THESIS FOR THE DEGREE OF LICENTIATE OF ENGINEERING

On the challenges of reversing multi-articulated vehicles

ZHAOHUI GE

Department of Mechanics and Maritime Sciences

CHALMERS UNIVERSITY OF TECHNOLOGY

Gothenburg, Sweden 2026

On the challenges of reversing multi-articulated vehicles

ZHAOHUI GE (葛朝晖)

Thesis of Licentiate of Engineering

Department of Mechanics and Maritime Sciences
Chalmers University of Technology
SE-412 96 Gothenburg
Sweden
Telephone + 46 (0)31-772 1000

Cover:
A “Dala horse” created by paths of reversing multi-articulated vehicles

Chalmers Digital Printing
Gothenburg, Sweden 2026

On the challenges of reversing multi-articulated vehicles

ZHAOHUI GE
Department of Mechanics and Maritime Sciences
Chalmers University of Technology

ABSTRACT

Reversing articulated heavy vehicles, particularly multi-articulated ones, is a difficult task for drivers. The task is challenging in two aspects. First, the articulation angles are unstable while driving backwards, requiring active steering control from the leading unit to avoid inter-unit clashes. Second, the travel direction of the last unit is indirectly controlled, requiring a thorough understanding of the articulated vehicle's complex dynamics. Many driver-assistance systems and autonomous functions have already been developed for this or similar tasks. As many of them are based on vehicle models, two typical models of articulated vehicles are presented and discussed in detail, namely, a kinematic model and a dynamic model. The existing functions and research in reversing control are divided into three categories: articulation stabilization, path planning, and path following, which are thoroughly reviewed here.

The present thesis covers three research directions within reversing control. The first direction is a study aimed at explaining why the rearview dynamic guideline is insufficient for assisting articulated vehicles and why driver aids related to control and planning, such as those in the three categories above, are needed. The present thesis also aims to identify directions for potential improvements based on existing research and promote future developments in those areas, so that a driver-assistance system can be attractive enough for industrialization. The second direction is based on a study that develops a new geometrical method to determine the fundamental limitation of articulation stabilization for vehicles with up to three articulations. The determined limitation provides a feasible range for articulation stabilization control to enhance performance and coverage. Path-following solutions can benefit from improved articulation stabilization. Vehicle poses that fall outside the fundamental limitation can be used for the last few meters to reach final poses, which opens more possibilities for path planning. The third and final direction presents a preliminary study conducted to quantify the mismatch between the widely used kinematic model and the actual vehicle dynamics at low speed using a single-articulated vehicle. The preliminary study shows the potential to improve the performance of model-based assistant solutions from a vehicle modeling perspective, and it also promotes ongoing research on multi-articulated vehicles.

The driver aid that can fully eliminate the challenge of reversing posed by the complex motion of articulated vehicles must involve controllers that assist with path planning and following. Presenting predicted vehicle motion based on current vehicle states and driver input as dynamic guidelines, along with a top-view, can assist the driver with collision avoidance, but not with maneuvers that require changes in steering angle. Most existing research has developed path-following and planning controllers based on the kinematic model. Before those controllers can be industrialized, it is critical to determine the limit in real operations at which the vehicle deviates from the model to the extent that the control approach becomes insufficiently robust. Scenarios outside the limit need to be handled with either more accurate models or more robust control approaches. A driver aid must be able to cover sufficient scenarios that can occur in real operations to be considered well-developed and attractive for industrialization, with additional costs to the vehicle.

Keywords: articulated vehicles, reverse assistant, articulation stabilization, path following, path planning

Acknowledgements

I would like to sincerely thank my main supervisor, Fredrik Bruzelius, for his guidance throughout this project and for the many meaningful discussions both within and beyond its scope. I am also grateful to my examiner, Bengt Jacobson, for sharing his expertise in vehicle dynamics and for the sense of humor that made this journey particularly enjoyable.

Special thanks to everyone who made testing with Rhino (a fantastic dark blue Volvo FH16) on the proving ground possible. This includes my co-supervisor Ola Benderius, Fredrik von Corswant and Daniel Poveda Pi from REVERE (Resource for Vehicle Research at Chalmers), Tommi Saarikoski from Volvo Group, and all others who contributed to the test activities. I would also like to thank Vladimir Lisovskii, the driver who provided a ride in an A-double combination vehicle during real operations and demonstrated outstanding reversing techniques.

I gratefully acknowledge the support of the project's industrial partner, Volvo Group, for providing vehicle models and valuable insights and discussions throughout the project.

Furthermore, I would like to thank all students who worked with me on their theses, projects, and course assignments. Exploring new ideas together has been highly rewarding, and your questions and perspectives will continue to contribute to future teaching.

I am thankful to everyone in the Vehicle Engineering and Autonomous Systems (VEAS) Division for creating such a positive and supportive work environment.

Finally, I would like to thank my parents and friends for their encouragement and support outside this project. I am also grateful to everyone who has contributed to shaping my previous work and study experiences. Many things I learned from these experiences are also helping with this project.

This work was supported by Vinnova under Grant [2021-05027].

Zhaohui Ge
Gothenburg, March 2026

List of included papers

Paper A
Z. Ge, F. Bruzelius, and B. Jacobson, “Long combination vehicles reverse strategies based on articulation angle gradient,” in *16th International Symposium on Advanced Vehicle Control. AVEC 2024. Lecture Notes in Mechanical Engineering*, G. Mastinu, F. Braghin, F. Cheli, M. Corno, and S. M. Savaresi, Eds., Springer, Cham, 2024, pp. 707–713. doi: 10.1007/978-3-031-70392-8_100.

