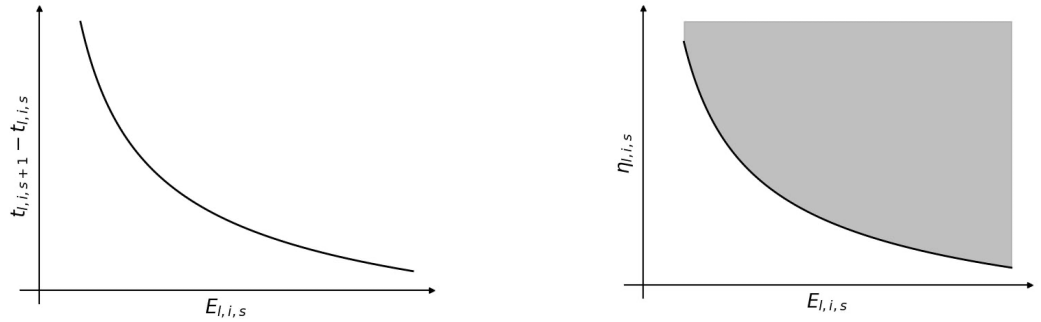
In subproblem [PS], the feasible region (represented by X) of the subproblem variables is nonempty and convex. X is defined by a combination of linear constraints (4.21)–(4.22), (4.24)–(4.36), and non-convex equality constraints (4.23), i.e. $t_{l,i,s+1} - t_{l,i,s} - \frac{1}{\sqrt{2E_{l,i,s}}}\cdot\Delta = 0$. By introducing nonnegative slack variables $\eta_{l,i,s}$, constraints (4.23) are relaxed to the following constraints,

$$t_{l,i,s+1} - t_{l,i,s} - \eta_{l,i,s} = 0 \qquad \forall l \in L, i \in I_l, s \in \{p_{l,i}, \cdots, s_{\mathrm{m}} - 1\} \tag{4.89}$$

$$\eta_{l,i,s} \geq \frac{1}{\sqrt{2E_{l,i,s}}}\cdot\Delta \qquad \forall l \in L, i \in I_l, s \in \{p_{l,i}, \cdots, s_{\mathrm{m}} - 1\} \tag{4.90}$$

$$\eta_{l,i,s} \geq 0 \qquad \forall l \in L, i \in I_l, s \in \{p_{l,i}, \cdots, s_{\mathrm{m}} - 1\}. \tag{4.91}$$

This relaxation transfers non-convex constraints (4.23) to convex constraints (4.89)–(4.90), as illustrated in Figure 4.4. Moreover, a penalty term for $\eta_{l,i,s}$ is added in the objective function to encourage $\boldsymbol{\eta}$ to equal $\frac{1}{\sqrt{2\boldsymbol{E}}} \cdot \Delta$, thereby ensuring adherence to the original equality constraints. Note that penalizing $\boldsymbol{\eta}$ results in a smaller value for $\boldsymbol{\eta}$, corresponding to larger $\boldsymbol{E}$, which means to maximizing vehicle speeds. Through this transformation, the transformed feasibility domain X becomes a convex set.



(a) Non-convex set of $t_{l,i,s+1} - t_{l,i,s} - \frac{1}{\sqrt{2E_{l,i,s}}} \cdot \Delta = 0$

(b) Convex set of $\eta_{l,i,s} \geq \frac{1}{\sqrt{2E_{l,i,s}}} \cdot \Delta$

Figure 4.4. Transformation of nonlinear kinematic equations.

(a) Non-convex set formed by the equality constraints (4.23); (b) Convex set formed by the relaxed inequality constraints (4.90).

In subproblem [PS], the objective function (denoted as f_{PS}), and the coupling constraint functions (represented by $\boldsymbol{G}$) are convex on the subproblem variables for each fixed complicating variables. This convexity criterion is satisfied because the objective function f_{PS} (i.e., Equation (4.44)) is formulated as a quadratic function form and the coupling constraints $\boldsymbol{G}$ (i.e., constraints (4.45)–(4.48)) are linear with respect to the subproblem variables when complicating variables become constant. The relaxed master problem [RMP] can be solved to optimality in each iteration. This optimality criterion is met because the model [RMP] is a mixed-integer linear programming model. □

4.3 Computational Experiments

Section 4.3.1 introduces the setup of the numerical experiments. Section 4.3.2 compares the proposed algorithm with the state-of-the-art solver Gurobi to evaluate algorithm performance in terms of solution quality and time. Section 4.3.3 introduces the baseline methods. Section 4.3.4 and 4.3.5 compares the proposed model with several commonly used models to assess the enhancement on traffic performance. Section 4.4.6 presents two case studies to analyse generated trajectories as well as merging and lane-changing decisions. Section 4.4.7 presents sensitivity analyses exploring the effects of different lengths of cooperative areas and the costs associated with lane changes. Finally, Section 4.4.8 discusses the benefits of the proposed integrated model.

(1) Experimental Setting

Numerical experiments are conducted using a microscopic traffic simulation tool named by SUMO. The simulation employs the Wiedemann 99 model to replicate car-following behaviour. Table 4.2 lists the vehicle parameter values used in the simulations.

Table 4.2. Parameter setting summary.

Parameters	Value
Vehicle length (l_{v})	4.37 m
Standstill distance (s_0)	1.5 m
Discrete distance interval (Δ)	5 m
Minimum headway (h)	1.5 s
Free flow speed (v_{f})	100 km/h
Maximum and minimum jerks ($\bar{b}, \underline{b}$)	$2\ \mathrm{m/s^3}, -2\ \mathrm{m/s^3}$
Maximum acceleration and deceleration ($\bar{a}, \underline{a}$)	$2.75\ \mathrm{m/s^2}, -2.75\ \mathrm{m/s^2}$

As shown in Figure 4.5, the simulated freeway segment comprises two mainline lanes and one on-ramp, partitioned into a cooperative lane-changing area, a cooperative merging area, and a downstream segment. The cooperative area, including both the lane-changing and merging areas, spans 700 meters, while the downstream segment extends one kilometre to fully cover impacted road sections. A trigger point is placed on the on-ramp lane.

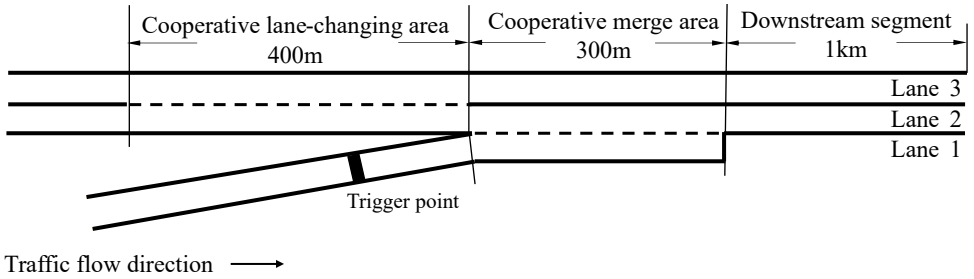


Figure 4.5. Road setting.

(2) Comparison of the Proposed Algorithm and the State-of-the-Art Solver

The proposed GBD-based algorithm is compared with the Gurobi solver in terms of objective function values and solution time for different scale instances ranging from three to nine vehicles. Table 4.3 compares the performance of the proposed GBD-based algorithm against the Gurobi solver across varying numbers of vehicles, ranging from three to nine. Let $\mathrm{OBJ_P}$, and $\mathrm{OBJ_G}$ represent the objective function values obtained by the proposed algorithm, and the Gurobi solver, respectively; $\mathrm{T_P}$, and $\mathrm{T_G}$ represent the solution time required by the proposed algorithm, and the Gurobi solver, respectively. As shown in Table 4.3, the proposed algorithm can solve all instances ranging from three to nine vehicles within 3.1 seconds. However, the solution time of the Gurobi solver is significantly longer. For small-scale instances with three to four vehicles, the Gurobi solver can find optimal solutions within a few seconds; for medium-scale instances with five to eight vehicles, it requires several hundred seconds to reach optimality; for large-scale instances with nine vehicles, it takes almost eight hours. Clearly, our algorithm significantly outperforms the Gurobi solver in terms of solution time, which further advances practical applications. The inefficiency of the method directly using the Gurobi solver arises from the fact that each additional vehicle not only introduces corresponding binary variables into the model, but also brings in hundreds of continuous trajectory variables and numerous convex nonlinear kinematic constraints. These extra variables and constraints greatly increase the model's complexity, making it extremely hard to solve. In contrast, the proposed GBD-based algorithm adds only a minimal number of essential constraints to the MILP master problem, thereby significantly enhancing solution efficiency. Consequently, our GBD-based algorithm can converge and find the optimal solution within 3.1 seconds for the large-scale instances where the Gurobi solver cannot find the optimal solution within eight hours.

Table 4.3. Comparison of the proposed GBD-based algorithm and the Gurobi solver.

Number of vehicles	Objective function value		Solution time (second)	
	OBJ_P	OBJ_G	T_P	T_G
3	25.829	25.829	0.8	5.3
4	28.044	28.044	1.3	11.3
5	25.527	25.527	1.5	245.6
6	27.120	27.120	1.8	447.3
7	27.863	27.863	2.2	482.7
8	27.952	27.952	3.0	597.4
9	27.228	27.228	3.1	28707.9

Notes: (1) OBJ_P and OBJ_G represent the objective function value obtained by the proposed GBD-based algorithm, and the Gurobi solver, respectively; (2) T_P and T_G represent the solution time of the proposed GBD-based algorithm, and the Gurobi solver, respectively.

(3) Baseline approaches

The proposed model has three primary characteristics: (1) sequencing optimization, (2) lane change optimization, and (3) utilization of microscopic position and speed information. To evaluate the advantages of these three characteristics, the proposed model is compared with the following three models:

(i) FIFO_NLC model: A heuristic first-in-first-out (FIFO) approach that schedules the sequences of on-ramp and outside-lane vehicles without permitting lane changes. Here, note that the rule of no lane changes (NLC) is strictly enforced.

(ii) OPT_SEQ_NLC model: An optimization method adapted from the proposed model, determining the optimal sequences of on-ramp and outside-lane vehicles (OPT_SEQ) while maintaining the rule of NLC.

(iii) Balanced flow model: An optimization-based method focused on balancing traffic flow distribution by controlling the number of lane-changing vehicles. This method combines heuristics for selecting lane-changing vehicles and an optimization model for sequencing on-ramp and outside-lane vehicles.

The experiments are conducted under various demand scenarios, including different demand ratios between lanes 2 to 1, total demands of lanes 2 and 1 (in vehicles per hour), and total demands of lane 3 (in vehicles per hour). Traffic performance metrics used in the comparison include total delay (in seconds) and average lane delay (in seconds), denoted by D^i and $\overline{D}_{l_j}^i$, respectively. Here, the superscript represents different methods, with F, O, B, and P corresponding to FIFO_NLC, OPT_SEQ_NLC, the balanced flow model, and the proposed model, respectively; the subscript specifies lanes 1, 2, and 3 by l_1, l_2, and l_3, respectively. Additionally, "D. I.", and "No. lc" represent the delay improvement ratio (%), and the number of lane changes, respectively.

(4) Comparative analysis: Benefits of sequencing and lane-changing optimization

Table 4.4 shows 18 demand scenarios with varying demand ratios and levels, each simulated in SUMO for one hour. Specifically, for the on-ramp and outside lane (i.e., lanes 1 and 2), two demand ratios, 2:1 and 3:1, between lane 2 to lane 1 are considered, along with three demand totals for lanes 1 and 2, 1,400, 1,600, and 1,800 vehicles per hour. For the inside lane (i.e., lane 3), three demand levels are set accordingly. Due to the interactions between the on-ramp and mainline flow, the lane capacity under each demand ratio varies. Here, note that, in principle, 1,400 vehicles/hour represents free flow,

1,600 vehicles/hour indicates near saturation, and 1,800 vehicles/hour is saturated or oversaturated.

Numerical experiments are conducted on two models that prohibit lane changes (i.e., FIFO_NLC and OPT_SEQ_NLC), and the proposed model in Table 4.4. The performance comparison between FIFO_NLC and OPT_SEQ_NLC investigates the influence of the first characteristic, namely the sequencing optimization. Then the comparison among FIFO_NLC, OPT_SEQ_NLC, and the proposed model examines the effect of the second characteristic, i.e., the lane change optimization.

Table 4.4. Comparison of travel delays.

Demand ratio Lane 2 / Lane 1	Demand of Lane1&2 veh/hour	Demand of Lane 3 veh/hour	FIFO_NLC			OPT_SEQ_NLC			Proposed model					
			D^{F} (s)	$\overline{D}_{l_1}^{\mathrm{F}}$ (s)	$\overline{D}_{l_2}^{\mathrm{F}}$ (s)	D^{O} (s)	$\overline{D}_{l_1}^{\mathrm{O}}$ (s)	$\overline{D}_{l_2}^{\mathrm{O}}$ (s)	D^{P} (s)	D. I.	$\overline{D}_{l_1}^{\mathrm{P}}$ (s)	$\overline{D}_{l_2}^{\mathrm{P}}$ (s)	$\overline{D}_{l_3}^{\mathrm{P}}$ (s)	No.lc
2:1	1,400	1,300	846	0.76	0.53	839	1.12	0.3	**512**	40%	0.43	0.18	0.11	154
		1,400							**576**	32%	0.50	0.18	0.12	130
		1,500						4	**635**	25%	0.56	0.21	0.12	106
	1,600	1,500	1,203	0.84	0.71	1,196	1.27	0.4	**845**	30%	0.54	0.28	0.18	146
		1,600							**949**	21%	0.66	0.28	0.18	120
		1,700						9	**1,123**	7%	0.96	0.39	0.12	55
	1,800	1,700	1,850	1.00	1.04	1,838	1.61	0.7	**1,538**	17%	0.98	0.49	0.21	99
		1,800							**1,762**	5%	1.51	0.63	0.06	19
		1,900						3	**1,827**	1%	1.63	0.68	0.01	3
3:1	1,400	1,300	773	0.86	0.45	770	1.32	0.3	**419**	46%	0.43	0.13	0.10	142
		1,400							**480**	38%	0.48	0.17	0.09	114
		1,500						0	**568**	27%	0.58	0.18	0.12	102
	1,600	1,500	1,106	0.95	0.60	1,099	1.47	0.4	**748**	32%	0.62	0.21	0.16	127
		1,600							**838**	24%	0.68	0.24	0.18	110
		1,700						3	**971**	12%	0.90	0.29	0.15	75
	1,800	1,700	1,790	1.12	0.95	1,776	2.05	0.6	**1,404**	22%	1.15	0.42	0.19	109
		1,800							**1,696**	5%	1.35	0.48	0.20	32
		1,900						3	**1,776**	1%	2.05	0.63	0.00	0

Notes: (1) the superscripts F, O, and P represent the three methods: FIFO_NLC, OPT_SEQ_NLC, and the proposed model, respectively; the subscripts l_1, l_2, and l_3 refer to the on-ramp, outside lane, and inside lane, respectively; (2) D denotes the total delay in seconds; (3) $\overline{D}_{l_1}$, $\overline{D}_{l_2}$, and $\overline{D}_{l_3}$ represent the average lane delays of the related lanes; (4) D. I. refers to the delay improvement ratio and can be calculated by $\frac{D^F - D^P}{D^F}$; (4) No. lc refers to the number of lane changes; (5) "vehicles per hour" is abbreviated as "veh/hour".

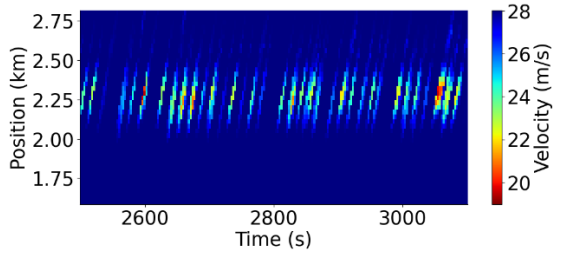
In the comparison of the methods without lane changes, namely FIFO_NLC and OPT_SEQ_NLC, the impact of sequencing optimization is evaluated. From Table 4.4, all values in the column D^{O} are smaller than values in the column D^{F}, indicating that OPT_SEQ_NLC has less total delay than FIFO_NLC. More importantly, values in the column $\overline{D}_{l_2}^{\mathrm{O}}$ are notably lower than those in the column $\overline{D}_{l_2}^{\mathrm{F}}$, highlighting that the average delay in the outside mainline lane by OPT_SEQ_NLC is significantly lower compared to that by FIFO_NLC. This comparison reflects that better vehicle sequencing can significantly mitigate disruptions from ramps, reducing them by up to 35.8%.

Then, to evaluate the impact of lane change optimization, the proposed model is compared with two other models. From the columns D^{P}, $\overline{D}_{l_1}^{\mathrm{P}}$, and $\overline{D}_{l_2}^{\mathrm{P}}$, we can see the significant improvements in the total and average lane delays. Additionally, there is a slight increase in average delay (0.1–0.2 seconds)

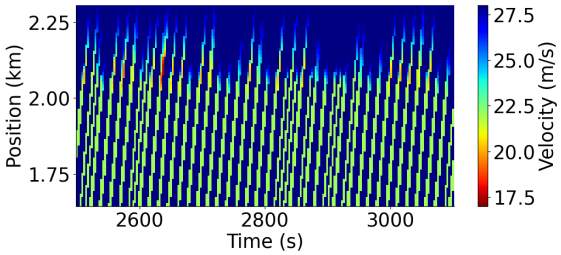
from inside-lane vehicles (column $\overline{D}_{l_3}^{\mathrm{P}}$). Here, note that the proposed model significantly decreases the total delay D^{P} by 12% to 46% when the inside lane is not saturated (i.e., when the demand of lane 3 is less than 1,800 vehicles per hour). On the other hand, the improvement ranges from 1% to 4% when the demand of the inside lane is near or exceeds capacity (i.e., when the demand of lane 3 is 1,800 to 1,900 vehicles per hour). This decrease in improvement is expected because there is no free (extra) space available in the inside lane to help absorb disruptions from ramps under the saturated condition. This demonstrates that the proposed model makes lane-changing decisions based on the consideration of the overall traffic efficiency. Moreover, the average outside lane delay $\overline{D}_{l_2}^{\mathrm{P}}$ is further reduced, indicating that leveraging multilane capacity enhances system resilience against disturbances from ramps.