Paper B
Z. Ge, F. Bruzelius, and B. Jacobson, “Determining the controllable range of reversing multi-articulated vehicles using a geometrical method,” submitted for publication.

Paper C
Z. Ge, F. Bruzelius, and B. Jacobson, “Challenges and opportunities of using rearview dynamic guidelines to assist articulated vehicles in reversing,” *Automotive Innovation*, accepted.

Paper D
Z. Ge, F. Bruzelius, B. Jacobson, F. Von Corswant, D. Poveda Pi, and T. Saarikoski, “Loading effects on low-speed motions of a tractor-trailer vehicle,” presented at the 29th IAVSD Symposium on the Dynamics of Vehicles on Roads and on Tracks. IAVSD 2025, Shanghai, China, Aug. 18-22, 2025, Paper 227

Nomenclature

Abbreviations

2D	2-Dimensional
ABC	Adaptive Bi-directional Control
ABS	Anti-lock Braking System
ALBRC	Adaptive Lane-Bounded Reversing Control
CAARP	Controllable Articulation Angle-oriented Reverse Principle
CAN	Controller Area Network
CAT	Central Axle Trailer
CL-RRT	Closed-Loop Rapidly exploring Random Tree
ESC	Electronic Stability Control
FSF	Full-State Feedback
GNSS	Global Navigation Satellite System
HCT	High-Capacity Transportation
HLUT	Heuristic Look-Up Table
HMI	Human-Machine Interface
IMU	Inertial Measurement Unit
LAARP	Locked Articulation Angle-oriented Reverse Principle
LBRC	Lane-Bounded Reversing Control
LCV	Long Combination Vehicle
LQR	Linear-Quadratic Regulator
MBD	Multibody Dynamics
MIMO	Multiple-Input Multiple-Output
MPC	Model Predictive Control
ODE	Ordinary Differential Equation
OEM	Original Equipment Manufacturer
P	Proportional
PI	Proportional–Integral
PID	Proportional–Integral–Derivative
PL	Pick-and-Link
PPP	Probabilistic Path Planner
RL	Reinforcement Learning
RRT	Rapidly-exploring Random Tree
RTK	Real-Time Kinematic
SIMO	Single-Input Multiple-Output
SISO	Single-Input Single-Output
SSCL	Steady-State Circling Limitation
VFO	Vector Fields Orientation
VTM	Volvo Transport Model

Notations

General subscripts

Unless specified separately, the following meaning applies to symbols in the present thesis.

Subscript	Values and meaning
$\langle dir \rangle$	x (longitudinal), y (lateral).
$\langle i \rangle$	$1, 2, 3, \cdots$. Unit index counting from the leading unit, which is the frontmost unit while an articulated vehicle is driving forward.
$\langle j \rangle$	$1, 2, 3, \cdots$. Axle index counting from the frontmost axle in the positive longitudinal direction of the unit.
$\langle k \rangle$	1 (left), 2 (right).
$\langle m \rangle$	1 (front), 2 (rear).
$\langle pos \rangle$	c (coupling), e (equivalent axle).

Symbol	Unit	Meaning
$a_{\langle dir \rangle \langle i \rangle}$	$\mathrm{m/s^2}$	$\langle dir \rangle$ acceleration at the center of gravity of unit $\langle i \rangle$ in the local coordinate of unit $\langle i \rangle$.
A	m	Lateral distance from the local origin to the center of rotation of a unit in its local coordinate system.
$C_{\langle i \rangle}$	-	Control point fixed in the local coordinate system of a unit. $\langle i \rangle = 1, n, nl$, see definition in Figure 3-3.
d	-	Ordinary differential
∂	-	Partial differential
f	-	Function
$F_{\langle dir \rangle \langle i \rangle c \langle m \rangle}$	N	$\langle dir \rangle$ coupling force at the $\langle m \rangle$ coupling of unit $\langle i \rangle$ in the local coordinate system of unit $\langle i \rangle$.
$F_{\langle dir \rangle t \langle i \rangle \langle j \rangle \langle k \rangle}$	N	$\langle dir \rangle$ tire force of tire(s) on the $\langle k \rangle$ side of axle $\langle j \rangle$ on unit $\langle i \rangle$ in the tire coordinate system.
$\boldsymbol{F}_{\langle dir \rangle t}$	N	$\langle dir \rangle$ tire force vector, includes every $F_{\langle dir \rangle t \langle i \rangle \langle j \rangle \langle k \rangle}$.
h	-	Function
$J_{\langle i \rangle}$	$\mathrm{kg \cdot m^2}$	Yaw inertial of unit $\langle i \rangle$ around its center of gravity.
$l_{\langle i \rangle a \langle j \rangle}$	m	Longitudinal displacement from the center of gravity of unit $\langle i \rangle$ to axle $\langle j \rangle$ of unit $\langle i \rangle$ in the local coordinate system of unit $\langle i \rangle$.
$l_{\langle i \rangle c \langle m \rangle}$	m	Longitudinal displacement from the local origin to the $\langle m \rangle$ coupling point of unit $\langle i \rangle$ in the local coordinate system of unit $\langle i \rangle$. When used with $l_{\langle i \rangle a \langle j \rangle}$, the local origin is the center of gravity of unit $\langle i \rangle$.
$l_{\langle i \rangle e \langle m \rangle}$	m	Longitudinal displacement from the local origin to the $\langle m \rangle$ equivalent axle of unit $\langle i \rangle$ in the local coordinate system of unit $\langle i \rangle$.
$l_{\langle i \rangle gw}$	m	Geometric wheelbase of unit $\langle i \rangle$, which is the longitudinal displacement from the rear geometric equivalent axle to the font geometric equivalent axle ($\langle i \rangle = 1$) or the front coupling ($\langle i \rangle = 2, 3, \cdots$) of unit $\langle i \rangle$ in the local coordinate system of unit $\langle i \rangle$.
$l_{\langle i \rangle IMU}$	m	Longitudinal displacement from the local origin to the IMU on unit $\langle i \rangle$ in the local coordinate system of unit $\langle i \rangle$.
$l_{\langle i \rangle t \langle j \rangle}$		Trackwidth of axle $\langle j \rangle$ on unit $\langle i \rangle$.
l_{la}	m	Lookahead distance from a given control point in the local longitudinal travel direction of the unit where the control point is fixed on.

Symbol	Unit	Meaning
$\boldsymbol{l}_{\langle p\rangle}$	m	Vehicle longitudinal dimension set that includes $l_{\langle i\rangle c\langle j\rangle}$ and $l_{\langle i\rangle e\langle j\rangle}$ from unit 1 ($\langle i\rangle = 1$) to unit $\langle p\rangle$ ($\langle i\rangle = \langle p\rangle$). $\langle p\rangle = 1, 2, 3, \cdots$, unit index counting from the leading unit.
l_R	m	Lookahead radius for determining a reference point on a reference path from a control point.
m_i	kg	Mass of unit $\langle i\rangle$.
$\boldsymbol{M}_{zt}$	Nm	Tire yaw moment vector, includes every $M_{zt\langle i\rangle\langle j\rangle\langle k\rangle}$.
$M_{zt\langle i\rangle\langle j\rangle\langle k\rangle}$	Nm	Tire yaw moment of tire(s) on the $\langle k\rangle$ side of axle $\langle j\rangle$ on unit $\langle i\rangle$ in the tire coordinate system.
n	-	Number of units within an articulated vehicle.
n_a	-	Number of axles within an articulated vehicle.
$n_{a\langle i\rangle}$	-	Number of axles on unit $\langle i\rangle$.
$R_{\langle i\rangle}$ or $R_{\langle i\rangle e2}$	m	Instantaneous turning radius at the rear equivalent axle center of unit $\langle i\rangle$.
$R_{\langle i\rangle\langle j\rangle}$	m	Instantaneous turning radius at the center of axle $\langle j\rangle$ on unit $\langle i\rangle$.
$R_{n\langle i\rangle}$	m	Reference point on a reference path. $\langle i\rangle = n1, n2, n3$, see definition in Figure 3-3.
RC	-	Rotational center.
$RC_{\langle i\rangle}$	-	Rotational center of unit $\langle i\rangle$.
s_1	m	Vehicle position along the path coordinate of the rear equivalent axle center of Unit 1.
$\mathrm{sgn}(x)$	-	The sign of variable x.
t	s	time
T	-	Transform matrix.
T^{-1}	-	Inverse of matrix T.
u	-	Input vector
u_0	-	Input vector at the linearization point
$v_{\langle dir\rangle\langle i\rangle}$	m/s	$\langle dir\rangle$ velocity at the local origin of unit $\langle i\rangle$ in the local coordinate of unit $\langle i\rangle$.
$v_{\langle i\rangle c\langle m\rangle}$	m/s	$\langle m\rangle$ coupling velocity on unit $\langle i\rangle$ in the local coordinate system of the unit.
$v_{\langle i\rangle c\langle m\rangle\langle dir\rangle}$	m/s	$\langle m\rangle$ coupling velocity on unit $\langle i\rangle$ in the $\langle dir\rangle$ direction in the local coordinate system of the unit.
$v_{\langle i\rangle te\langle m\rangle\langle dir\rangle}$	m/s	$\langle dir\rangle$ velocity of unit $\langle i\rangle$ at its $\langle m\rangle$ equivalent axle in the tire coordinate system.
$\overline{v_{x1}}$	m/s	Mean longitudinal velocity of the leading unit in its local coordinate system.
$V_{\langle i\rangle c\langle m\rangle}$	m/s	$\langle m\rangle$ coupling velocity on unit $\langle i\rangle$ in the global coordinate system.
x	-	State vector
x_0	-	State vector at the linearization point
$x_{\langle i\rangle} - y_{\langle i\rangle}$	-	Local coordinate system of unit $\langle i\rangle$ in the horizontal plane
$x_{\langle i\rangle te\langle m\rangle} - y_{\langle i\rangle te\langle m\rangle}$	-	Tire coordinate system on the $\langle m\rangle$ equivalent axle of unit $\langle i\rangle$.
$X_{\langle i\rangle\langle pos\rangle\langle m\rangle}$	m	Longitudinal position of the $\langle m\rangle$ $\langle pos\rangle$ of unit $\langle i\rangle$ in the global coordinate system
$X - Y$	-	Global coordinate system in the horizontal plane
y	-	Output vector
$Y_{\langle i\rangle\langle pos\rangle\langle m\rangle}$	m	Lateral position of the $\langle m\rangle$ $\langle pos\rangle$ of unit $\langle i\rangle$ in the global coordinate system.
$\beta_{\langle i\rangle c1}$	rad	Virtual steering angle at the front coupling of unit $\langle i\rangle$ ($\langle i\rangle \geq 2$) from the longitudinal direction of the unit's local coordinated system
δ	rad	Roadwheel steering angle of the front equivalent axle on unit 1 in the local coordinate system of unit 1.