Moreover, we illustrate the improvement in speed variation achieved by the proposed model compared with FIFO_NLC and OPT_SEQ_NLC. High variation of speed is a known precursor to collision accidents (Tian et al., 2019). In Figure 4.6, the time-space region under evaluation is defined as [2400, 3200] seconds by [1750, 2750] meters. The cooperative control area starts at 2 kilometers, and the merge gore is at 2.4 kilometers. Figures 4.6(a), 4.6(c), and 4.6(e) depict the heatmap of the average speed in the outside lane under the control of FIFO_NLC, OPT_SEQ_NLC, and the proposed model, respectively. Similarly, Figures 4.6(b), 4.6(d), and 4.6(f) present the velocity heatmap in the ramp and acceleration lane under these three control methods. Observations indicate that in the outside lane, velocity fluctuations start from two kilometers and extend downstream. Although OPT_SEQ_NLC produces fewer fluctuations in the outside lane compared to FIFO_NLC, it leads to larger fluctuations in on-ramp. Importantly, the proposed model significantly reduces velocity variance in both the outside and on-ramp lanes.

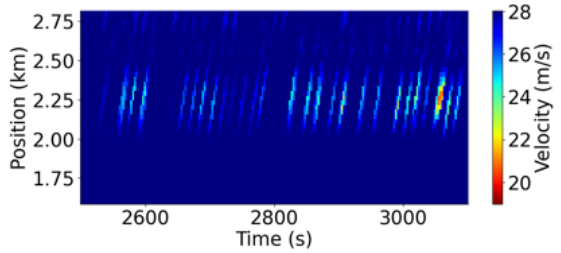
In summary, the benefits of the proposed model's two features, i.e., sequencing optimization and lane change optimization, are validated by comparing its total and average lane delays with those obtained from approaches without lane changes.



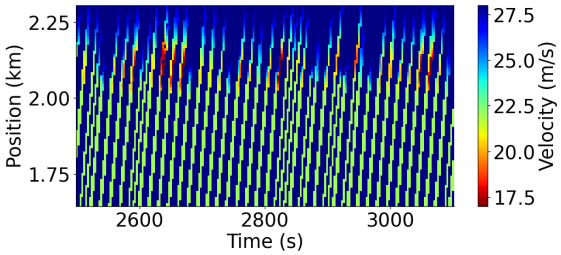
(a) FIFO_NLC; outside lane



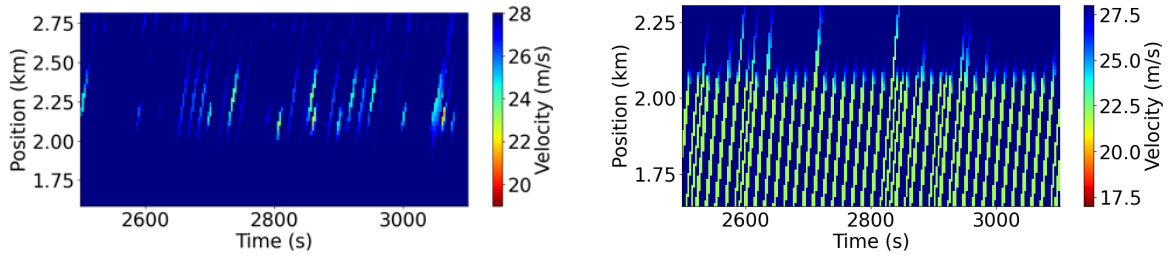
(b) FIFO_NLC; ramp and acceleration lane



(c) OPT_SEQ_NLC; outside lane



(d) OPT_SEQ_NLC; ramp and acceleration lane



(e) Proposed model; outside lane (f) Proposed model; ramp and acceleration lane

Figure 4.6. The heatmap of average CAV speed.

Average speed in different lanes by FIFO_NLC, OPT_SEQ_NLC, and Proposed model, respectively. (a), (c), (e): average CAV speed in the outside lane by the three methods; (b), (d), (f): average CAV speed in the ramp and acceleration lane by the three methods.

(5) Comparative Analysis: Utilization of Microscopic Vehicle Information

To evaluate the third characteristic of the proposed model, i.e., utilization of microscopic position and speed information, this study further compares the proposed model with the balanced flow model. The balanced flow model determines lane-changing volumes based on the macroscopic lane demand parity.

The results presented in Table 4.5 show that values in columns D^{P}, $\overline{D}_{l_1}^{\mathrm{P}}$, and $\overline{D}_{l_2}^{\mathrm{P}}$ are consistently lower than those in columns D^{B}, $\overline{D}_{l_1}^{\mathrm{B}}$, and $\overline{D}_{l_2}^{\mathrm{B}}$, which indicates that the proposed model significantly reduces total delays, average ramp delays, and average outside-lane delays compared to the balanced flow model. Notably, the most significant reduction in the total delay observed is 35%, as shown in column D. I. Nevertheless, the improvement in delay becomes 0 when the demand in lane 3 reaches 1,900 vehicles per hour, exceeding its capacity. In this case, the numbers of lane changes (values in columns $\mathrm{No.lc}^{\mathrm{P}}$ and $\mathrm{No.lc}^{\mathrm{B}}$) decided from both the macroscopic balanced flow perspective and the microscopic vehicle information perspective are zero, thereby no difference in travel delays.

Meanwhile, values in column $\mathrm{No.lc}^{\mathrm{P}}$ are higher than those in column $\mathrm{No.lc}^{\mathrm{B}}$, suggesting that the proposed model facilitates more lane changes compared to the balanced flow model. This enhancement enables the proposed model to better exploit lane-changing opportunities, thereby outperforming the balanced flow model.

The balanced flow model determines the number of lane changes (i.e., $\mathrm{No.lc}^{\mathrm{B}}$) based on the difference in the number of vehicles between the inside and outside lanes. In the scenarios tested, the maximum vehicle number difference in the inside and outside lanes is 100, resulting in a peak of approximately 50 lane changes. When there is almost no difference in the vehicle numbers (i.e., when demands in the inside and outside lanes are nearly identical), the number of lane changes drops to fewer than 10. On the other hand, the proposed model dynamically adjusts lane-changing decisions by leveraging microscopic vehicle position and speed information, aiming to minimize the total travel delay rather than merely equalizing traffic flow between lanes.

In summary, by incorporating microscopic vehicle spacing and speed data into the decision-making process, our proposed model not only reduces both total and average lane delays, but also outperforms methods that rely solely on macroscopic flow statistics. This demonstrates the efficiency of the proposed approach in enhancing traffic management through more precise and adaptive usage of lane changes.

Table 4.5. Comparison of travel delays by the balanced flow model and proposed model

Demand ratio Lane 2 / Lane 1	Demand of Lanes 1&2 (veh/hour)	Demand of Lane 3 (veh/hour)	Balanced flow model D^{B} (s)	$\bar{D}_{l_1}^{\mathrm{B}}$ (s)	$\bar{D}_{l_2}^{\mathrm{B}}$ (s)	$\bar{D}_{l_3}^{\mathrm{B}}$ (s)	No. lc^{B}	Proposed model D^{P} (s)	D. I.	$\bar{D}_{l_1}^{\mathrm{P}}$ (s)	$\bar{D}_{l_2}^{\mathrm{P}}$ (s)	$\bar{D}_{l_3}^{\mathrm{P}}$ (s)	No. lc^{P}
3:1	1,400	1,300	634	1.01	0.22	0.04	54	**419**	34%	0.43	0.13	0.10	142
		1,400	742	1.25	0.28	0.01	6	**480**	35%	0.48	0.17	0.09	114
		1,500	770	1.32	0.30	0.00	0	**568**	26%	0.58	0.18	0.12	102
	1,600	1,500	918	1.14	0.31	0.06	53	**748**	19%	0.62	0.21	0.16	127
		1,600	1,080	1.49	0.40	0.01	5	**838**	22%	0.68	0.24	0.18	110
		1,700	1,099	1.47	0.43	0.00	0	**971**	12%	0.90	0.29	0.15	75
	1,800	1,700	1,608	1.58	0.52	0.12	51	**1,404**	13%	1.15	0.42	0.19	109
		1,800	1,770	2.02	0.63	0.01	2	**1,696**	4%	1.35	0.48	0.20	32
		1,900	1,776	2.05	0.63	0.00	0	**1,776**	0%	2.05	0.63	0.00	0
4:1	1,400	1,300	537	1.00	0.19	0.03	52	**396**	26%	0.58	0.12	0.08	107
		1,400	669	1.37	0.25	0.004	5	**436**	35%	0.57	0.13	0.10	100
		1,500	688	1.41	0.26	0.00	0	**453**	34%	0.60	0.13	0.10	95
	1,600	1,500	854	1.31	0.26	0.07	52	**670**	22%	0.64	0.18	0.16	118
		1,600	1,036	1.83	0.34	0.01	3	**781**	25%	0.76	0.21	0.17	98
		1,700	1,043	1.86	0.35	0.00	0	**874**	16%	0.99	0.24	0.14	69
	1,800	1,700	1,501	1.62	0.51	0.11	48	**1,290**	14%	0.96	0.37	0.24	104
		1,800	1,689	2.18	0.63	0.00	0	**1,557**	8%	1.61	0.51	0.13	44
		1,900	1,689	2.18	0.63	0.00	0	**1,689**	0%	2.18	0.63	0.00	0

Notes: (1) the superscripts B and P represent the balanced flow model and the proposed model, respectively; the subscripts l_1, l_2, and l_3 refer to the on-ramp, outside lane, and inside lane, respectively; (2) D denotes the total delay in seconds; (3) $\bar{D}_{l_1}$, $\bar{D}_{l_2}$, and $\bar{D}_{l_3}$ represent the average lane delays of the related lanes; (3) D. I. refers to the delay improvement ratio and can be calculated by $(D^B - D^P)/D^B$; (4) No. lc refers to number of lane changes; (5) "vehicles per hour" is abbreviated as "veh/hour".

(6) Vehicle Trajectory Analysis

To fully understand under what conditions the proposed model decides certain outside-lane CAVs to change to the inside lane, and instruct which CAVs to do so, this study conducts two case studies and analyzes the results of the proposed model: sequencing as well as lane-changing optimization, and the detailed trajectories.

Case study 1

The first case study examines the interactions among two inside-lane CAVs, two outside-lane CAVs, and one on-ramp CAV, with their initial states and results of this case study presented in Table 4.6. Figure 4.7 illustrates the longitudinal and lateral position trajectories of these five CAVs at different time points, obtained by solving the proposed model. Circled numbers in Figure 4.7 indicate specific time moments, and number 1 represents the beginning time moment. As detailed in Table 4.6 and depicted in Figure 4.7, the on-ramp CAV is assigned to merge ahead of outside-lane CAV 1, and outside-lane CAV 2 is assigned as the lane-changing vehicle to follow the two inside-lane CAVs.

Table 4.6. Initial vehicle states and output of the proposed model in case study 1.

	Inside-lane CAV 1	Inside-lane CAV 2	Outside-lane CAV 1	Outside-lane CAV 2	On-ramp CAV
Initial position (m)	630	590	590	550	600
Initial velocity (m/s)	28	27	28	27	26
Lane change decisions	-	-	No	Yes	-
Inside-lane sequence decisions	1st	2nd	-	3rd	-
Outside-lane sequence decisions	-	-	2nd	-	1st
Planned driving time (s)	12.82	14.53	15.62	16.24	13.91

Note: - means not applicable.

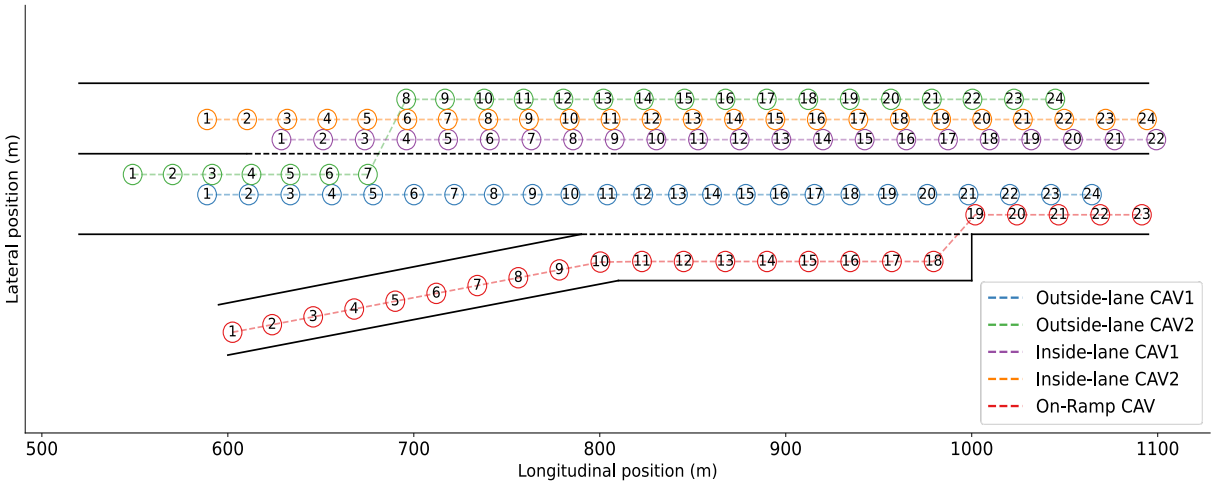
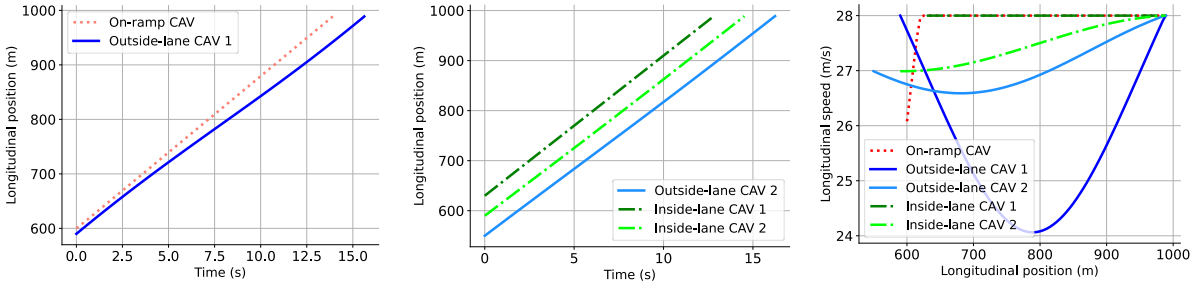


Figure 4.7. Lateral and longitudinal positions of CAVs in case study 1.

The reasons behind the above decisions are detailed as follows. Initially, the on-ramp CAV is already positioned some distance ahead of outside-lane CAV 1, with both CAVs driving at similar speeds. This observation indicates that the on-ramp CAV can move through the merging section earlier, thereby allowing the following outside-lane CAVs to pass earlier as well. Therefore, having the on-ramp CAV merge in front of the two outside-lane CAVs can save their total travel time. After making this merging decision, outside-lane CAV 1 opts to remain in its original lane to facilitate the merging of the on-ramp vehicle by creating the necessary gape. This decision to maintain its lane is driven by the fact that outside-lane CAV 1 is behind the on-ramp CAV and aligned longitudinally with inside-lane CAV 2. If outside-lane CAV 1 changes lanes, it will decelerate more to create a safe gap with inside-lane CAV 2. Outside-lane CAV 2 is then determined to change to the inside lane, following inside-lane CAV 2 to avoid the speed fluctuations from its preceding outside-lane CAV 1.

To illustrate the process of creating the required spacing among the merging CAVs, the lane-changing CAVs, and their facilitating CAVs, Figures 4.8(a) and 4.8(b) display the longitudinal position and time curves of CAVs in the outside and inside lanes, respectively. Specifically, Figure 4.8(a) shows the formation of the time headway between the on-ramp CAV and outside-lane CAV 1; Figure 4.8(b) depicts the positional relationships between lane-changing CAV (outside-lane CAV 2) and the targeting preceding and facilitating CAVs in the inside lane. A required time headway of 1.5 seconds is achieved at the end of the merging section (positioned at 1,000 meters). We can see that at beginning, there is a subtle time headway between the on-ramp CAV and outside-lane CAV 1, as shown in Figure 4.8(a), but

a clear time headway between inside-lane CAV 2 and outside-lane CAV 2, as shown in Figure 4.8(b). This suggests that outside-lane CAV 1 must undergo a greater speed reduction than outside-lane CAV 2 to create the necessary time headway. Correspondingly, Figure 4.8(c) depicts the vehicle speeds along their longitudinal positions, highlighting that while the on-ramp CAV directly accelerates to free-flow speed due to its priority, outside-lane CAV 1 decelerates significantly (around 4 m/s) to help the on-ramp CAV merge. Then, outside-lane CAV 2 also decelerates slightly, i.e., approximately 0.5 m/s, to ensure a safe time headway with its target preceding CAV, i.e., inside-lane CAV 1.



(a) (b) (c)

Figure 4.8. Longitudinal position and velocity trajectories of CAVs in case study 1. (a) Time and longitudinal position trajectories of the on-ramp and facilitating outside-lane CAVs; (b) Time and longitudinal position trajectories of the lane-changing and inside-lane CAVs; (c) Longitudinal position and speed trajectories of all CAVs.

Case study 2

The second case study presents different decisions for the five CAVs with slightly different initial states, as summarized in Table 4.7. A key difference from the first case study is the initial position of inside-lane CAV 2, which is at 550 meters instead of 590 meters. This adjustment creates a free gap between the two inside-lane CAVs initially, thereby enabling outside-lane CAV 1 to change lanes to avoid the influence of the on-ramp vehicle.