Symbol	**Unit**	**Meaning**
$\dot{\delta}$	rad/s	Roadwheel steering rate of the front equivalent axle on unit 1 in the local coordinate system of unit 1.
δ_{ff}	rad	Roadwheel steering angle request from a feedforward controller.
δ_{fb}	rad	Roadwheel steering angle request from a feedback controller.
δ_H	°	Handwheel steering angle.
$\delta_{\langle i\rangle e\langle m\rangle}$	rad	Roadwheel steering angle of the $\langle m\rangle$ equivalent axle on unit $\langle i\rangle$ in the local coordinate system of unit $\langle i\rangle$.
$\delta_{\langle i\rangle\langle j\rangle\langle k\rangle}$	rad	Roadwheel steering angle of the tire on the $\langle k\rangle$ side of axle $\langle j\rangle$ on unit $\langle i\rangle$ in the local coordinate system of unit $\langle i\rangle$.
δ_{max}	°	Maximum roadwheel steering angle limited by the steering system
δ_{min}	°	Minimum roadwheel steering angle limited by the steering system
θ	rad	Current articulation angle of a single-articulated vehicle, which is the yaw angle from the longitudinal axis of the trailer to the longitudinal axis of the leading unit.
$\boldsymbol{\theta}$	rad	Current articulation angle vector, which includes all current articulation angles in an articulated vehicle.
$\theta_{1,CAARP}$	rad	Upper CAARP articulation angle limit for a single articulated vehicle.
$\boldsymbol{\theta}_{base}$	rad	Set of vehicle poses on the boundary of a CAARP space before expansion
$\boldsymbol{\theta}_{CAARP}$	rad	Closed vehicle pose set capable for CAARP
$\boldsymbol{\theta}_{CAARP,0}$	rad	Initial Closed vehicle pose set capable for CAARP
$\boldsymbol{\theta}_{CAARP,Boundary}$	rad	Set of vehicle poses on the boundary of the largest CAARP space for the vehicle.
$\boldsymbol{\theta}_{extend}$	rad	Set of vehicle poses on the boundary of a new CAARP space expanded from $\boldsymbol{\theta}_{base}$.
$\theta_{\langle i\rangle}$	rad or °	Articulation angle, yaw angle from the longitudinal axis of unit $\langle i+1\rangle$ to the longitudinal axis of unit $\langle i\rangle$. $\langle i\rangle = 1, 2, 3, \cdots$, articulation joint index counting from the leading unit. ° is only used as the unit in figures.
$\dot{\theta}_{\langle i\rangle}$	rad/s	Articulation angle rate.
$\ddot{\theta}_{\langle i\rangle}$	rad/s^2	Articulation angle acceleration.
$\boldsymbol{\theta}_{\langle p\rangle}$	rad	Articulation angle set that includes $\theta_{\langle i\rangle}$ from $\langle i\rangle = 1$ to $\langle i\rangle = \langle p\rangle$.
θ_{max}	°	Maximum articulation angle limited by the mechanical design
θ_{min}	°	Minimum articulation angle limited by the mechanical design
θ_r	rad	Reference articulation angle of a single-articulated vehicle.
$\boldsymbol{\theta}_r$	rad	Reference articulation angle vector, which includes all reference articulation angles for an articulated vehicle.
$\Delta l_{\langle i\rangle v\langle m\rangle}$	m	Longitudinal displacement from the $\langle m\rangle$ geometric equivalent axle to its corresponding virtual axle position that has zero sideslip on unit $\langle i\rangle$ in the local coordinate system of unit $\langle i\rangle$.
$\Delta\mathrm{s}$	m	Step size of s_1 in simulations
$\Delta\delta$	rad	Step size oof discretized roadwheel steering angle
$\Psi_{\langle i\rangle}$ or $\omega_{\langle i\rangle}$	rad	Yaw (heading) angle of unit $\langle i\rangle$ in the global coordinate system.
$\dot{\Psi}_{\langle i\rangle}$	rad/s	Yaw rate of unit $\langle i\rangle$ in the global coordinate system.
$\ddot{\Psi}_{\langle i\rangle}$	rad/s^2	Yaw acceleration of unit $\langle i\rangle$ in the global coordinate system.
$\forall$	-	For all
$\emptyset$	-	Empty set
$\exists$	-	There exists
$\subset$	-	Subset of
$[\boldsymbol{x}]^T$	-	Transpose of vector $\boldsymbol{x}$.

Table of Contents

ABSTRACT i
Acknowledgements iii
List of included papers iv
Nomenclature v
Abbreviations v
Notations vi
Chapter 1 Introduction 1
1.1 Background 1
1.2 Motivation 2
1.3 Research questions 3
1.4 Limitations 4
1.5 Contributions 4
Chapter 2 Vehicle models 7
2.1 Kinematic model 7
2.2 Dynamic model 11
2.3 High-fidelity model 16
2.4 Summary and discussion 16
Chapter 3 Reversing control 19
3.1 Articulation stabilization 19
3.2 Path following 23
3.3 Path planning 32
3.4 Summary and discussion 41
Chapter 4 Summary and discussion of included papers 43
4.1 Paper A 43
4.2 Paper B 44
4.3 Paper C 45
4.4 Paper D 46
Chapter 5 Conclusion and outlook 49
5.1 Outlook 50
References 53
List of licentiate theses from Chalmers Vehicle Dynamics group 61

Chapter 1
Introduction

Road transport plays a crucial role in freight transportation. More than half of all transported goods are transported by road in many countries [1]. Many of these goods are transported by tractor-trailers, which are articulated vehicles made up of a tractor and a semi-trailer. Increasing the vehicles' load capacity is a way to boost transportation efficiency and lower emissions per ton-kilometer. One solution is to attach additional trailers to a vehicle to form a long combination vehicle (LCV), increasing the volume and weight of goods transported per vehicle. An increased number of articulation joints poses a challenge for drivers to reverse those vehicles. Various assistant and autonomous driver aids for reversing articulated vehicles have been developed in both academia and industry, though they come with certain limitations. This thesis aims to identify gaps between existing solutions and potential real-world applications, paving the way for commercially available solutions to address the challenges of reversing articulated vehicles. This thesis also provides a high-level view of the connections between the works done in the included papers and the challenges of reversing articulated vehicles.

1.1 Background

Since December 1, 2023, Sweden has increased the length limit for trucks from 25.25 meters to 34.5 meters, allowing LCVs that meet specific requirements with up to three articulation joints to operate on certain parts of the Swedish road network [2]. Similar LCVs are also used in other countries, such as Finland, Denmark, Germany, Canada, the United States, Australia, etc. Figure 1-1 shows nine examples of articulated trucks from [3]. LCVs in Figure 1-1.b to e are within the 25.25-meter limit with two articulation joints. LCVs no longer than 34.5 meters in length are shown in Figure 1-1.f to i, they can have two or three articulation joints.

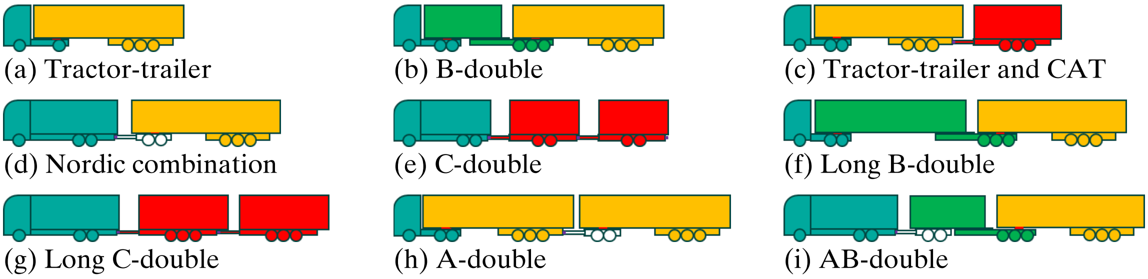
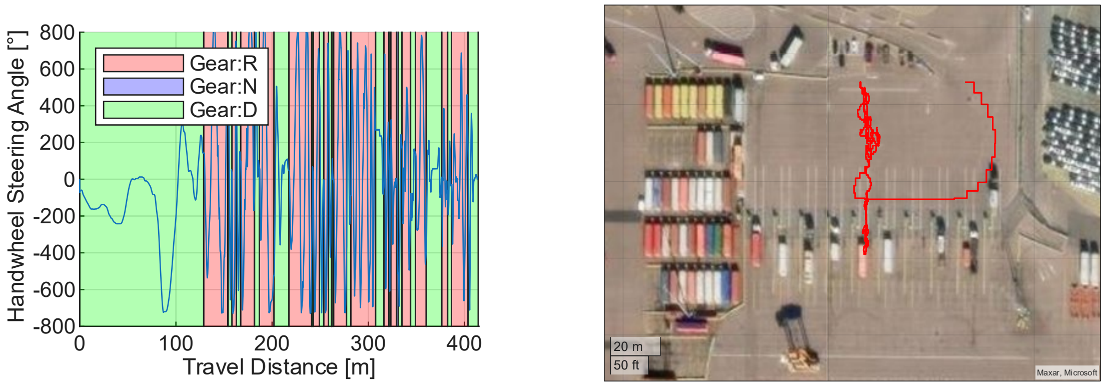

(a) Tractor-trailer
(b) B-double
(c) Tractor-trailer and CAT
(d) Nordic combination
(e) C-double
(f) Long B-double
(g) Long C-double
(h) A-double
(i) AB-double

Figure 1-1 Articulated trucks

From a pilot project using LCVs in Sweden, reverse maneuvers have been identified as a challenging task for drivers [4]. One test vehicle in the project is an A-double truck, similar to the one shown in Figure 1-1.h. The A-double consists of a tractor and two semi-trailers with a dolly in between. It also contains two fifth-wheel couplings and one drawbar coupling. When the vehicle needs to load or unload at specific locations in the Port of Gothenburg, drivers must reverse the entire triple-articulated vehicle into the loading positions. Figure 1-2 shows how an experienced driver controls an A-double to complete the reverse task. The driver changed direction 32 times and frequently made large steering adjustments. The driver has extensive experience reversing the A-double combination and completed the task within 8 minutes, as this is a daily task at the harbor in this pilot project. The safety regulation at the harbor makes the reverse task even harder as the driver is not allowed to leave the cab to check the vehicle's position from outside. Feedback from haulage in the pilot project indicated that some drivers refused the job because they were required to perform similar tasks. The reverse challenge is not unique to the Port of Gothenburg; even in newly built ports, external trucks are required to reverse into loading and unloading spaces. Most warehouses also have loading docks designed for perpendicular parking, allowing more vehicles to connect to the warehouse at the same time. Many trailers and containers are not designed for side loading, which means they only support perpendicular docking. Infrastructure-wise, it will be costly, or even infeasible, for the transportation industry to implement changes and eliminate reverse tasks for LCVs at terminals.



(a) Gear and steering angle (b) Tractor path

Figure 1-2 Driver's inputs and tractor path when parking an A-double

When the A-double in the pilot project is set to deliver containers to warehouses, drivers will detach the dolly and the last trailer at a shunting yard of the haulage company and then drive out as a tractor-trailer to deliver containers one by one to end customers. The pilot project for A-double trucks in Denmark prevents the entire vehicle from going to freight terminals by establishing decoupling areas near motorways [5]. However, the extra coupling and decoupling are strenuous tasks that drivers do not prefer. Moreover, decoupling only partially weakens the challenge in reverse LCVs.