Table 4.7. Initial vehicle states and output of the proposed model in case study 2.

	Inside-lane CAV 1	Inside-lane CAV 2	Outside-lane CAV 1	Outside-lane CAV 2	On-ramp CAV
Initial position (m)	630	550	590	550	600
Initial velocity (m/s)	28	27	28	27	26
Lane change decisions	-	-	Yes	No	-
Inside-lane sequence decisions	1st	3rd	2nd	-	-
Outside-lane sequence decisions	-	-	-	2nd	1st
Planned driving time (s)	12.82	16.24	14.53	15.69	13.91

Note: - means not applicable.

As detailed in Table 4.7 and illustrated in Figure 4.9, the on-ramp CAV still drives through the merging section before both outside-lane CAVs. Unlike the decisions in Case study 1, outside-lane CAV 1 decides to move to the inside lane to utilize the available gap there, consequently leaving a substantial gap in the outside lane. This strategic lane-changing maneuver allows outside-lane CAV 2 to remain in

the outside lane without being affected by the on-ramp vehicle. Figures 4.10(a), 4.10(b), and 4.10(c) demonstrate that with the help of the proposed model, the on-ramp CAV can directly accelerate to merge, and outside-lane CAV 2 remains unaffected. The lane change maneuver executed by outside-lane CAV 1 results in a minor speed reduction of 0.8m/s for itself and a speed decrease of 0.4 m/s for inside-lane CAV 2.

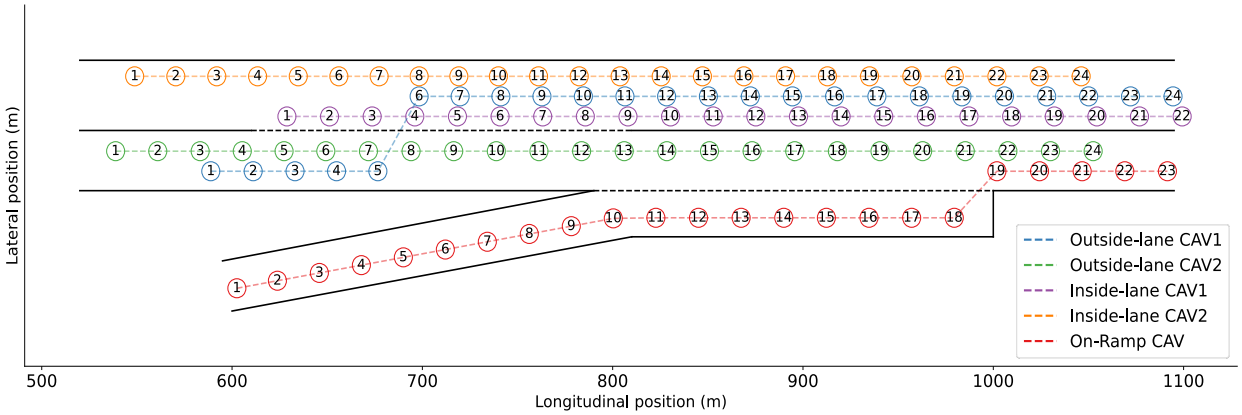
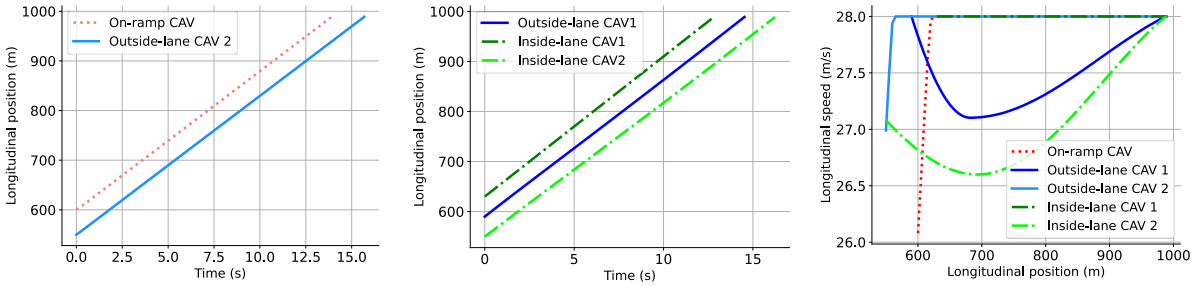


Figure 4.9. Lateral and longitudinal positions of CAVs in case study 2.



(a) (b) (c)

Figure 4.10. Longitudinal position and velocity trajectories of CAVs in case study 2.

(a) Time and longitudinal position trajectories of the on-ramp and facilitating outside-lane CAVs; (b) Time and longitudinal position trajectories of the lane-changing and inside-lane CAVs; (c) Longitudinal position and speed trajectories of all CAVs.

(7) Sensitivity analyses

The lengths of the cooperative lane-changing area (L_{lc}), the merging area (L_{merge}), and the lane-changing cost (w_{lc}) are three important predefined parameters. To further investigate the proposed model, sensitivity analyses are conducted on the above mentioned three parameters under the on-ramp and outside mainline lane demand at 1,800 vehicles/hour and the inside mainline lane demand at 1,700 vehicles/hour.

Figures 4.11(a), 4.11(b), and 4.11(c) illustrate the total travel delay and the number of lane changes for different values of L_{lc}, L_{merge}, and w_{lc}, respectively. Specifically, Figure 4.11(a) highlights the positive effect of L_{lc} on travel delay. As L_{lc} increases, the total travel delay decreases significantly, dropping from 1,600 seconds to 1,400 seconds. Meanwhile, the number of lane changes increases greatly, from 30 to 90, until reaching a L_{lc} threshold at 350 meters, beyond which the number of lane

changes stabilizes. Figure 4.11(b) demonstrates the beneficial influence of L_{merge} on travel delay, with the total delay reducing sharply up to $L_{\text{merge}} = 200$ meters. Beyond 200 meters, the total travel delay tends to a stable value. Also, L_{merge} almost has no influence on the number of lane changes which ranges from 103 to 105. Hence, we can conclude that the number of lane changes is primarily influenced by L_{lc}. Finally, Figure 4.11(c) explores the impact of w_{lc}, showing that as the cost of changing lanes increases, lane-changing maneuvers that only yield minor efficiency improvements are no longer adopted, consequently leading to an increase in total travel delay. Overall, the results suggest that increasing the values of L_{lc} and L_{merge}, while decreasing the value of w_{lc}, generally leads to reduced travel delays. Meanwhile, the number of lane changes is directly affected by L_{lc} and indirectly by w_{lc}.

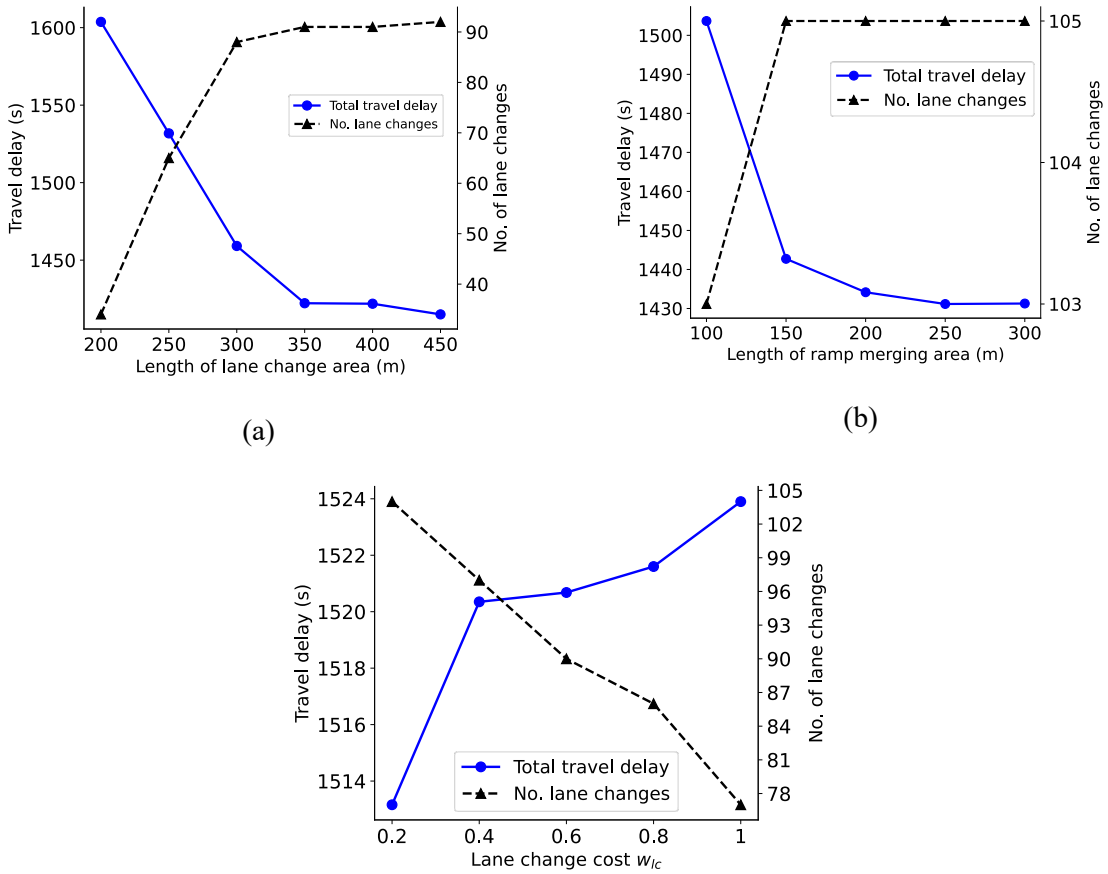
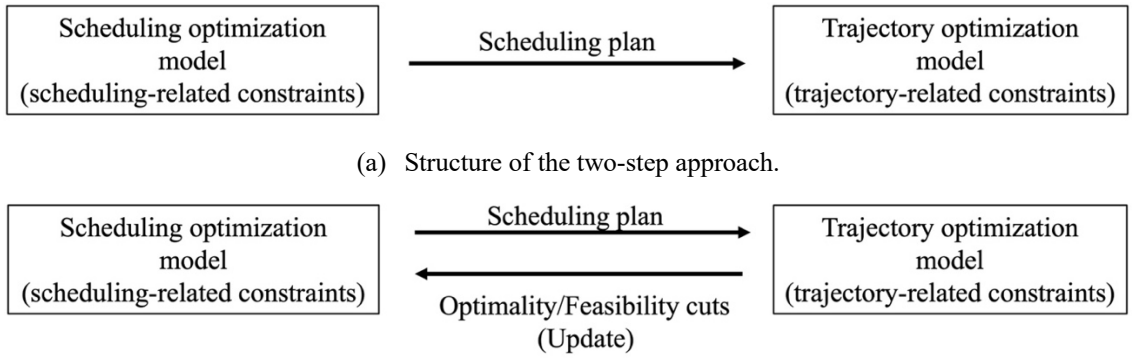


Figure 4.11. Sensitivity analysis

(a) Impact of L_{lc} on travel delay and lane changes; (b) Impact of L_{merge} on travel delay and lane changes; (c) Impact of w_{lc} on travel delay and lane changes.

(8) Benefits of integration

In this paper, we develop an original approach that integrates scheduling (i.e., lane selection and vehicle sequencing) optimization and trajectory optimization. In this section, we compare the proposed integrated approach with a two-step approach that separately solves the scheduling optimization and trajectory optimization, in consecutive steps. Figure 4.12 illustrates the differences between the two approaches. Essentially, the integrated approach iteratively solves the problem multiple times, whereas the two-step approach performs the optimization process only once.



(a) Structure of the two-step approach.

(b) Structure of the integrated approach.

Figure 4.12. Structures of the two-step approach and the integrated approach

Table 4.8. Comparison of travel delays by the two-step approach and proposed integrated approach

Demand ratio Lane 2 / Lane 1	Demand of Lanes 1&2 (veh/hour)	Demand of Lane 3 (veh/hour)	Two-step approach					Proposed integrated approach					
			D^{T} (s)	$\overline{D}_{l_1}^{\mathrm{T}}$ (s)	$\overline{D}_{l_2}^{\mathrm{T}}$ (s)	$\overline{D}_{l_3}^{\mathrm{T}}$ (s)	No. lc$^{\mathrm{T}}$	D^{P} (s)	D. I.	$\overline{D}_{l_1}^{\mathrm{P}}$ (s)	$\overline{D}_{l_2}^{\mathrm{P}}$ (s)	$\overline{D}_{l_3}^{\mathrm{P}}$ (s)	No. lc$^{\mathrm{P}}$
3:1	1,400	1,300	566	0.72	0.21	0.07	90	**496**	12%	0.65	0.18	0.06	93
		1,400	629	0.76	0.23	0.08	77	**553**	12%	0.70	0.21	0.06	80
		1,500	668	0.84	0.26	0.07	60	**602**	10%	0.80	0.23	0.05	62
	1,600	1,500	900	0.84	0.31	0.13	109	**795**	12%	0.78	0.27	0.11	115
		1,600	994	0.89	0.36	0.13	86	**889**	11%	0.83	0.31	0.11	95
		1,700	1,093	0.98	0.40	0.13	66	**988**	10%	0.93	0.36	0.11	74
	1,800	1,700	1,460	1.06	0.52	0.17	96	**1,330**	9%	1.04	0.46	0.14	101
		1,800	1,683	1.25	0.61	0.17	66	**1,556**	8%	1.21	0.54	0.16	75
		1,900	1,792	1.56	0.70	0.07	18	**1,739**	3%	1.55	0.65	0.09	28
	2,000	1,900	2739	1.91	1.09	0.08	19	**2,648**	3%	2.06	0.96	0.09	30
		2,000	2784	2.03	1.18	0.00	1	**2,747**	1%	2.29	1.05	0.01	3
		2,100	2783	2.09	1.16	0.00	1	**2,769**	1%	2.34	1.06	0.00	1
4:1	1,400	1,300	492	0.73	0.18	0.07	82	**431**	12%	0.66	0.16	0.05	83
		1,400	530	0.77	0.19	0.07	68	**471**	11%	0.73	0.16	0.06	70
		1,500	555	0.81	0.21	0.06	59	**497**	11%	0.77	0.19	0.05	61
	1,600	1,500	772	0.85	0.25	0.12	94	**682**	12%	0.76	0.22	0.10	100
		1,600	858	1.00	0.27	0.12	79	**766**	11%	0.90	0.23	0.11	88
		1,700	936	1.02	0.32	0.12	64	**846**	10%	0.99	0.27	0.11	69
	1,800	1,700	1,323	1.00	0.43	0.20	102	**1,187**	10%	0.96	0.38	0.17	109
		1,800	1,522	1.35	0.53	0.15	60	**1,397**	8%	1.32	0.46	0.14	69
		1,900	1,670	1.69	0.67	0.05	14	**1,630**	2%	1.72	0.60	0.08	24
	2,000	1,900	2,586	1.88	1.03	0.10	25	**2,465**	5%	2.00	0.92	0.10	33
		2,000	2,648	2.11	1.13	0.00	1	**2,616**	1%	2.31	1.04	0.02	3
		2,100	2,646	2.12	1.12	0.00	1	**2,631**	1%	2.35	1.05	0.00	1

Notes: (1) the superscripts T and P represent the two-step approach and the proposed integrated approach, respectively; the subscripts l_1, l_2, and l_3 refer to the on-ramp, outside lane, and inside lane, respectively; (2) D denotes the total delay in seconds; (3) $\overline{D}_{l_1}$, $\overline{D}_{l_2}$, and $\overline{D}_{l_3}$ represent the average lane delays of the related lanes; (3) D. I. refers to the delay improvement ratio and can be calculated by $(D^T - D^P)/D^T$; (4) No. lc refers to number of lane changes; (5) "vehicles per hour" is abbreviated as "veh/hour

Table 4.8 and Table 4.9 record comparison results between the integrated model and the two-step model in terms of traffic performance, and computational efficacy, respectively. From Table 7, we can see that, compared to the two-step model, the integrated model can reduce the total delay by up to 12%. Moreover, when comparing the delays of outside-lane vehicles (based on the values in columns $\overline{D}_{l_2}^{\mathrm{T}}$ and

$\overline{D}_{l_2}^{\mathrm{P}}$), the integrated model achieves an obvious delay reduction for outside-lane vehicles, showing its effectiveness in mitigating disturbances caused by on-ramp vehicles to mainline traffic. Additionally, the integrated model exhibits more lane changes (see the values in columns $\mathrm{No.lc^T}$ and $\mathrm{No.lc^P}$) while maintaining similar or even lower delays for inside-lane vehicles (based on the values in columns $\overline{D}_{l_3}^{\mathrm{T}}$ and $\overline{D}_{l_3}^{\mathrm{P}}$). This result indicates that the integrated model can effectively identify more advantageous lane-changing opportunities. On the other hand, as shown in Table 4.8, which records the computation time for both methods under different number of vehicles, ranging from three to ten vehicles (each contains ten random computational instances). While the two-step method requires less computation time, both methods complete the computation within approximately one second, with no big difference. This is because the integrated model only takes a few iterations to obtain a solution. Therefore, the computation time for the integrated model is considered acceptable.

Table 4.9. Comparison of computation time by the two-step model and proposed integrated model

Number of CAVs		3	4	5	6	7	8	9	10
Mean value of computation time (second)	Two-step model	0.18	0.25	0.33	0.42	0.51	0.65	0.75	0.97
	Integrated model	0.28	0.39	0.56	0.68	0.83	0.91	1.06	1.52

4.4 Summary

This chapter investigates the enhancement of traffic performance through the joint optimization of lane assignment, vehicle sequences, and trajectories in multilane freeway merging scenarios. To this end, we formulate the problem of merging on-ramp CAV flows into multiple mainline CAV flows as an MINLP model, which incorporates spatiotemporal relationships between vehicles, vehicle kinematics, and road geometry, with the aim of maximizing traffic efficiency, avoiding unnecessary lane changes, and generating the smoothest vehicle trajectories. Here, note that lane-changing and merging time points are determined optimally by the model, thereby achieving superior performance compared to those heuristically predefined methods. Furthermore, the feature of joint determination of trajectories and scheduling decisions (lane changes and sequences) ensures that scheduling solutions are contingent on the existence of feasible trajectories, and the costs associated with these trajectories can influence scheduling decision.