At a grocery chain in Sweden, tractor-trailers (Figure 1-1.a) and Nordic combinations (Figure 1-1.d) are used to deliver goods from the warehouse to multiple local stores. The company observed that new drivers might need more time to dock at each store, preventing them from reaching all their assigned stores within their shift. Even for experienced tractor-trailer drivers, maneuvering in tight spaces may require getting out of the cab on their own or asking colleagues to stand outside to guard blind spots. The challenge in reverse is not unique to multi-articulated trucks but applies to all articulated vehicles, which also includes passenger cars with trailers.

In general, reversing articulated vehicles is a challenging task that will remain part of articulated vehicle operations. It not only means reduced transport efficiency but also increases safety risks [6]. A well-developed assistance or autonomous driver aid will reduce or eliminate the challenges of reversing for drivers and even promote the adoption of LCVs.

1.2 Motivation

Developing assistant aids for reversing articulated vehicles is not a new topic. Some existing research and industrialized functions are listed in Table 1-1 and divided into three categories. First, articulation stabilization control vehicles approach their unstable steady states in response to a driver's input, typically a turning radius. Second, path-following provides the required steering angle to follow predefined paths based on the vehicle's current state. Third, path-planning produces feasible paths that a vehicle can follow without internal and external collisions. Detailed discussion and examples for all three types of solutions are presented in Chapter 3.

One observation from Table 1-1 is that, as of the time of writing, industrialized aids for reversing articulated vehicles known to the author are only articulation stabilization functions for single-articulated vehicles. More specifically, they are all from passenger vehicles or pick-up trucks and are designed for central axle drawbar trailers. More advanced assistance than other industrialized aids is offered in [7], including maintaining the trailer's heading and changing it by 90 degrees. However, it doesn't count as path following because it doesn't show the path to drive before the maneuver. The limited applications of industrialized aids do not mean research are far from being usable in real operations. For example, specially adapted tractor-trailers can operate autonomously with reversing tasks at a specially adapted mining site [8]. Commercializing and industrializing research involves many practical problems, such as articulation angle measurements and vehicle localization. Instead of focusing on the practical aspect, this study concentrates on the functionality aspect of existing research.

Today, ABS (Anti-lock Braking System) and ESC (Electronic Stability Control) are standard features in most modern vehicles due to their safety benefits. With those systems, wheel speed sensors and steering angle sensors are considered standard components. The redundancy requirements in autonomous vehicles promote OEMs (Original equipment manufacturers) and suppliers to have multiple wheel speed sensors or a redundant wheel speed sensor at each wheel. The industrialization of the two systems is an example of how the automotive industry would adopt additional systems as standard configurations if OEMs or related authorities recognized their performance. This motivates the present thesis to focus on identifying areas for improvement based on existing research. Future development will focus on the identified critical areas for real-world operations and on improving the performance of the assistant or autonomous solutions to a level that will not hinder industrialization by additional costs on vehicles and infrastructure.

Table 1-1 Existing solutions related to reversing articulated vehicles

Technology readiness		Industrialized	Research		
Number of units		2	2	3	≥4
Solution type	Articulation stabilization	[7], [9], [10], [11]	[12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23], [24], [25]	[13], [18], [21]	[13], [18], [21], [26]
	Path following		[27], [28], [29], [30], [31], [32], [33], [34], [35], [36], [37], [38], [39], [40], [41], [42], [43], [44], [45], [46], [47], [48], [49], [50], [51], [52], [53], [54], [55], [56], [57], [58], [59], [60], [61], [62], [63], [64], [65], [66], [67], [68], [69], [70], [71], [72], [73], [74], [75]	[27], [32], [34], [37], [40], [44], [45], [46], [52], [53], [59], [62], [63], [68], [73], [75], [76], [77], [78], [79], [80], [81], [82], [83], [84], [85], [86]	[27], [34], [37], [40], [44], [45], [46], [52], [53], [62], [63], [68], [73], [74], [75], [76], [79], [87]
	Path planning		[29], [31], [44], [45], [49], [51], [52], [58], [61], [62], [63], [66], [67], [68], [69], [74], [88], [89], [90], [91], [92], [93], [94], [95]	[44], [45], [52], [62], [63], [74], [81], [86], [88], [89], [92], [93], [96], [97], [98]	[44], [45], [52], [62], [63], [68], [74], [88], [89], [92], [93]

1.3 Research questions

This study is based on a large amount of existing research related to reversing articulated vehicles, as listed in Table 1-1. The listed works are not limited to articulated trucks but also include related solutions from passenger vehicles and robotics. When studying those, differences between the vehicles used by other researchers and the typical articulated trucks in Figure 1-1 are always considered. Meanwhile, industrial partners within AUTOFREIGHT 2 [4] have provided insight into the demands of reversing articulated vehicles in real-world operations. By looking into the existing research from both theoretical and practical perspectives, this thesis addresses the following research questions:

- What types of vehicle models are used in existing research? What are the advantages and disadvantages of those various types when used in designing reversing aids? Especially for the widely used kinematic model, how it will diverge from the actual vehicle motion at low speeds.
- What types of solutions are available? Which scenarios do they handle well, and which ones are too challenging for them? How does the motion of the articulated vehicle affect the usability of driver aids and make certain scenarios impossible to solve with an aid due to fundamental limitations of articulated vehicles?
- Where do the gaps exist between current research and the actual needs of real-world operations?

1.4 Limitations

In the present thesis, the primary focus is on articulated vehicles with steering actuation only on the front axle group of their leading unit. The leading unit also has a non-steerable rear axle group. All trailing units have only one non-steerable axle group each. This axle configuration applies to all the included papers. The kinematic and dynamic models presented in Chapter 2 are ready to be expanded to account for additional steering axles. A small amount of existing research discussed in Chapter 3 uses actuators more than the front axle steering. Vehicle states necessary for driver aids are assumed to be available, including but not limited to articulation angles, positions, and velocities. The accessibility of those states on existing vehicles is not considered in this thesis. The design of the HMI (Human-Machine Interface) between driver aids and drivers is also not included in this thesis. It is an essential part not only to assist functions that help drivers to control the vehicle, but also to autonomous functions in early stages for better supervision.

1.5 Contributions

The contributions of the present thesis are as follows:

- Divide articulated vehicle poses, defined by all articulation angles within a vehicle, into controllable and uncontrollable ranges based on whether the vehicle can reverse into a straight pose without inter-unit clashes. Articulation angle gradients, which are the changes of articulation angles with respect to the travel distance of the tractor in rad/m, are found to be potentially useful for steering guidance in reversing articulated vehicles to a straight pose [Paper A]. A geometrical approach based on articulation angle gradients is developed to identify the boundary between the controllable and uncontrollable ranges [Paper B]. The identified controllable range is the feasible range for an ideal articulation stabilization control and can guide improvements to existing articulation stabilization controllers. Improvements in articulation stabilization control can further improve path-following and path-planning aids built on it, and result in better aids for reversing articulated vehicles.
- The rearview dynamic guidelines, which are the predicted paths for corners and wheels on the vehicle shown with the real-time video from a rearview camera or a top-view system, offer great assistance for reversing non-articulated vehicles can only assist articulated vehicles in a limited way. This is mainly due to the absence of a one-to-one relationship between steering angles and reversing paths in articulated vehicles, which means that the challenges of estimating and planning while reversing remain with drivers [Papers C]. Therefore, to eliminate the difficulties for drivers in understanding the complex motion of

articulated vehicles when reversing, it is necessary to develop a driver aid capable of handling both path planning and path following.

- The inverse kinematic method, which calculates the steering angle of the first unit from a turning radius on the last unit through the kinematic chain, has limited assistance ability for reversing several typical articulated vehicles. The feasible turning radii of the last unit within the steering limit can be heavily affected and limited by the vehicle's current articulation angles [Paper C]. Using the inverse kinematic method for path following imposes strict constraints on both the reference path and the vehicle configuration, making it feasible only for path planners that account for nearly all nonholonomic constraints and articulated vehicles without on-axle hitches. This shows the necessity of using a path-following control that can overcome the limitations of the inverse kinematic method in a driver aid, making the aid applicable to more general vehicles and imperfectly planned reference paths.
- Tests on real vehicles with a tractor-trailer at low speeds on dry asphalt show that the longitudinal offsets between equivalent zero-sideslip axles and the physical axles or the centers of bogies can be more than 20% of the tractor's wheelbase [Paper D]. This observation encourages further research on the deviations between the kinematic model and multi-articulated vehicles. Understanding these deviations is critical to defining the feasible boundary for many driver aids in existing research, as most are developed based on kinematic models. The understanding can also facilitate the development of existing aids by improving their coverage of different scenarios.

The connections between the included papers are shown in Figure 1-3. All papers use the kinematic model that is introduced in chapter 2.1. The link between articulation angle gradients and the controllability of a vehicle pose is observed in Paper A. In Paper B, a method that defines the controllable limit based on articulation angle gradients is presented. A high-fidelity model is used to verify the method developed using the kinematic model. The deviation between the two models observed in Paper B leads to Paper D with real vehicle tests. Results from the kinematic model and a high-fidelity model are compared with measurements from the real vehicle tests in Paper D. Paper C is not directly connected to the rest of the papers and focuses on the opportunity of using a reversing aid from non-articulated vehicles on articulated vehicles.

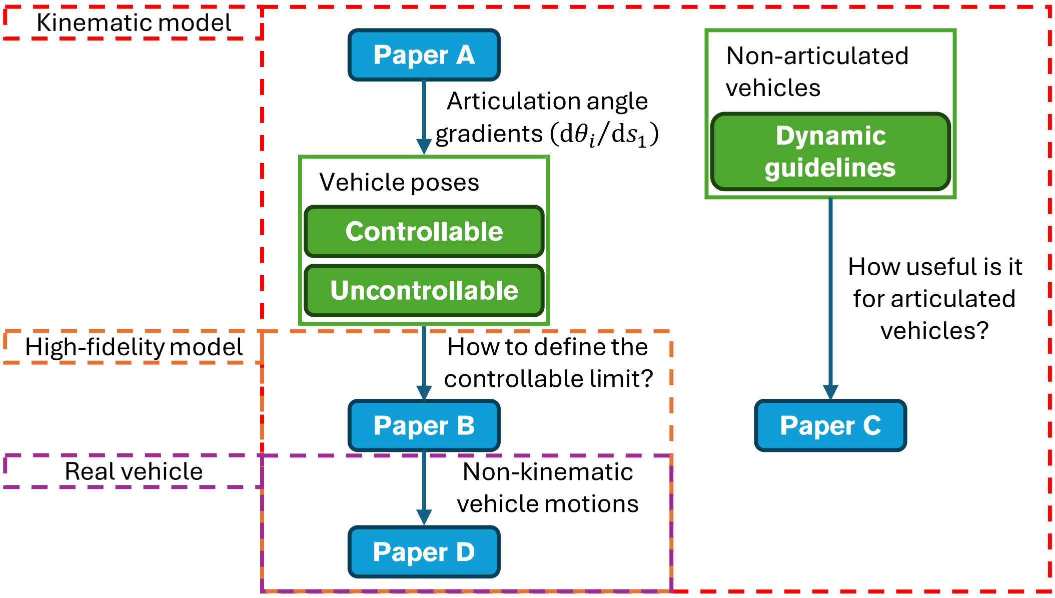

Kinematic model
Paper A
Articulation angle gradients ($d\theta_i/ds_1$)
Vehicle poses
Controllable
Uncontrollable
Non-articulated vehicles
Dynamic guidelines
High-fidelity model
How to define the controllable limit?
Paper B
How useful is it for articulated vehicles?
Paper C
Real vehicle
Non-kinematic vehicle motions
Paper D