To efficiently solve the proposed MINLP model, we develop a GBD-based solution algorithm. This approach subtly decomposes the model into a relaxed master problem and a primal subproblem. This decomposition allows for a more efficient handling by standard solvers and significantly accelerates the computation by reducing the number of variables and constraints required, compared with solving the original problem directly. Furthermore, a feasibility-guaranteed primal subproblem is developed by introducing additional elastic slack variables and penalizing these slacks in the objective function to enhance the efficiency of the GBD-based solution algorithm in this study. The effectiveness and efficiency of the proposed algorithm are validated through a comparison with the state-of-the-art solver Gurobi.

Demonstrations under different traffic conditions show notable enhancements in traffic efficiency. The model's benefits, e.g., vehicle sequencing, maximizing multilane utilization, and leveraging microscopic vehicle information, are highlighted through comparative analyses with three baseline models. In addition, our findings suggest that the most time-efficient setting includes a cooperative lane-changing area of 300-to-350 meters and a cooperative merging area of 150-to-200 meter.

Chapter 5

Learning Lane Selection and Driving Orders for Multi-Lane Freeway Merging

This chapter also addresses the multi-lane freeway merging problem and proposes a hybrid bi-level control approach that integrates deep reinforcement learning with optimization modelling. The vehicle sequences, lane selections, and trajectories are optimized to minimize total travel delay. Specifically, the upper level serves as the scheduling planner, where we design an attention-based neural network to make decisions on target lanes and right-of-way. The lower level employs a nonlinear model predictive controller to continuously update trajectories, ensuring vehicles reach the designated lanes and follow the planned sequence. The chapter is organized as follows. Section 5.1 mathematically describes the cooperative merging of mainline and on-ramp traffic streams at a multi-lane freeway on-ramp merging section; Section 5.2 introduces the proposed hybrid bi-level controller; Section 5.3 introduces the upper-level learning-based scheduling planner; Section 5.4 introduces the low-level optimization-based trajectory planner; Section 5.5 validates the proposed method by numerical experiments; Section 5.5 concludes this chapter.

5.1 Problem Description

We first introduce the multi-lane freeway merging scenario considered in this chapter and then formulate a general mathematical programming model to describe the problem.

(1) Multi-lane Freeway Merging Scenario

This chapter also consider a typical freeway on-ramp merging section, which includes two mainline lanes (an inside lane and an outside lane) and an on-ramp lane that extends into an acceleration lane, as shown in Fig. 5.1.

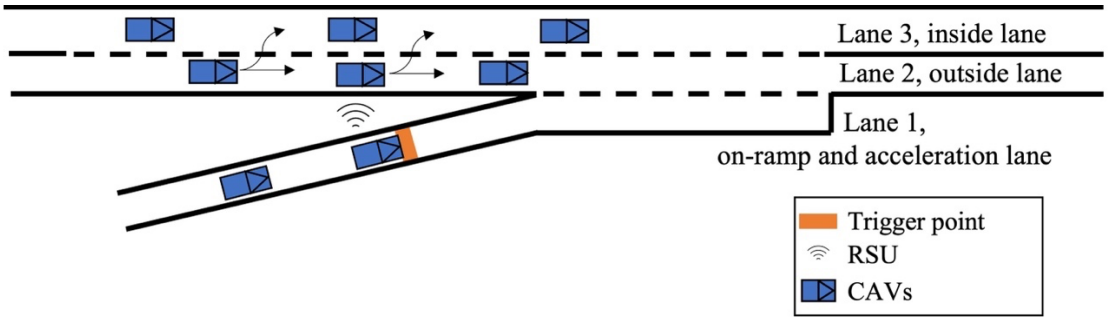


Figure 5.1. Multi-lane freeway merging scenario

This merging section also includes a roadside unit (RSU) and a trigger point. Specifically, the RSU is positioned upstream of the merge gore, serving to collect CAV information and transmit commands to CAVs; the trigger point is located on the on-ramp lane to activate the controller when an on-ramp vehicle approaches. Before any vehicle approaches the trigger point, all CAVs operate in a car-following mode. Once an on-ramp vehicle passes the trigger point, the RSU initiates a control cycle. Within each control cycle, a planned CAV group is formed by grouping together the on-ramp vehicle passing the trigger point, the following on-ramp vehicles and nearby mainline vehicles. This CAV group no longer operates in the car-following mode but instead follows the controller's instructions to adjust speed and perform the lane-changing and merging manoeuvres. Upon completing commands, these CAVs revert to the car-following mode.

(2) Mathematical Problem Description

Controlling multiple CAV streams includes three tasks: lane selection, vehicle sequencing, and trajectory planning. Lane selection and vehicle sequencing are responsible for determining target lanes and passing orders for CAVs that drive through the merging section. Trajectory planning is responsible for generating trajectories from their initial positions to the end of the merging section, ensuring that CAVs reach selected lanes, follow assigned sequences, and keep safe throughout the process.

These tasks are interrelated in the sense that lane selection and vehicle sequences influence the trajectory design. Meanwhile, trajectory cost, in turn, affects the determination of the optimal lane selection and vehicle sequences.

To formally describe this complex decision problem, we introduce some notations in Table 5.1.

Let $L \coloneqq 1,2,3$ denote the set of lanes, corresponding to the on-ramp, outside lane, and inside lane, respectively; let $I \coloneqq 1,2,\dots,|I|$ represent a group of $|I|$ planned CAVs. The variables $p_{i,t}$, $v_{i,t}$, and $u_{i,t}$ respectively denote the longitudinal position, speed, and acceleration of vehicle $i \in I$ at time t, where longitudinal positions refer to the positions along each lane.

Table 5.1. Notations in the general model.

Indices and Sets	
L	set of lanes, $l \in L$.
I	set of all CAVs, $i, j \in I$.
Input Parameters	
$p_{i,init}$	initial position of vehicle $i \in I$.
$v_{i,init}$	initial velocity of vehicle $i \in I$.
p_{final}	end position of acceleration lane.
Decision Variables	
$\gamma_{i,j}$	binary, equals 1 if CAV $i \in I$ is assigned to lane $l \in L \setminus \{1\}$, 0 otherwise.
$\delta_{i,j}$	binary, equals 1 if CAVs i and j ($i \in I, j \in I, i < j$) are assigned to the same lane, 0 otherwise.
$\alpha_{i,j}$	binary, equals 1 if CAV $i \in I$ is assigned in front of CAV j ($j \in I, j > i$), and both are assigned to the same lane, 0 otherwise.
T_i	continuous, travel time for CAV $i \in I$.
$p_{i,t}$	continuous, the longitudinal position of CAV $i \in I$ at time t.
$v_{i,t}$	continuous, the longitudinal velocity of CAV $i \in I$ at time t.
$u_{i,t}$	continuous, the control input of CAV $i \in I$ at time t.

According to the notation introduced, the mixed integer nonlinear programming (MINLP) model

is formulated as follows:

$$\min \sum_{i \in I} T_i \tag{5.1}$$

subject to:

$$\sum_{l \in L \setminus \{1\}} \gamma_{i,l} = 1 \qquad \forall i \in I \tag{5.2}$$

$$\sum_{l \in L\{1\}} |\gamma_{i,l} - \gamma_{j,l}| = 1 - \delta_{i,j} \qquad \forall i \in I \tag{5.3}$$

$$\alpha_{i,j} + \alpha_{j,i} = \delta_{i,j} \qquad \forall i,j \in I \tag{5.4}$$

$$p_{i,0} = p_{i,\text{init}} \qquad \forall i \in I \tag{5.5}$$

$$v_{i,0} = v_{i,\text{init}} \qquad \forall i \in I \tag{5.6}$$

$$p_{i,T_i} \leq p_{\text{final}} \qquad \forall i \in I \tag{5.7}$$

$$\delta_{i,j}\left[\left(2\alpha_{i,j} - 1\right)\left(p_{i,T_i} - p_{j,T_i}\right) - h\right] \geq 0 \qquad \forall i,j \in I \tag{5.8}$$

$$O\left(p_{i,t}\right) \cap O\left(p_{j,t}\right) = \emptyset \qquad \forall i,j \in I, t \leq T_i \tag{5.9}$$

$$v_{i,t+1} = v_{i,t} + u_{i,t}\Delta t \qquad \forall i \in I, t \leq T_i - 1 \tag{5.10}$$

$$p_{i,t+1} = p_{i,t} + \frac{v_{i,t} + v_{i,t+1}}{2}\Delta t \qquad \forall i \in I, t \leq T_i - 1 \tag{5.11}$$

$$\gamma_{i,l} \in \{0,1\} \qquad \forall i,j \in I \tag{5.12}$$

$$\delta_{i,j} \in \{0,1\} \qquad \forall i,j \in I \tag{5.13}$$

$$\alpha_{i,j} \in \{0,1\} \qquad \forall i,j \in I \tag{5.14}$$

$$u_{i,k} \in \mathcal{U} \qquad \forall i \in I, t < T_i \tag{5.15}$$

$$v_{i,k} \in \mathcal{V} \qquad \forall i \in I, t < T_i \tag{5.16}$$

$$p_{i,k} \in \mathcal{P} \qquad \forall i \in I, t < T_i \tag{5.17}$$

1) Objective Function: The goal is to minimize total travel time of a CAV group.

2) Combinatorial Constraints: Constraints 2–4 ensure the orderly distribution of traffic flow on the mainline lanes. Specifically, constraints 1 state that each CAV must select a target lane from the mainline lanes. Constraints 2 and 3 indicate that if any two CAVs select the same target lane, they need to establish a unique driving sequence. Additionally, constraints 4–6 state the binary nature of related variables.

3) Trajectory Constraints: Constraints 5–11 define the vehicle's motion from its initial state to the final state that completes merging and lane-changing manoeuvres. Constraints 5 and 6 specify the initial position and speed conditions of the vehicles. Constraints 7 and 8 outline the terminal conditions. Specifically, constraints 7 require that the merging and lane changes must be completed before the end of the merging section. Constraints 8 ensure that the timing of vehicles driving through the end of the merging section is consistent with the decision variables for lane selection and vehicle orders. Constraints 9 guarantee no collisions at any time step, in which $O\left(x_{i,t}\right)$ represents the space occupied by vehicle i at time t. Constraints 10 and 11 describe the kinematic equations. Finally, constraints 15–17 define the permissible ranges for control inputs, speeds, and positions, respectively.

In summary, CAV-based merging control involves strategic scheduling decisions aimed at minimizing total travel time, along with multi-vehicle trajectory planning that ensures safe movement through the merging section while adhering to specified target lanes and vehicle sequences.

5.2 Bi-level Control Framework

To solve the above mentioned MINLP model in real time, we design a hybrid bi-level control framework that combines the rapid inference capabilities of neural networks with the safety guarantees provided by mathematical models, as depicted in Figure 5.2. The upper-level module handles scheduling-related decisions, and the lower-level module determines trajectory-related variables.

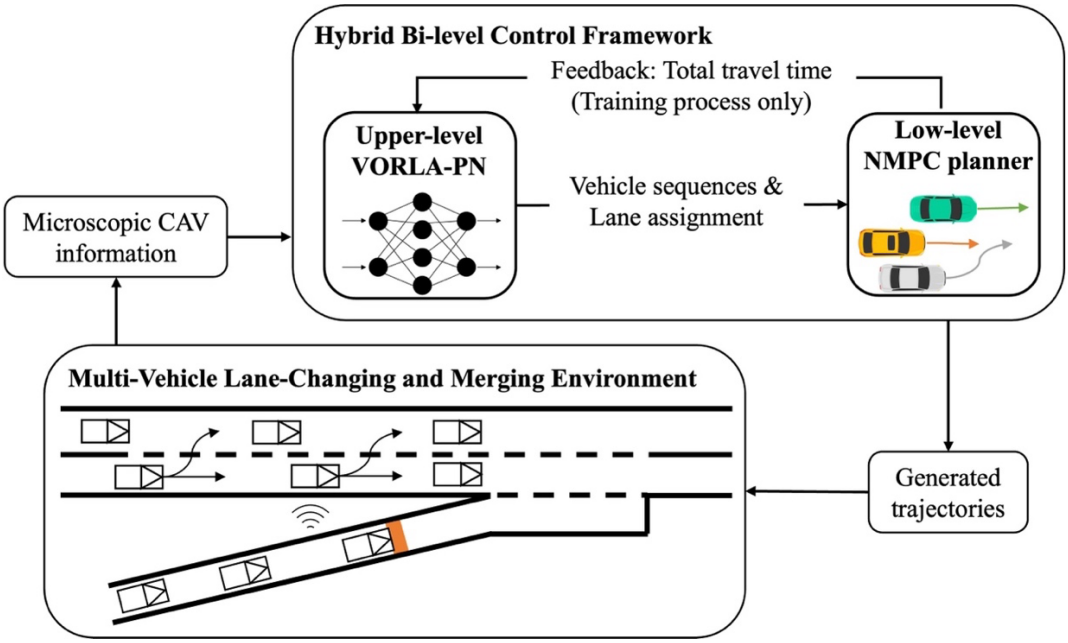


Figure 5.2. Hybrid bi-level control framework.

In the upper-level scheduling problem, we introduce a novel **V**ehicle **Or**dering and **L**ane **A**ssignment **P**olicy **N**etwork (VORLA-PN) to determine passing orders and lane selections for a fleet of CAVs, given the position, speed, and lane information of a CAV group.

In the lower-level trajectory planning problem, a nonlinear model predictive control (NMPC) planner is developed to generate collision-free trajectories, in compliance with the sequence and lane decisions specified by the VORLA-PN. Also, it provides total travel delay to guide the learning process of VORLA-PN towards improving overall traffic efficiency.

5.3 VORLA Policy Network

(1) Markov Decision Process Formulation

We formulate the problem of determining vehicle orders and lane assignments as an $|I|$-step Markov Decision Process (MDP), denoted as a tuple $\langle I,\ S,\ A,\ R,\ T\rangle$, where I denote the set of CAVs, S the state space, A the action space, R the reward function, and T the state transition function. The key MDP elements are defined as follows:

State: The state at each decision step k, defined as $s_k = \{\boldsymbol{X}, \boldsymbol{N}_k) \in S$, comprises two parts: the initial CAV states $\boldsymbol{X}$ and the dynamic occupation vector $\boldsymbol{N}_k$. $X = \left[X^{(1)}, \dots, X^{(|I|)}\right]^T$ contains all CAVs' information at the moment the trigger point is activated, where $X^{(i)} = \left[p^{(i)}, v^{(i)}, \ell^{(i)}\right]^T$ captures the longitudinal position, velocity, and lane ID of CAV $i \in I$. $N_k = \left[N_k^{l_2},\ N_k^{l_3}\right]^T$ represents the occupation

status of the mainline lanes 2 and 3. Each column vector, $N_k^{l_2} and N_k^{l_3}$, is a one-hot vector of length $|I|$ consisting entirely of zeros except for a single cell. The single cell is marked as one to uniquely indicate the current occupant of the corresponding mainline lane at step k. For instance, if CAV i was assigned to lane 2 at the last decision step $k-1$, then $N_k^{l_2}$ would become $[0, \dots, 1, \dots, 0]^T$ where 1 is located in the i^{th} cell.

Action: The action is represented as $a_k = (c_k,\ l_k) \in A$, where $c_k \in C_k$ denotes the selection of one CAV from a set of available CAVs at step k, and $l_k \in L_k$ refers to the selection of a target lane from a set of permissible target mainline lanes. Thus, each decision step involves choosing a CAV and assigning it a target lane, until all CAVs are assigned. It is important to note that the sequence in which the CAVs are selected forms a "driving order".

Then, the available CAV set C_k is determined by the following two masking rules:

- Each CAV can be assigned only once during the process.
- Based on the initial order of vehicles in each lane, the leading CAV must be assigned before the subsequent CAVs.

The permissible lane set L_k is dependent on the initial lane ID of the selected vehicle.

Mainline vehicles can freely choose between the two mainline lanes (i.e., lane 1 and lane 2), whereas on-ramp vehicles are restricted to choose the outside lane (i.e., lane 2).

Then, the corresponding masking $M_k^C \in \mathbb{R}^{|I|}$ and $M_k^L \in \mathbb{R}^{2\times|I|}$ are constructed based on the rules of C_k and L_k, to prohibit invalid actions.

Reward: The objective is to minimize the total travel delay for a group of CAVs. Accordingly, the reward function includes a sparse termination reward at the final decision step, denoted by $R_{|I|} = -\sum_{i\in I} T_i$, which corresponds to the negative of the objective function (Equation 1). The rewards for all preceding decision steps are set to zero. Hence, the reward function is written as:

$$R_k = \begin{cases} 0,\ \forall k \in \{1, \dots, |I|-1\} \\ -\sum_{i\in I} T_i,\ k = |I| \end{cases} \tag{5.18}$$

State transition: Executing action a_k in state s_k results in the next state s_{k+1}, based on the state transition rules $T(s_{k+1}|s_k, a_k)$. Specifically, given the action (c_k, l_k), the corresponding column of $\boldsymbol{N}_k$ is updated, while the other column remains unchanged from the previous state. The updates are as follows:

$$N_{k+1}^l = \begin{cases} e_{c_k},\ l = l_k \\ N_k^l,\ \forall l \neq l_k \end{cases} \tag{5.19}$$

where N_{k+1}^l reflects the updated assignment of vehicles and lanes; e_{c_k} is a one-hot vector with a 1 at $the\ c_k^{\text{th}}$ position and 0s elsewhere. At each step, one column is updated because only a single CAV is assigned to one of the mainline lanes. Also, X, which includes initial positions, velocities, and lane information, stays constant. If all CAVs have been assigned, the next state is the terminal state.

(2) Sequential Decision-Making

Given initial CAV states **X**, the policy network VORLA approximates a stochastic policy π_θ, which outputs the probability of a solution $\boldsymbol{a}$. A complete solution, $\boldsymbol{a} = (a_1, \dots, a_{|I|})$, compromises a series of decisions $a_k = (c_k, l_k)$ at each step k. Here, c_k represents the selection of a vehicle, and l_k

denotes the assignment of that vehicle to a specific lane. The resulting sequence $(c_1, \ldots, c_{|I|})$ forms the driving order for $|I|$ CAVs.

The process of generating a complete a can be factorized into a chain of conditional probabilities:

$$\pi_\theta(a|X) = \prod_{k=1}^{|I|} \pi_\theta(a_k|N_{k-1}, X) \tag{5.20}$$

This factorization implies that the solution can be constructed incrementally, with the decision at each step a_k depending on the decisions from previous steps N_{k-1} and $\boldsymbol{X}$. Consequently, a sequential decision-making process is developed to generate each partial solution a_k iteratively, thereby constructing the complete solution a.

The proposed VORLA network comprises a policy network π_θ and a baseline network b_ϕ. The policy network π_θ utilizes a transformer encoder to learn the interrelationships among a group of CAVs. Additionally, it includes a decoder that is repeatedly executed during the sequential decision-making process to produce the action sequence a. The baseline network b_ϕ is employed to facilitate learning by variance reduction. Fig. 5.3 illustrates the VORLA network structure, which is explained as follows.

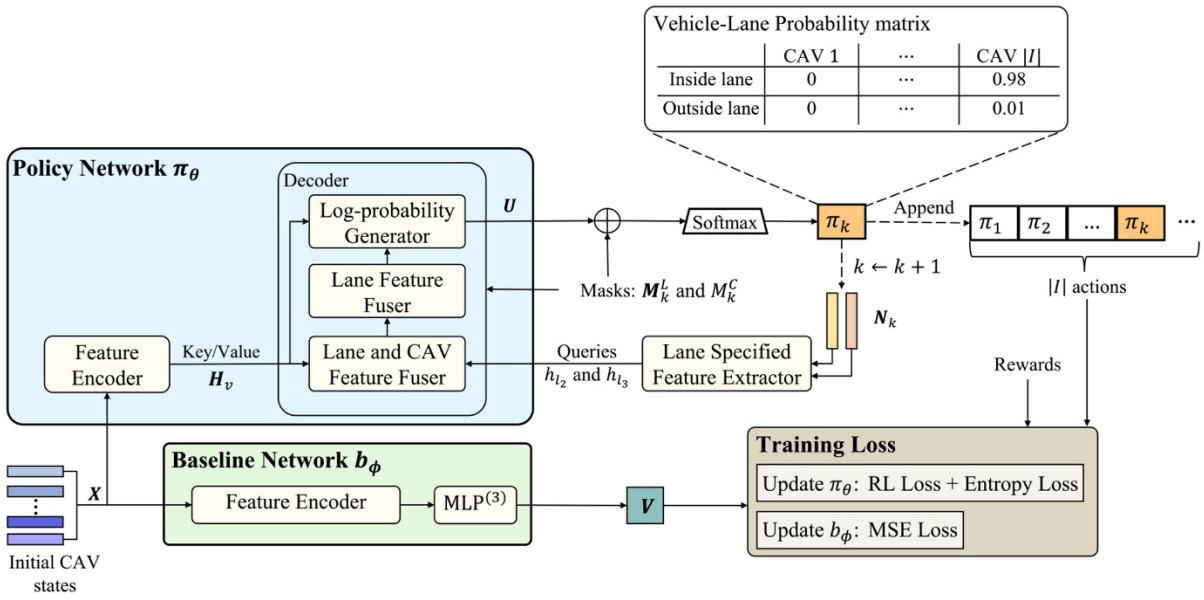


Figure 5.3. The network structure of proposed VORLA-PN

Feature encoder: This encoder takes in the initial CAV states $\mathbf{X} \in R^{|I|\times 3}$ and attends to information from all CAVs. The resulting vehicle state embedding $\boldsymbol{H_v}$ is represented as:

$$\boldsymbol{H_v} \in R^{|I|\times d} = \text{TransformerEnc}(X), \tag{5.21}$$

where d denotes the embedding dimensions, and the transformer encoder, introduced in Appendix D, is adopted to process the input states.

Sequential decoder: The decoding *process* executes the decoder repeatedly, with each step inferring the correlation between lane situations and CAV states and generating a joint probability distribution for the CAVs and mainline lanes.

(i) Lane specified feature extractor: Given the lane occupation states $\boldsymbol{N_k}$, the two feature vectors $h_l, l \in \{l_2, l_3\}$, are first selected from the output of encoder $\boldsymbol{H_v}$:

$$h_l \in R^d = \left(N_k^l\right)^T \cdot \boldsymbol{H}_v, \quad l \in \{l_2, l_3\} \tag{5.22}$$

h_l provides information about which CAVs are currently assigned to the mainline lanes. Also, note that when a mainline lane has not yet been selected, the corresponding h_l is filled with trainable parameters since no CAVs have been assigned to this lane.

(ii) Lane and CAV feature fuser: Next, we mine the correlation between the current lane occupation and candidate CAVs. We input the lane-specific features h_l as the query source and the candidate CAV features as the key-value source into cross-attention layers to output $h_l' \in \mathbb{R}^d$, effectively encoding the contextual information of the lane occupation and candidate CAVs.

Multi-head attention (MHA), as a vital component of our network, is detailed in Appendix C. The candidate CAV features are extracted using the operation $\boldsymbol{H}_v \odot M_k^C$, where the symbol $\odot$ represents the element-wise product, and the mask M_k^C is broadcasted across each column of $\boldsymbol{H}_v$ to filter the relevant CAVs from. The resulting lane feature vector h_l' is computed as:

$$h_l' \in \mathbb{R}^d = \mathrm{CrossAttention}\left(h_l, \boldsymbol{H}_v \odot M_k^C\right), \quad l \in \{l_2, l_3\} \tag{5.23}$$

Note that two lane occupation features are separately processed with candidate CAV features, each through its dedicated MHA layer.

(iii) Lane feature fuser: Subsequently, we concatenate the features of both lanes and pass them into a self-attention layer to produce the output h_{ls}, representing the implicit context of the entire mainline lanes:

$$h_{\mathrm{ls}} \in \mathbb{R}^{2d} = \mathrm{SelfAttention}(h_{\mathrm{c}}, h_{\mathrm{c}}) \tag{5.24}$$

where $h_{\mathrm{c}} \in \mathbb{R}^{2d} = \mathrm{Concat}\left(h_{l_2}', h_{l_3}'\right)$.

Then, we compute the attention scores $U_l \in \mathbb{R}^{|I|}, l \in \{l_2, l_3\}$, among the mainline lanes and candidate CAV representations using Equation5.25 and clip the result (before masking) within $[-C, C]$ $(C = 10)$ using the tanh function before applying the mask (Bello et al., 2016):

$$U_l \in R^{|I|} = C \cdot \tanh\left(\frac{h_{\mathrm{ls}}^{(l)}(\boldsymbol{H}_v)^{\mathrm{T}}}{\sqrt{d}}\right), \quad l \in \{l_2, l_3\} \tag{5.25}$$

where $h_{\mathrm{ls}}^{(l_2)}$ and $h_{\mathrm{ls}}^{(l_3)} \in \mathbb{R}^d$ are spitted from h_{ls}.

Lastly, $U_{l_2}$and U_{l_3} are concatenated together to obtain $U \in R^{\mathbb{2}\times|I|}$, and processed by the SoftMax operation to represent the target lane and vehicle probability matrix $\boldsymbol{P} \in R^{\mathbb{2}\times|I|}$. Each column of U contains two values, corresponding to the scores of assigning a CAV to each of the two mainline lanes. Hence, the next passing CAV and the associated lane selection are determined by $\boldsymbol{P}$. The decoder uses the dependencies between lane situations and candidate CAV features to produce a selection probability matrix. During the training phase, actions are sampled from this multinomial probability distributions; during the inference phase, actions are selected greedily.

(3) Leader-and-Lane-specific Credit Assignment

During training, it is important yet challenging to correctly attribute the sparse termination cost, i.e., the total delay, to a sequence of vehicle and lane selections. We propose a **leader-and-lane-specific**

credit assignment rule to compute return G at each step, as outlined by the equation:

$$G\left(s_k, \underbrace{c_k, l_k}_{a_k}\right) = -\left[T(c_k) + \sum_{k'=k}^{|I|} T(c_{k'}) \cdot \mathbb{I}(l_{k'} = l_k)\right], \tag{5.26}$$

where action a_k contains two terms: C_k and L_k, which respectively denote the selected vehicle and lane at step k; T represents the travel time of the selected vehicle; and $\mathbb{I}\left(l_{k'} = l_k\right)$ is an indicator function that equals 1 solely when $l_{k'}$ equals l_k. This equation means that the return calculation only sums the travel time cost of subsequent CAVs assigned to the same lane, rather than summing subsequent all costs.

This rule associates an action only with the rewards obtained afterward, as prior rewards have no bearing on how good the action is, which is also consistent with microscopic traffic flow models: the leader's driving behaviour can influence the followers, but the followers cannot influence the leader.

Moreover, this rule differentiates the mutual vehicle influences from different lanes. For example, lane-changing CAVs direct impact the target-lane subsequent vehicles, without impacting those in the previous lane. Also, for inside-lane CAVs, the impact of assignment of on-ramp CAVs on them is challenging to assess. However, the influence of CAVs moving from the outside lane to the inside lane is evident.

(4) Policy Optimization

The REINFORCE algorithm (Williams 1992) is utilized to update the policy network π_θ, with a learnable baseline network b_ϕ to reduce the variance of gradient estimates. The REINFORCE loss, $\mathcal{L}_{\text{RL}}$, is formulated as follows:

$$\mathcal{L}_{\text{RL}} = -\mathbb{E}_{\tau\sim\pi_\theta}\left[\left(G(\tau) - b_\phi(\tau)\right) \cdot \log \pi_\theta(\tau)\right], \tag{5.27}$$

where $\mathbb{E}_{\tau\sim\pi_\theta}$ denotes the expectation over the trajectories $\tau = (s_0, a_0, s_1, a_1, \ldots)$ (i.e., the sequences of actions, states, and rewards) sampled from π_θ; G_k refers to the return at step k, which is calculated based on Equation (5.26); b_ϕ is the baseline reward of instance calculated by the baseline network and π_θ is the action probability by the policy network. To discourage premature convergence, a negative entropy loss is integrated as follows:

$$\mathcal{L}_{\text{entropy}} = -\text{Entropy}(\pi_\theta) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{s,a\in\tau} \pi_\theta(a|s) \log\left(\pi_\theta(a|s)\right)\right] \tag{5.28}$$

Consequently, the total loss for the policy network denoted as $\mathcal{L}_\pi$, combines both the REINFORCE loss and the entropy loss:

$$\mathcal{L}_\pi = \mathcal{L}_{\text{RL}} + c_1 \cdot \mathcal{L}_{\text{entropy}} \tag{5.29}$$

where c_1 is the coefficient of entropy loss. By minimizing $\mathcal{L}_\pi$, the policy distribution is optimized towards minimizing total travel delay.

Regarding the baseline network, its loss is defined as follows:

$$\mathcal{L}_b = \mathbb{E}_{\tau\sim\pi_\theta}\left[\left(G(\tau) - b_\phi(\tau)\right)^2\right], \tag{5.30}$$

which aims to minimize the mean squared error (MSE) between the unbiased returns from the

environment $G(\tau)$ and the estimated baseline values $b_\phi(\tau)$.

Lastly, the details of training procedure are outlined in Algorithm 1.

Table 5.2. Training process for VORLA network.

Algorithm 1. Training process for VORLA network

Input: Training dataset D, number of epochs E, numbers of CAVs per epoch I, batch size B.
Output: Network parameters θ and ϕ

for epoch:= 1 to E **do**
  Sample B instances from dataset D.
  for instance:= 1 to B **do**
    set initial CAV states $\boldsymbol{X}$.
    set initial lane selection states $\boldsymbol{N}_1$.
    set step counter $k \leftarrow 1$.
    while $k < I$
      $\boldsymbol{a}_k \leftarrow \pi_k(\boldsymbol{N}_k, \boldsymbol{X})$.
    end while
    Execute actions $(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_T)$ by low-level NMPC and obtain the delays of all CAVs.
    Calculate returns $(G_1, \cdots, G_{|I|})$ based on Equation (5.26).
  end for
  Calculate loss $\mathcal{L}_\pi$ based on (5.29).
  Calculate loss $\mathcal{L}_b$ based on (5.30).
  $\theta \leftarrow \mathrm{ADAM}(\theta, \nabla\mathcal{L}_\pi)$.
  $\phi \leftarrow \mathrm{ADAM}(\phi, \nabla\mathcal{L}_b)$.
end for
return θ and ϕ

5.4 Low-level Nonlinear Model Predictive Controller

We formulate the trajectory generation for a short horizon as an unconstrained optimization problem, which is solved by an open-source library LBFGS-Lite.

$$\min_{U_i, V_i, P_i} \left[J_e, J_s, J_a, J_i, J_k, J_f\right] \cdot \lambda \tag{31}$$

where $U_i = \{u_{i,t}\}_{t=0}^{T}$, $V_i = \{v_{i,t}\}_{t=0}^{T}$, and $P_i = \{p_{i,t}\}_{t=0}^{T}$ describe the trajectory of CAV i; λ is the weight vector used to trade off each cost term.

(1) Traffic Efficiency J_e

To optimize traffic efficiency, we minimize the cumulative difference between each vehicle's speed and the desired speed over the entire time horizon. This is defined as:

$$J_e = \sum_{t=0}^{T} \left(v_{i,t} - \bar{v}\right)^2 \tag{5.32}$$

(2) Target Sequence and Lane J_s

The upper-level controller decides the target mainline lane and vehicle order for navigating through the end of the merging section, which consequently determines the target leading vehicle for

each CAV. Hence, NMPC must form the minimum spacing between any CAV i and its target leading vehicle over the time horizon:

$$J_s = \sum_{t=0}^{T} \psi\left(p_{i,t}, v_{i,t}, \hat{p}_t\right)^2 \tag{5.33}$$

where $\hat{p}_t$ represents the position of the target leading vehicle at time t, and τ is the minimum time headway. Note that the leading vehicle is determined by the upper-level controller. $\psi\left(p_{i,t}, v_{i,t}\right) = \min\{\hat{p}_t - p_{i,t} - v_{i,t} - d_0 \cdot \tau, 0\}$ indicates insufficient spacing, where the gap is smaller than the product of the follower's velocity and the minimum time headway, plus the standstill distance d_0.

(3) Reciprocal Avoidance J_a

The points which are close to the current leader are selected and penalized:

$$J_a = \sum_{t=0}^{T} \psi\left(p_{i,t}, v_{i,t}, \tilde{p}_t\right)^2 \tag{5.34}$$

where $\tilde{p}_t$ represents the position of current leading vehicle in the same lane at time t,. $\psi\left(p_{i,t}, v_{i,t}\right) = min\{\tilde{p}_t - p_{i,t} - v_{i,t} - d_0 \cdot \tau, 0\}$ indicates insufficient spacing. $\tilde{p}_t$ denotes the position of current leading vehicle.

(4) Initial Condition J_i

We specify the starting position and speed conditions for the trajectory of each CAV.

$$J_i = \left(p_{i,0} - p_{i,\text{init}}\right)^2 + \left(v_{i,0} - v_{i,\text{init}}\right)^2 \tag{5.35}$$

(5) Kinematic Condition J_k

J_k links the control inputs, velocities, and positions between consecutive time steps, which is defined by the following equation:

$$J_k = \sum_{t=0}^{T} \left(v_{i,t+1} - v_{i,t} - u_{i,t}\Delta t\right)^2 + \left(p_{i,t+1} - p_{i,t} - 0.5\left(v_{i,t} + v_{i,t+1}\right) \cdot \Delta t\right)^2 \tag{5.36}$$

Here, the difference in velocities is linearly dependent on the control input, while the difference in positions has a parabolic relationship with velocity.

(6) Feasibility Condition J_f

We limit the value of velocity and control input within feasible regions.

$$J_f = \sum_{t=0}^{T} \psi_v\left(v_{i,t}\right) + \psi_u\left(u_{i,t}\right), \tag{5.37}$$

where ψ_v and ψ_u are calculated as:

$$\varphi\left(x_{i,t}\right) = \begin{cases} \left(x_{i,t} - \underline{x}\right)^2, & x_{i,t} < \underline{x} \\ 0, & \underline{x} \le x_{i,t} \le \bar{x} \\ \left(x_{i,t} - \bar{x}\right)^2, & x_{i,t} \ge \bar{x} \end{cases} \tag{5.38}$$

5.5 Experiments

Extensive numerical experiments are conducted to evaluate and analyse our approach. Section 5.4.1 describes the experimental settings. Section 5.4.2 assesses the efficacy of our method by comparing it with three baseline methods in terms of traffic performance and computation time. Section 5.4.3 evaluates the effectiveness of the proposed leader-and-lane specific credit assignment mechanism. Section 5.4.4 presents an ablation study conducted to assess the impact of a key component of our method. Lastly, Section 5.4.5 provides a case study to illustrate the decision process and the trajectories generated.

(1) Experimental Settings

The microscopic traffic simulator SUMO is adopted to conduct numerical experiments. The simulated freeway segment consists of two mainline lanes and one on-ramp, as depicted in Fig. 5.4. The mainline lanes extend for 1.8 kilometres, comprising three sections: the upstream segment, where CAVs are randomly generated; the cooperation segment for merges and lane changes; and the downstream segment, which fully covers the affected areas.

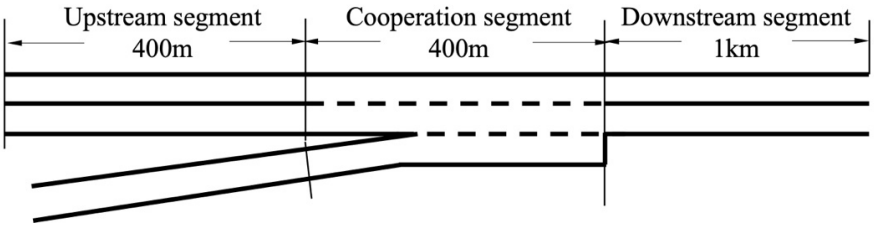


Figure 5.4. Road layout

During training, each episode randomly generates a group of 12–15 CAVs filling the upstream segment, with 25–30% on the on-ramp, 35–40% on the outside lane, and 30–40% on the inside lane. The initial time headway between any two CAVs in one lane is distributed within 1.2–2 seconds, with 1.2 seconds being the minimum time headway. The initial speeds of mainline CAVs and on-ramp CAVs are randomized between 100–120 km/h and 80–104 km/h, respectively. Table 5.2 provides the values of vehicle parameters and the hyper-parameters for training DRL methods.

Table 5.3. Vehicle Parameters and hyperparameters for training DRLs.

Vehicle parameter	Value	Hyper-parameter	Value
Minimum time headway τ	1.2s	Batch size $\|B\|$	64
Standstill distance d_0	10m	Learning rate η_θ	1e-5
Maximum speed $\bar{v}$	120 km/h	Learning rate η_ϕ	1e-5
Maximum acceleration $\bar{a}$	$5\mathrm{m/s^{-2}}$	Entropy weight c_1	5e-3
Maximum deceleration $\bar{b}$	$8\mathrm{m/s^{-2}}$	Embedding dim	128

(2) Comparative Analysis

We compare our method with following two methods to evaluate its performance:

- SA: Simulated Annealing (SA) is a metaheuristic method with guided neighbourhood search to

approximate global optimization. Here, it is employed to search for vehicle orders and lane assignments. Specifically, each solution is generated by randomly picking some vehicles from different lanes and then assigning them to new target lanes. The process of SA detailed in Appendix E is respectively run within time limits for stopping: 10 minutes (SA-10) and 40 minutes (SA-40).

- FIFO: First rotate CAVs from the ramp and outside lanes to a shared straight line and then orders them based on the first-in-first-out (FIFO) way.

Table 5.3 presents the results of different methods on 128 instances. These instances are randomly generated using the RL training seed and other random seeds, respectively. Each instance randomly generates 15 vehicles, resulting in a total of 1920 vehicles. The evaluation metrics include average vehicle delay and running time. Lower average vehicle delay indicates a higher quality solution, and shorter computation time reflects better suitability for real-time applications.

Table 5.4. Performance comparison.

		Ours	SA-10	SA-40	FIFO
Training seeds	Average travel time (s)	4.23	4.41	4.36	5.01
	Computation time (s)	< 0.01	600	2400	< 0.01
Validation seeds	Average travel time (s)	4.33	5.08	4.92	5.28
	Computation time (s)	< 0.01	600	2400	< 0.01

The data in the table demonstrates that our method outperforms the SA-10, SA-40, and FIFO approaches in both average travel time and computation time. Our method achieves the lowest average travel time in both training (4.23s) and validation (4.33s) phases, indicating more efficient traffic management. Moreover, the computation time is significantly lower than that of the SA-10 and SA-40 methods, which take 600s and 2400s, respectively. This suggests that our method is not only more effective in reducing travel time but also far more computationally efficient, making it highly suitable for real-time applications.

(3) Analysis on credit assignment methods

To demonstrate the effectiveness of our proposed leader-and-lane-specific credit assignment mechanism, we compare it with other reward shaping approaches:

- Terminal reward: An entire sequence of actions is associated with a sparse terminal reward, i.e., the negative average travel delay of a CAV group.
- Vehicle-specific reward: Each action is only associated with the negative travel delay of the selected vehicle.

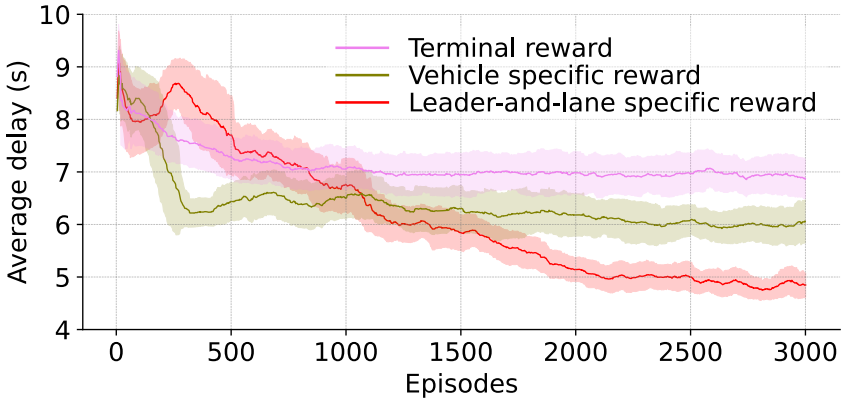


Figure 5.5. Learning curves under different credit assignment approaches.

As illustrated in Figure 5.5, under various credit assignment approaches, the policy network converges stably but achieves different performance. The proposed assignment approach significantly helps the policy network converge to solutions with lower average delays. It is noteworthy that this approach explicitly associates each action with the delays of a selected vehicle and those subsequently affected, rather than with the delays of all vehicles. In contrast, the terminal reward approach attempts to directly optimize an entire action sequence towards the minimum overall delay. However, the network struggles to learn effective solutions due to the difficulty of understanding the impact of each vehicle and lane choice on the delays of subsequent vehicles. Another vehicle-specific approach associates each action solely with the delay of the selected vehicle. Due to its explicitness, the network converges to a local optimum more effectively than with the terminal reward approach. However, because it does not account for the costs to following vehicles, the resulting traffic performance is inferior to that achieved by our proposed assignment approach.

(4) Ablation studies

Our DRL method incorporates learnable baseline and entropy regulation techniques. We conducted ablation studies to assess the effectiveness of these techniques. Fig. 5.6 displays the learning curves under three conditions: with baseline and entropy regulation, without the baseline, and without entropy.

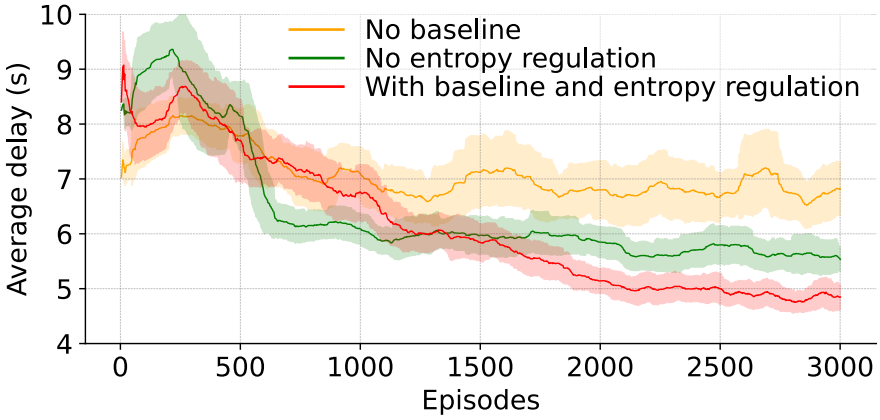


Figure 5.6. Ablation study.

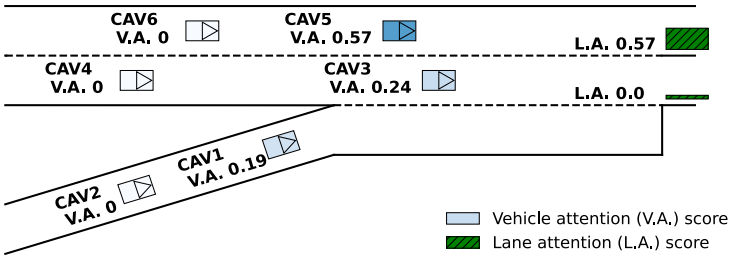
The key observations are as follows: First, the learning curve, aided by the two technologies, achieves noticeably lower cost (i.e., average delay), demonstrating their effectiveness in converging to

superior solutions. Second, in the absence of the baseline, variance increases, exerting adverse effects on the learning process. Lastly, without entropy regulation, the learning curve exhibits a pronounced bend after 500 episodes, indicating rapid and premature convergence.

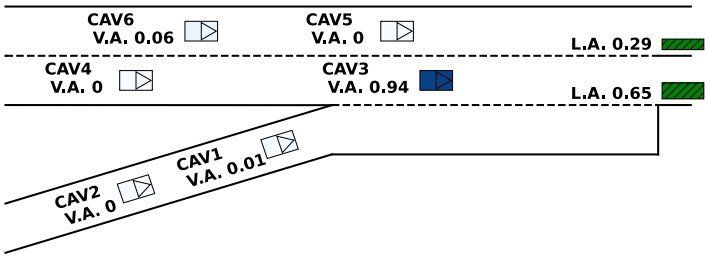
(5) Case study

To visualize the decision process and detailed trajectories generated by the proposed approach, a case study on a group of six CAVs making lane-changing and merging decisions is carried out. Figure 5.7 (a)-(f) displays how attention mechanisms are applied to different lanes and vehicles to make merging decisions. The attention values help prioritize which vehicles and lanes should be focused on during the merging process. For target lane selection, each vehicle has two scores for two mainline lanes, referred to as the lane attention (L.A.) values. For vehicle selection, each vehicle attention (V.A.) score is calculated by summing the two L.A. values. Higher V.A. and L.A. values correspond to a higher probability of choosing and assigning the vehicle to the lane.

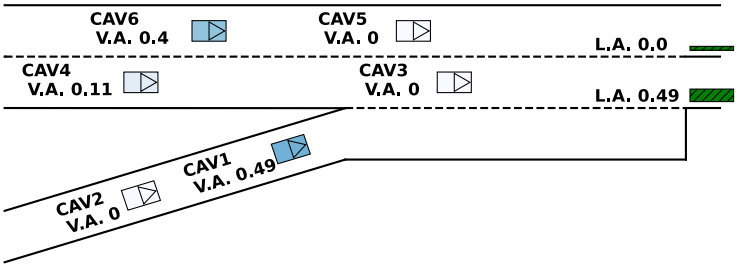
In the first step, as shown in Figure 5.7(a), based on their initial positions and velocities, CAV5 is the first vehicle to be selected and assigned to the inside lane. CAV3 has the second highest vehicle attention value, making it a strong candidate for selection as the first vehicle. Unsurprisingly, in the second step, CAV3 is then selected and assigned to the outside lane, consistent with the decision made in the first step. In the third step, both CAV1 and CAV6 are strong candidates for selection, especially compared to CAV4, due to their more downstream positions. The policy network selects CAV1 first, followed by CAV6 in the subsequent step. Finally, CAV4 and CAV2 are selected in order, with CAV4 chosen first because of its higher speed and more downstream position. Notably, CAV4 changes to the inside lane to avoid the influence of the two on-ramp vehicles, which is a preferred decision for optimizing overall traffic efficiency.



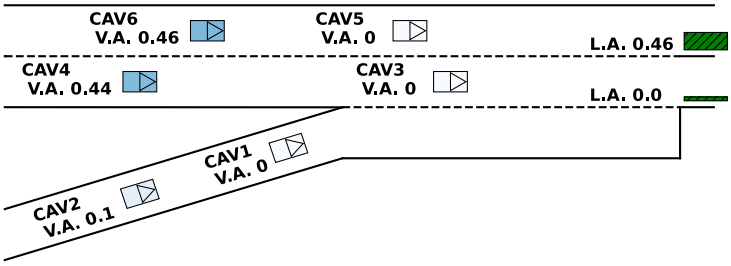
(a) Step 1: CAV5 is selected and assigned to the inside mainline lane.



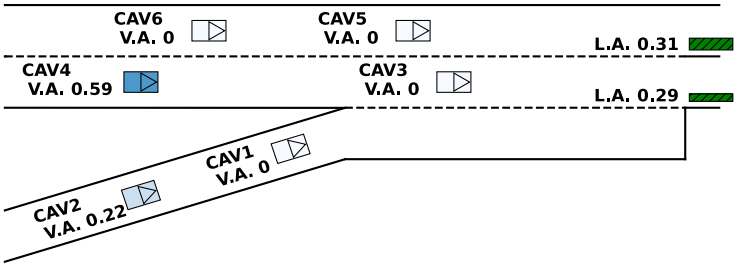
(b) Step 2: CAV3 is selected and remains in the outside mainline lane.



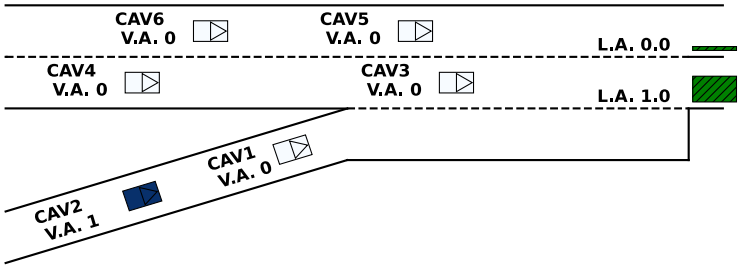
(c) Step 3: CAV1 is selected and assigned to the outside mainline lane.



(d) Step 4: CAV6 is selected and assigned to the inside mainline lane.



(e) Step 5: CAV4 is selected and changed to the inside mainline lane.



(f) Step 6: CAV2 is selected and assigned to the outside mainline lane.

Figure 5.7. Vehicle and lane selection at each step.

5.6 Summary

In this chapter, we mathematically describe and define CAV-based multi-lane freeway merging control, clarifying the connections between scheduling decisions and trajectory planning. We then propose a hybrid bi-level control approach that integrates deep reinforcement learning with optimization

modelling to optimize vehicle sequences, lane selections, and trajectories, aiming to combine the strengths of both methods. Thanks to the low-level NMPC providing accurate delay data, the upper-level policy network can correctly and unbiasedly estimate the impact of scheduling decisions under various vehicle scenarios.

By incorporating NMPC into DRL methods, we achieve comprehensive strategies for various decision combinations, significantly improving traffic efficiency rather than focusing solely on individual vehicles. Additionally, by formulating the scheduling problem as a MDP, our method enables a more informed and effective search process for the merging problem.

Experiments demonstrate that our method quickly achieves superior solution quality compared to other meta-heuristic and rule-based methods. It effectively balances computational time with solution quality while also offering promising scalability across scenarios with varying numbers of vehicles.

Chapter 6

Conclusion

6.1 Summary

This thesis presents three CAV-based control strategies aimed at improving overall traffic efficiency, covering scenarios from single-lane freeway merging sections to multi-lane freeway merging sections. Mathematical models are formulated for each case, revealing the close coupling between vehicle scheduling and trajectory planning, which introduces significant computational challenges. To address these challenges, the solution algorithms are developed from the following three perspectives to expedite the solving process:

In Chapter 3, seeking the optimal vehicle sequence is treated as a tree search problem, with the straightforward strategy to expedite the search process being the pruning of tree branches. Motivated by this idea, an optimal condition is derived based on the properties of single-lane freeway merging scenarios, effectively narrowing down the search nodes.

In Chapter 4, the scope is extended to the multi-lane freeway merging problem, making the formulated model more complex compared to the model in Chapter 1. It is noted that, although the overall model is complex, it can be more easily solved by existing solvers if certain variables are fixed. To leverage this, a GBD-based decomposition method is designed to iteratively solve two more manageable sub-models.

In Chapter 5, inspired by the application of deep reinforcement learning (DRL) in combinatorial optimization, a DRL-based method is designed to directly search for effective scheduling decisions, while low-level trajectories are generated by optimization methods to ensure safety.

6.2 Contributions

The contribution of this thesis can be summarized according to the three main works above.

The first work in Chapter 3 for the single-lane freeway merging problem has the following six contributions.

- A mixed integer nonlinear programming (MINLP) model is formulated to model the cooperative merging of on-ramp and mainline traffic streams. The proposed model jointly determines both the optimal merging sequence of these vehicles and their trajectories to minimize the disruption of on-ramp merging traffic to mainline traffic.
- An integrated solution algorithm is proposed to simultaneously obtain the optimal merging sequence and detailed trajectories. The vehicle sequence search process is designed based

on the necessary optimality condition of the model that we identify and prove, so that the solution space for integer variables, i.e., merging sequences, is greatly refined, while ensuring optimality. During the search process, the relaxed nonlinear programming (NLP) model is solved efficiently by an iterative linear programming method. Thus, the proposed methods can satisfy real-time computation.

- The proposed approach, composed of the proposed model and algorithm, is an integrated approach so that the resulting merging sequence can guarantee feasible and high-quality trajectories. Moreover, the optimal traffic efficiency is obtained by the resulting optimal merging sequence, rather than heuristically assigning a facilitating vehicle to an on-ramp vehicle, as in many existing studies.
- Trajectories are characterized by continuous-time functions so that artificial setting of numerous discrete points is avoided, and the number of decision variables is reduced. The convex hull property of the Bernstein basis is incorporated to ensure that all constraints, such as car-following, merging safety and constraints on vehicle speeds and accelerations, are guaranteed at any time, rather than only at discrete time points.
- The merging time and locations are determined by the model instead of relying on external computational procedures. In other words, the merging time and locations are part of the outcome of the proposed model.
- The traffic efficiency, safety, and computational efficiency of the proposed approach are demonstrated under different traffic conditions and compared with three alternative methods, on the NGSIM dataset.

The second work discussed in Chapter 4, which focuses on decomposing scheduling and trajectory planning decisions for the multi-lane freeway merging problem, makes three key contributions.

- This chapter proposes a mixed integer nonlinear programming (MINLP) model that can simultaneously optimize lane changes, vehicle sequences, and detailed trajectories. The proposed model optimizes not only traffic performance (i.e., minimizing the total travel time and reducing unnecessary lane-changing manoeuvres) but also trajectory quality (i.e., maximizing the smoothness of speed and acceleration profiles). Thus, from the macroscopic perspective, the proposed model fully utilizes the capacity of multiple lanes; from the microscopic perspective, the proposed model comprehensively evaluates the impact of each role of vehicles, i.e., as a facilitating, lane-changing, or car-following vehicle, on the overall system benefit. This comprehensive evaluation helps determine the most cost-effective driving manoeuvres and trajectories for each vehicle.
- To solve the proposed model, we have designed a Generalized Benders Decomposition (GBD)-based algorithm. A relaxed master problem and a primal subproblem are derived from the proposed MINLP model. To accelerate convergence, the feasibility-guaranteed primal subproblem is introduced. Moreover, the property of finite convergence is proved. The proposed algorithm significantly reduces computational time compared to Gurobi due to this subtle decomposition of the nonlinear programming problem and the integer programming problem.
- A series of comprehensive evaluation experiments using the open-source SUMO simulator over various traffic conditions are conducted. Our model outperforms all

benchmark models, demonstrating the benefits of optimal vehicle lane-changing and sequencing decisions, and leveraging microscopic vehicle information. Additionally, we analyse the impact of the lengths of cooperative areas on our model's performance and determine the recommended lengths for the optimal traffic efficiency.

The third work discussed in Chapter 5, which integrates the rapid inference capabilities of DRL for vehicle scheduling decisions with explainable and safe optimization modelling for vehicle trajectory planning, makes five key contributions.

- A novel hybrid paradigm that combines DRL for solving vehicle combinatorial scheduling decisions with optimization methods for trajectory generation is introduced. In this approach, the trajectory optimization provides accurate delay information to the DRL during the training phase, allowing the policy network to unbiasedly evaluate the impact of scheduling decisions.
- The VORLA policy network is designed to determine the target lanes and right-of-way for a group of vehicles. A sequential decoding process is proposed to generate the passing order and lane assignment one vehicle at a time.
- The leader-and-lane specific credit assignment mechanism is developed, leveraging domain knowledge to effectively learn a sequence of actions that minimize overall travel delay.
- The nonlinear model predictive controller is formulated to safely accomplish the scheduling tasks, ensuring that the vehicles follow the required sequence and lane assignments.
- Experiments demonstrate that our method consistently achieves superior solution quality compared to other meta-heuristic and rule-based methods. It effectively balances computational time with solution quality and shows promising scalability across scenarios with varying numbers of vehicles.

6.3 Future work

The following directions can be further explored in the future.

- As a future endeavour, first, it would be desirable to extend the current method to more complex multi-lane scenarios (with three or more mainline lanes), where intricate vehicle interactions, such as consecutive lane-changing behaviours, need to be considered.
- Second, the assumption of 100% CAVs can be relaxed to accommodate mixed traffic conditions, better aligning with near-future reality. One perspective could be to secure safe and efficient CAV decision-making with the consideration of uncertain human-driven vehicles.
- Third, the combination of microscopic trajectory generation methods and flow-based merging control methods appear to be promising in mitigating congestion and may deserve further examination. By integrating these approaches, it is potential to create a more seamless and efficient merging process, particularly in over-saturated traffic scenarios, which could lead to significant improvements in overall traffic flow and congestion reduction.
- Fourth, more efficient trajectory planners can be investigated to further reduce the computation time, making real-time applications more feasible. Improving the planners

enhances the responsiveness and reliability of CAV systems, particularly in dynamic traffic environments.

- Fifth, integrating multi-vehicle cooperation, trajectory planning, and environmental perception is crucial to fully leveraging the capabilities of connected communication. This approach would enable real-time optimization of driving behaviours and the detection of the surrounding driving environment.
- Last but not least, Large Language Models (LLMs) could offer powerful generalization capabilities for designing algorithms applicable to various merging sections. Additionally, LLMs could facilitate natural language interactions between controllers and humans, enhancing communication and safety in complex traffic scenarios.

Appendix

Appendix A

Direct Deduction of Proposition 1 in chapter 3:

All on-ramp vehicles are split into two groups: $\{1^{st}, \dots, k^{th}\}$ on-ramp vehicles and $\{k+1^{th}, \dots, |I_r|^{th}\}$ on-ramp vehicles. Correspondingly, the mainline vehicles can be two associated groups: $1^{st}, \dots, n^{th}\}$ mainline vehicles and $\{n+1^{th}, \dots, |I_m|^{th}\}$ mainline vehicles. Note that the n^{th} mainline vehicle refers to the one in front of the k^{th}on-ramp vehicle, although the "n" is unknown.

Minimizing the first group of on-ramp and mainline vehicles can be written as:

$$\min_{\boldsymbol{\gamma}_1,\dots,\boldsymbol{\gamma}_{|I_r|}} \left(\sum_{j=1}^{k} t_{r,j} + \sum_{i=1}^{n} t_{m,i}\right) \tag{A.1}$$

Then, it is straightforward that the decision variables $\boldsymbol{\gamma}_{k+1}, \dots, \boldsymbol{\gamma}_{|I_r|}$ do not affect (A.1). Hence, (A.1) can be express as (A.2).

$$\begin{aligned} &\min_{\boldsymbol{\gamma}_1,\dots,\boldsymbol{\gamma}_k} \left(\sum_{j=1}^{k} t_{r,j} + \sum_{i=1}^{n} t_{m,i}\right) \\ &= t_{r,1}^* + \cdots + t_{r,k}^* + t_{m,1}^* + \cdots + t_{m,n}^* \end{aligned} \tag{A.2}$$

Similarly, minimizing another group of on-ramp and mainline vehicles can be written as (A.3) and can be extended as (A.6):

$$\min_{\boldsymbol{\gamma}_1,\dots,\boldsymbol{\gamma}_{|I_r|}} \left(\sum_{j=k+1}^{|I_r|} t_{r,j} + \sum_{i=n+1}^{|I_m|} t_{m,i}\right) \tag{A.3}$$

$$= \min_{\boldsymbol{\gamma}_1,\dots,\boldsymbol{\gamma}_{|I_r|}} \left[\sum_{j=k+1}^{|I_r|} (\Delta\, t_{r,j} + \min_{\boldsymbol{\gamma}_1,\dots,\boldsymbol{\gamma}_k} t_{r,k}) + \sum_{i=n+1}^{|I_m|} (\Delta\, t_{m,i} + \min_{\boldsymbol{\gamma}_1,\dots,\boldsymbol{\gamma}_k} t_{m,n})\right] \tag{A.4}$$

$$= \min_{\boldsymbol{\gamma}_1,\dots,\boldsymbol{\gamma}_{|I_r|}} \left[\sum_{j=k+1}^{|I_r|} (\Delta\, t_{r,j} + t_{r,k}^*) + \sum_{i=n+1}^{|I_m|} (\Delta\, t_{m,i} + t_{m,n}^*)\right] \tag{A.5}$$

$$= \min_{\boldsymbol{\gamma}_{k+1},\dots,\boldsymbol{\gamma}_{|I_r|}} \left[\sum_{j=k+1}^{|I_r|} (\Delta\, t_{r,j} + t_{r,k}^*) + \sum_{i=n+1}^{|I_m|} (\Delta\, t_{m,i} + t_{m,n}^*)\right] \tag{A.6}$$

$$= t_{r,k+1}^* + \cdots + t_{r,|I_r|}^* + t_{m,n+1}^* + \cdots + t_{m,|I_m|}^* \tag{A.7}$$

where $\Delta\, t_{r,j}$ and $\Delta\, t_{m,i}$ are the part of time delay of the j^{th} on-ramp vehicle and i^{th} mainline vehicle caused by the on-ramp vehicles after the k^{th} on-ramp vehicle, respectively. From (A.6), we can see that the optimal time of following vehicles, i.e., $t_{r,k+1}^* + \cdots + t_{r,|I_r|}^*$and $t_{m,n+1}^* + \cdots + t_{m,|I_m|}^*$, requires the optimal time of preceding vehicles $t_{r,k}^*$ and $t_{m,n}^*$. Therefore, Proposition 1 is obtained.

Appendix B

Derivation of Equation (6) in Chapter 3:

The polynomial position equation in (3) can be transformed as follows:

$$x(t) = \theta_3 \cdot t^3 + \theta_2 \cdot t^2 + \theta_1 \cdot t + \theta_0 = t_f^3\theta_3\left(\frac{t}{t_f}\right)^3 + t_f^2\theta_2\left(\frac{t}{t_f}\right)^2 + t_f\theta_1\frac{t}{t_f} + \theta_0, \quad \frac{t}{t_f} \in (0,1] \qquad \text{(B.1)}$$

The related three order Bézier curve can be expanded as follows:

$$\begin{aligned} P^x(s) &= \textstyle\sum_{j=0}^{3} B_j^3(s)P_j^x \\ &= B_0^3(s)P_0^x + B_1^3(s)P_1^x + B_2^3(s)P_2^x + B_3^3(s)P_3^x \\ &= (1-s)^3P_0^x + 3s(1-s)^2P_1^x + 3s^2(1-s)P_2^x + s^3P_3^x \\ &= (P_3^x - 3P_2^x + 3P_1^x - P_0^x)s^3 + (3P_2^x - 6P_1^x + 3P_0^x)s^2 + (3P_1^x - 3P_0^x)s + P_0^x, \quad s \in [0,1] \end{aligned} \qquad \text{(B.2)}$$

Then, a set of equations can be written to build the relationship between $\boldsymbol{\theta}$ and P^x:

$$\begin{cases} P_3^x - 3P_2^x + 3P_1^x - P_0^x = t_f^3\theta_3 & \text{(B.3)} \\ 3P_2^x - 6P_1^x + 3P_0^x = t_f^2\theta_2 & \text{(B.4)} \\ 3P_1^x - 3P_0^x = t_f\theta_1 & \text{(B.5)} \\ P_0^x = \theta_0 & \text{(B.6)} \end{cases}$$

By solving (B.3)–(B.6), Equation (6) is derived.

Derivation of Equation (7) in Chapter 3.

Similarly, the polynomial velocity equation in (3) can be transformed as follows:

$$\begin{aligned} v(t) &= 3\theta_3 \cdot t^2 + 2\theta_2 \cdot t + \theta_1 \\ &= 3\theta_3 t_f^2\left(\frac{t}{t_f}\right)^2 + 2\theta_2 t_f\frac{t}{t_f} + \theta_1, \quad \frac{t}{t_f} \in (0,1] \end{aligned} \qquad \text{(B.7)}$$

The related two order Bézier curve can be expanded as follows:

$$\begin{aligned} P^v(s) &= \textstyle\sum_{j=0}^{2} B_j^2(s)P_j^v \\ &= B_0^2(s)P_0^v + B_1^2(s)P_1^v + B_2^2(s)P_2^v \\ &= (P_0^v - 2P_1^v + P_2^v)s^2 + (2P_1^v - 2P_0^v)s + P_0^v \end{aligned} \qquad \text{(B.8)}$$

Then, a set of equations can be written as:

$$\begin{cases} P_0^v - 2P_1^v + P_2^v = 3\theta_3 t_f^2 & \text{(B.9)} \\ 2P_1^v - 2P_0^v = 2\theta_2 t_f & \text{(B.10)} \\ P_0^v = \theta_1 & \text{(B.11)} \end{cases}$$

By solving (B.9)–(B.11), Equation (7) is derived.

Appendix C

The multi-head attention (MHA) module in Chapter 5.

The multi-head attention (MHA) module (Vaswani et al. 2017) is a crucial component of our network. The MHA module takes in a query source, $h^q \in \mathbb{R}^d$, and a key-value source, $h^{k,v} \in \mathbb{R}^d$.

Both h^q and $h^{k,v}$ are projected H times into different subspaces using linear layers, where H refers to the number of heads. For each head $h \in \{1, 2, \ldots, H\}$, the query, key, and value vectors are calculated as follows:

$$Q_h = W_h^Q h^q \tag{C.1}$$

$$K_h = W_h^K h^{k,v} \tag{C.2}$$

$$V_h = W_h^V h^{k,v}, \tag{C.3}$$

where h^q is the query source; $h^{k,v}$is the key-value source; $\mathbf{W_h^Q}$, $\mathbf{W_h^K}$, and $\mathbf{W_h^V} \in R^{d_h \times d}$ are learnable weight matrices, and $d_h = d/H$.

Each attention-based head α_h is then determined through the scaled-dot product operation:

$$\alpha_h = \text{Attention}(Q_h, K_h, V_h) = \text{Softmax}\left(\frac{Q_h K_h^T}{\sqrt{d_h}}\right) V_h \tag{C.4}$$

The last operation of the MHA module is to concatenate all heads together:

$$\text{MHA}\left(h^q, h^{k,v}\right) = \text{Concat}(\alpha_1, \alpha_2, \cdots, \alpha_H)\boldsymbol{W}^O \tag{C.5}$$

where $\boldsymbol{W}^O \in \mathbb{R}^{d \times d}$ is the learnable matrix for the output layer. Hence, the output of MHA aggregates the key/value, guided by the query.

Appendix D

Transformer Encoder module in Chapter 5.

The core elements of the Transformer encoder are the multi-head attention (MHA) mechanism, position-wise feed-forward network (FFN), residual connections, and layer normalization.

The multi-head attention sublayer is responsible for facilitating communication between tokens (input elements) to effectively capture their relationships. Each token may have multiple semantics or functions depending on the surrounding tokens. MHA modules allow the model to focus on various aspects of the input simultaneously, enriching the token embeddings with contextual information.

Following the MHA sublayer, (FFN) further processes the embeddings. The FFN consists of two linear projection layers with a ReLU activation function in between. The dimensionality of the embeddings is first expanded and then reduced, enhancing the model's ability to introduce non-linearity while preserving information.

Then, residual connections play a crucial role in the encoder by carrying over the previous embeddings to subsequent layers. This is done through an element-wise addition of the original input with the output of each sublayer (either MHA or FFN). These connections help mitigate the vanishing gradient problem, ensuring that the model continues to learn effectively as layers deepen.

After each residual connection, layer normalization is applied to stabilize the training process. Unlike batch normalization, which operates across batches, layer normalization normalizes the activations within each embedding vector. This reduces the effect of covariant shift, making training more stable and allowing for faster convergence.

Appendix E

Pseudo code of simulated annealing in Chapter 5.

	Simulated Annealing (SA)
1	Initialize current solution S and temperature T.
2	Initialize best solution S^*
3	Set time limit t_1
4	Set cooling factor α (0.95).
5	**For** $t = 0, t_1$ **Do**
6	Generate a new solution S' in the neighbourhood of S.
7	Compare the total delay with the best solution $\Delta D = D(S') - D(S^*)$
8	**If** $\Delta D < 0$
9	$S^* \leftarrow S'$
10	**Else if** random (0, 1) $< e^{-\frac{\Delta D}{T}}$
11	$S \leftarrow S'$
12	**End if**
13	Decrease the temperature $T = \alpha \cdot T$.
14	**End for**
15	**Return** S^*

Reference

Aashto, A. (2001) Policy on geometric design of highways and streets. *American Association of State Highway and Transportation Officials* 1(990): 158.

Bello, I., Pham, H., Le, Q. V., Norouzi, M., and Bengio, S. (2016) Neural combinatorial optimization with reinforcement learning. *arXiv preprint arXiv* 1611.09940.

Cai, X., McKinney, D. C., Lasdon, L. S., and Watkins Jr, D. W. (2001) Solving large nonconvex water resources management models using generalized benders decomposition. *Operations Research* 49(2): 235–245.

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

Carlson, R. C., Papamichail, I., Papageorgiou, M., and Messmer, A. (2010) Optimal motorway traffic flow control involving variable speed limits and ramp metering. *Transportation Science* 44(2): 238–253.

Chen, D., Ahn, S., and Hegyi, A. (2014) Variable speed limit control for steady and oscillatory queues at fixed freeway bottlenecks. *Transportation Research Part B: Methodological* 70: 340–358.

Chen, D., Hajidavalloo, M. R., Li, Z., Chen, K., Wang, Y., Jiang, L., and Wang, Y. (2023a). Deep multi-agent reinforcement learning for highway on-ramp merging in mixed traffic. *IEEE Transactions on Intelligent Transportation Systems* 24(11): 11623–11638.

Chen, J., Chen, X., and Liu, S. (2023b). Trajectory planning of autonomous mobile robot using model predictive control in human-robot shared workspace. *2023 IEEE 3rd International Conference on Electronic Technology, Communication and Information ICETCI*: 462–467.

Chen, N., van Arem, B., Alkim, T., and Wang, M. (2020) A hierarchical model-based optimization control approach for cooperative merging by connected automated vehicles. *IEEE Transactions on Intelligent Transportation Systems* 22(12): 7712–7725.

Chen, N., van Arem, B., and Wang, M. (2022a) Hierarchical optimal maneuver planning and trajectory control at on-ramps with multiple mainstream lanes. *IEEE Transactions on Intelligent Transportation Systems* 23(10): 18889–18902.

Chen, R., and Yang, Z. (2022b) A cooperative merging strategy for connected and automated vehicles based on game theory with transferable utility. *IEEE Transactions on Intelligent Transportation Systems* 23(10): 19213–19223.

Chen, T., Wang, M., Gong, S., Zhou, Y., and Ran, B. (2021) Connected and automated vehicle distributed control for on-ramp merging scenario: A virtual rotation approach. *Transportation Research Part C: Emerging Technologies* 133: 103451.

Cichella, V., Kaminer, I., Walton, C., and Hovakimyan, N. (2017) Optimal motion planning for differentially flat systems using Bernstein approximation. *IEEE Control Systems Letters* 2(1): 181–186.

Elefteriadou, L. A. (2016) The highway capacity manual 6th edition: A guide for multimodal mobility analysis. *Ite Journal*, 86(4).

Elefteriadou, L., Kondyli, A., Washburn, S., Brilon, W., Lohoff, J., Jacobson, L., Hall, F., and Persaud, B. (2011) Proactive ramp management under the threat of freeway-flow breakdown. P*rocedia-Social and Behavioral Sciences* 16: 4–14.

Frese, C., and Beyerer, J. (2011) A comparison of motion planning algorithms for cooperative collision avoidance of multiple cognitive automobiles. In Proceedings of *2011 IEEE Intelligent Vehicles Symposium (IV)*: 1156–1162, Baden-Baden, Germany.

Gao, C., Wei, N., and Walteros, J. L. (2023) An exact approach for solving pickup-and-delivery traveling salesman problems with neighborhoods. *Transportation Science* 57(6): 1560–1580.

Gao, Z., Wu, Z., Hao, W., Long, K., Byon, Y. J., and Long, K. (2021) Optimal trajectory planning of connected and automated vehicles at on-ramp merging area. *IEEE transactions on Intelligent Transportation Systems* 23(8): 12675–12687.

Geoffrion, A. M. (1972) Generalized benders decomposition. *Journal of Optimization Theory and Applications* 10: 237–260.

Häfner, B., Bajpai, V., Ott, J., and Schmitt, G. A. (2021) A survey on cooperative architectures and maneuvers for connected and automated vehicles. *IEEE Communications Surveys & Tutorials* 24(1): 380–403.

Hamednia, A., Sharma, N. K., Murgovski, N., and Fredriksson, J. (2021) Computationally efficient algorithm for eco-driving over long look-ahead horizons. *IEEE Transactions on Intelligent Transportation Systems* 23(7): 6556–6570.

Han, Y., and Ahn, S. (2018) Stochastic modeling of breakdown at freeway merge bottleneck and traffic control method using connected automated vehicle. *Transportation Research Part B: Methodological* 107: 146–166.

Hang, P., Lv, C., Huang, C., Xing, Y., and Hu, Z. (2021) Cooperative decision making of connected automated vehicles at multi-lane merging zone: A coalitional game approach. *IEEE Transactions on Intelligent Transportation Systems* 23(4): 3829–3841.

He, Y., Liu, Z., Zhou, X., and Zhong, B. (2017, May) Analysis of urban traffic accidents features and correlation with traffic congestion in large-scale construction district. In Proceedings of *2017 International Conference on Smart Grid and Electrical Automation (ICSGEA)*: 641–644, Changsha, China.

Hou, K., Zheng, F., Liu, X., and Guo, G. (2023) Cooperative on-ramp merging control model for mixed traffic on multi-lane freeways. *IEEE Transactions on Intelligent Transportation Systems* 24(10): 10774–10790.

Hu, F., and Yu, H. (2023) Safety-Critical Lane-Change Control for CAV Platoons in Mixed Autonomy Traffic Using Control Barrier Functions. *arXiv.org*.

Hu, J., Li, X., Hu, W., Xu, Q., and Kong, D. (2024) A cooperative control methodology considering dynamic interaction for multiple connected and automated vehicles in the merging zone. IEEE Transactions on Intelligent Transportation Systems 25(9): 12669–12681.

Hu, X., and Sun, J. (2019) Trajectory optimization of connected and autonomous vehicles at a multilane freeway merging area. *Transportation Research Part C: Emerging Technologies* 101: 111–125.

Hwang, S., Lee, K., Jeon, H., and Kum, D. (2022) Autonomous vehicle cut-in algorithm for lane-merging scenarios via policy-based reinforcement learning nested within finite-state machine. *IEEE Transactions on Intelligent Transportation Systems* 23(10): 17594–17606.

Karimi, M., Roncoli, C., Alecsandru, C., and Papageorgiou, M. (2020) Cooperative merging control via trajectory optimization in mixed vehicular traffic. *Transportation Research Part C: Emerging Technologies* 116: 102663.

Jiang, C., Liu, H., Qiu, C., Zhang, S., and Zhuang, W. (2024) Ramp merging sequence and trajectory optimization for connected and autonomous vehicles using deep reinforcement learning. *2024 IEEE 18th International Conference on Advanced Motion Control (AMC):* 1–7.

Karlsson, J., Murgovski, N., and Sjöberg, J. (2020) Computationally efficient autonomous overtaking on highways. *IEEE Transactions on Intelligent Transportation Systems* 21(8): 3169–3183.

Lee, C., Hellinga, B., and Ozbay, K. (2006) Quantifying effects of ramp metering on freeway safety. *Accident Analysis & Prevention* 38(2): 279–288.

Lee, C., Hellinga, B., and Saccomanno, F. (2003) Real-time crash prediction model for application to crash prevention in freeway traffic. *Transportation Research Record* 1840(1): 67–77.

Letter, C., and Elefteriadou, L. (2017) Efficient control of fully automated connected vehicles at freeway merge segments. *Transportation Research Part C: Emerging Technologies* 80: 190–205.

Li, L., Wen, D., and Yao, D. (2013) A survey of traffic control with vehicular communications. *IEEE Transactions on Intelligent Transportation Systems* 15(1): 425–432.

Li, N., Fan, A. Z., Fischer, R., Kontar, W., and Ran, B. (2021a) A Prioritized Trajectory Planning Algorithm for Connected and Automated Vehicle Mandatory Lane Changes. *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*: 770–775.

Li, P., and Zhou, X. (2017) Recasting and optimizing intersection automation as a connected-and-automated-vehicle (CAV) scheduling problem: A sequential branch-and-bound search approach in phase-time-traffic hypernetwork. *Transportation Research Part B: Methodological* 105: 479–506.

Li, S., Zhou, Y., Ye, X., Jiang, J., and Wang, M. (2023) Sequencing-enabled hierarchical cooperative on-ramp merging control for connected and automated vehicles. *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*: 5146–5153, Bilbao, Spain.

Li, T., Chen, D., Zhou, H., Laval, J., and Xie, Y. (2021b) Car-following behavior characteristics of adaptive cruise control vehicles based on empirical experiments. *Transportation Research Part B: Methodological* 147: 67–91.

Li, X., Ghiasi, A., Xu, Z., and Qu, X. (2018) A piecewise trajectory optimization model for connected automated vehicles: Exact optimization algorithm and queue propagation analysis. *Transportation Research Part B: Methodological* 118: 429–456.

Lin, S. C., Hsu, H., Lin, Y. T., Lin, C. W., Jiang, I. H. R., and Liu, C. (2020) A dynamic programming approach to optimal lane merging of connected and autonomous vehicles. In Proceedings of *2020 IEEE Intelligent Vehicles Symposium (IV)*: 349–356, Las Vegas, USA.

Mellinger, D., and Kumar, V. (2011) Minimum snap trajectory generation and control for quadrotors. In Proceedings of *2011 IEEE International Conference on Robotics and Automation*: 2520–2525, Shanghai, China.

Milanés, V., Godoy, J., Villagrá, J., and Pérez, J. (2010) Automated on-ramp merging system for congested traffic situations. *IEEE Transactions on Intelligent Transportation Systems* 12(2): 500–508.

Mu, C., Du, L., and Zhao, X. (2021) Event triggered rolling horizon based systematical trajectory planning for merging platoons at mainline-ramp intersection. *Transportation Research Part C: Emerging Technologies* 125: 103006.

Ni, D., Leonard, J. D., Jia, C., and Wang, J. (2016) Vehicle longitudinal control and traffic stream modeling. *Transportation Science* 50(3): 1016–1031.

Nilsson, J., Brännström, M., Fredriksson, J., and Coelingh, E. (2016) Longitudinal and lateral control for automated yielding maneuvers. *IEEE Transactions on Intelligent Transportation Systems* 17(5): 1404–1414.

Ntousakis, I. A., Nikolos, I. K., and Papageorgiou, M. (2016) Optimal vehicle trajectory planning in the context of cooperative merging on highways. *Transportation Research Part C: Emerging Technologies* 71: 464–488.

Tengilimoglu, O., Carsten, O., and Wadud, Z. (2023) Implications of automated vehicles for physical road environment: A comprehensive review. Transportation research part E: logistics and transportation review 169: 102989.

Pan, T., Guo, R., Lam, W. H., Zhong, R., Wang, W., and He, B. (2021) Integrated optimal control strategies for freeway traffic mixed with connected automated vehicles: A model-based reinforcement learning approach. *Transportation Research Part C: Emerging Technologies* 123: 102987.

Papageorgiou, M. (1998) Some remarks on macroscopic traffic flow modelling. *Transportation Research Part A: Policy and Practice* 32(5): 323–329.

Papageorgiou, M., and Kotsialos, A. (2002) Freeway ramp metering: An overview. *IEEE Transactions on Intelligent Transportation Systems* 3(4): 271–281.

Papageorgiou, M., Mountakis, K. S., Karafyllis, I., Papamichail, I., and Wang, Y. (2021) Lane-free artificial-fluid concept for vehicular traffic. *Proceedings of the IEEE* 109(2): 114–121.

Pei, H., Feng, S., Zhang, Y., and Yao, D. (2019) A cooperative driving strategy for merging at on-ramps based on dynamic programming. *IEEE Transactions on Vehicular Technology* 68(12): 11646–11656.

Rios-Torres, J., and Malikopoulos, A. A. (2016a) A survey on the coordination of connected and automated vehicles at intersections and merging at highway on-ramps. *IEEE Transactions on Intelligent Transportation Systems* 18(5): 1066–1077.

Rios-Torres, J., and Malikopoulos, A. A. (2016b) Automated and cooperative vehicle merging at highway on-ramps. *IEEE Transactions on Intelligent Transportation Systems* 18(4): 780–789.

Rios-Torres, J., and Malikopoulos, A. A. (2018) Impact of partial penetrations of connected and automated vehicles on fuel consumption and traffic flow. *IEEE Transactions on Intelligent Vehicles* 3(4): 453–462.

Sharma, S., Papamichail, I., Nadi, A., Van Lint, H., Tavasszy, L., and Snelder, M. (2021) A multi-class lane-changing advisory system for freeway merging sections using cooperative ITS. *IEEE Transactions on Intelligent Transportation Systems* 23(9): 15121–15132.

Shen, J., Kammara, E. K. H., and Du, L. (2022a) Fully distributed optimization-based CAV platooning control under linear vehicle dynamics. *Transportation Science* 56(2): 381–403.

Shen, J., Kammara, E. K. H., and Du, L. (2022b) Nonconvex, fully distributed optimization based CAV platooning control under nonlinear vehicle dynamics. *IEEE Transactions on Intelligent Transportation System* 23(11): 20506–20521.

Shi, H., Chen, D., Zheng, N., Wang, X., Zhou, Y., and Ran, B. (2023a) A deep reinforcement learning based distributed control strategy for connected automated vehicles in mixed traffic platoon. *Transportation Research Part C: Emerging Technologies* 148: 104019.

Shi, J., Li, K., Chen, C., Kong W., and Luo. Y. (2023b) Cooperative merging strategy in mixed traffic based on optimal final-state phase diagram with flexible highway merging points. *IEEE Transactions on Intelligent Transportation Systems* 24(10): 11185–11197.

Sun, J., and Sun, J. (2018, November) Investigating the oscillation characteristics and mitigating its impact with low-penetration connected and automated vehicles. In Proceedings of *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*: 2339–2345, Maui, USA.

Tajdari, F., Roncoli, C., and Papageorgiou, M. (2022) Feedback-based ramp metering and lane-changing control with connected and automated vehicles. *IEEE Transactions on Intelligent Transportation Systems* 23: 939–951.

Talebpour, A., and Mahmassani, H. S. (2016) Influence of connected and autonomous vehicles on traffic flow stability and throughput. *Transportation Research Part C: Emerging Technologies* 71: 143–163.

Tang, Z., Zhu, H., Zhang, X., Iryo-Asano, M., and Nakamura, H. (2022) A novel hierarchical cooperative merging control model of connected and automated vehicles featuring flexible merging positions in system optimization. *Transportation Research Part C: Emerging Technologies* 138: 103650.

Thomas, L., and Boyd, S. (2014) Minimum-time speed optimization over a fixed path. *International Journal of Control* 87(6): 1297–1311.

Tian, J., Zhang, H. M., Treiber, M., Jiang, R., Gao, Z. Y., and Jia, B. (2019) On the role of speed adaptation and spacing indifference in traffic instability: Evidence from car-following experiments and its stochastic model. *Transportation Research Part B: Methodological* 129: 334–350.

Uno, A., Sakaguchi, T., and Tsugawa, S. (1999, October) A merging control algorithm based on inter-vehicle communication. In Proceedings of *IEEE/IEEJ/JSAI International Conference on Intelligent Transportation Systems*: 783–787, Tokyo, Japan.

U.S. Department of Transportation Federal Highway Administration. (2016). Next Generation Simulation (NGSIM) Program I-80 Videos. [Dataset]. Provided by ITS DataHub through Data.transportation.gov.

van Arem, B., van Driel, C. J., and Visser, R. (2006) The impact of cooperative adaptive cruise control on traffic-flow characteristics. *IEEE Transactions on Intelligent Transportation Systems* 7(4): 429–436.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017) Attention is all you need. *Advances in neural information processing systems*: 30.

Wang, H., Meng, Q., Chen, S., and Zhang, X. (2021) Competitive and cooperative behaviour analysis of connected and autonomous vehicles across unsignalised intersections: A game-theoretic approach. *Transportation Research Part B: Methodological* 149: 322–346.

Wang, M., Daamen, W., Hoogendoorn, S. P., and van Arem, B. (2016) Connected variable speed limits control and car-following control with vehicle-infrastructure communication to resolve stop-and-go waves. *Journal of Intelligent Transportation Systems* 20(6): 559–572.

Wang, Z., Bian, Y., Shladover, S. E., Wu, G., Li, S. E., and Barth, M. J. (2019) A survey on cooperative longitudinal motion control of multiple connected and automated vehicles. *IEEE Intelligent Transportation Systems Magazine* 12(1): 4–24.

Williams, R. J. (1992) Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning* 8: 229–256.

Werling, M., Ziegler, J., Kammel, S., and Thrun, S. (2010, May) Optimal trajectory generation for dynamic street scenarios in a frenet frame. In Proceedings of *2010 IEEE International Conference on Robotics and Automation*: 987–993, Anchorage, USA.

Xiao, L., Wang, M., Schakel, W., and van Arem, B. (2018) Unravelling effects of cooperative adaptive cruise control deactivation on traffic flow characteristics at merging bottlenecks. *Transportation Research Part C: Emerging Technologies* 96: 380–397.

Xiao, Y., Coulombel, N., and de Palma, A. (2017) The valuation of travel time reliability: Does congestion matter?. *Transportation Research Part B: Methodological* 97: 113–141.

Xie, Y., Zhang, H., Gartner, N. H., and Arsava, T. (2017) Collaborative merging strategy for freeway ramp operations in a connected and autonomous vehicles environment. *Journal of Intelligent Transportation Systems* 21(2): 136–147.

Xu, Q., and Sengupta, R. (2003, June) Simulation, analysis, and comparison of ACC and CACC in highway merging control. In Proceedings of *IEEE IV2003 Intelligent Vehicles Symposium*: 237–242, Columbus, USA.

Yang, D., Jia, B., Dai, L., Jin, J. P., Xu, L., Chen, F., Zheng, S., and Ran, B. (2022) Optimization model for the freeway-exiting position decision problem of automated vehicles. *Transportation Research Part B: Methodological* 159: 24–48.

Yang, L., Zhan, J., Shang, W. L., Fang, S., Wu, G., Zhao, X., and Deveci, M. (2023) Multi-lane coordinated control strategy of connected and automated vehicles for on-ramp merging area based on cooperative game. *IEEE Transactions on Intelligent Transportation Systems* 24(11): 13448–13461.

Ye, F., Guo, J., Kim, K. J., Orlik, P. V., Ahn, H., Di Cairano, S., and Barth, M. J. (2019, June) Bi-level optimal edge computing model for on-ramp merging in connected vehicle environment. In Proceedings of *2019 IEEE Intelligent Vehicles Symposium (IV)*: 2005–2011, Paris, France.

Yu, C., Sun, W., Liu, H. X., and Yang X. (2019) Managing connected and automated vehicles at isolated intersections: From reservation to optimization-based methods. *Transportation Research Part B: Methodological* 122: 416–435.

Yuan, K., Knoop, V. L., and Hoogendoorn, S. P. (2017) A microscopic investigation into the capacity drop: Impacts of longitudinal behavior on the queue discharge rate. *Transportation Science* 51(3): 852–862.

Zhang, H., Du, L., and Shen, J. (2022) Hybrid MPC system for platoon based cooperative lane change control using machine learning aided distributed optimization. *Transportation Research Part B: Methodological* 159: 104–142.

Zhang, J., Li, S., and Li, L. (2023) Coordinating CAV swarms at intersections with a deep learning model. *IEEE Transactions on Intelligent Transportation Systems* 24(6): 6280–6291

Zhang, K., Batterman, S., and Dion, F. (2011) Vehicle emissions in congestion: Comparison of work zone, rush hour and free-flow conditions. *Atmospheric Environment* 45(11): 1929–1939.

Zhou, Y., Cholette, M. E., Bhaskar, A., and Chung, E. (2018) Optimal vehicle trajectory planning with control constraints and recursive implementation for automated on-ramp merging. *IEEE Transactions on Intelligent Transportation Systems* 20(9): 3409–3420.

Zhou, Y., Chung, E., Bhaskar, A., and Cholette, M. E. (2019a) A state-constrained optimal control based trajectory planning strategy for cooperative freeway mainline facilitating and on-ramp merging maneuvers under congested traffic. *Transportation Research Part C: Emerging Technologies* 109: 321–342.

Zhou, Y., Chen, J., Chung, E., and Ozbay, K. (2023) CAV-Enabled Active Resolving of Temporary Mainline Congestion Caused by Gap Creation for On-Ramp Merging Vehicles. *IEEE Transactions on Intelligent Transportation Systems* 25(7): 6873–6888.

Zhou, Y., Wang, M., and Ahn, S. (2019b) Distributed model predictive control approach for cooperative car-following with guaranteed local and string stability. *Transportation Research Part B: Methodological* 128: 69–86.

Zhu, J., Tasic, I., and Qu, X. (2022) Flow-level coordination of connected and autonomous vehicles in multilane freeway ramp merging areas. *Multimodal Transportation* 1(1): 100005.