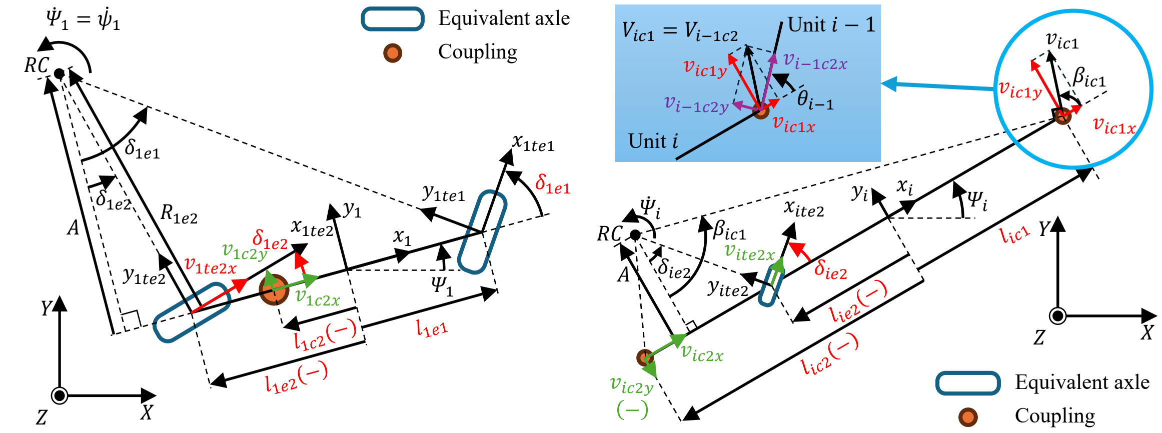
Figure 1-3 Connections between included papers and used models

Chapter 2 Vehicle models

Vehicle models are used in various ways when researching the reverse operation of multi-articulated vehicles. Some model-free controllers are designed with model-based tuning to reach desired performance. Model-based controllers are widely used to solve SIMO (Single-Input Multiple-Output) and MIMO (Multiple-Input Multiple-Output) problems. Models and simulations are used to validate developed controllers in virtual environments. This chapter introduces the three most common types of models used to develop reversing controllers, referred to as kinematic, dynamic, and high-fidelity models. In this study, the leading unit refers to the unit at the front of an articulated vehicle in the forward driving direction, such as a tractor or a rigid truck. Trailing units are the units in an articulated vehicle other than the leading unit. The last trailing unit is the unit at the rearmost of an articulated vehicle when counting from the leading unit. These names do not change with the driving direction.

2.1 Kinematic model

The kinematic model is among the most commonly used in the development of reversing control for articulated vehicles. The main assumption of the kinematic model is that there is no tire sideslip, which is reasonable for a typical two-axle vehicle with perfect Ackermann steering on flat ground and small rolling resistances. However, this assumption imposes simplifications on typical heavy vehicle configurations, such as bogie axles. Usually, the first simplification converts vehicles into single-track planar vehicles by replacing each axle with a single tire at its center and considering only horizontal-plane motions. Then proceed to the leading unit. Leading units are rigid enough that the whole unit should be assumed to have a common center of rotation while cornering. As shown in Figure 2-1.a, the rotational center of a two-axle vehicle is located at the intersection of the two lateral axes of the tires. Any additional axle without its lateral axis passing through the rotational center will cause the vehicle to have more than one rotational center under the zero-sideslip assumption, which violates the rigid body assumption. Such an additional axle can be either a non-steerable axle or a steerable axle that doesn’t follow the Ackermann geometry. Therefore, the leading unit of an articulated truck can have up to two independent steering axles in a kinematic model. Two typical methods for determining the positions of zero-sideslip equivalent axles are described in Paper D. Once the motion of the two-axle lending unit is determined, it also determines the motion of the coupling point between the two-axle lending unit and the trailing unit behind it. According to kinematics, the rotational center of the trailing unit must be on the line passing through the trailer’s front coupling and perpendicular to the coupling’s velocity. The exact position of the rotational center on the line is then determined by the single axle on the trailing unit, at the point when the tire’s lateral axis crosses the line, which is shown in Figure 2-1.b. Meanwhile, the motion of the rear coupling is determined. Then, the motions of the remaining trailing units are determined using the same kinematic chain.



(a) Two-axle leading unit (b) Single-axle trailing unit

Figure 2-1 Vehicle units in a single-track kinematic model

The following equations show how to build a general kinematic model for a multi-articulated vehicle formed with a two-axle leading unit and multiple single-axle trailing units. The model is general, as there is no limit on the coupling positions and all axles are steerable. Assume the known inputs of a n-unit articulated vehicle system are the longitudinal velocity of the leading unit's rear equivalent axle in its tire coordinate system (v_{1te2x}), and the steering angles of all $n+1$ equivalent axles ($\delta_{1e1}, \delta_{1e2}, \delta_{2e2}, \cdots, \delta_{ne2}$). The following geometrical relationship can be seen in Figure 2-1.a:

$$A \tan \delta_{1e1} - A \tan \delta_{1e2} = l_{1e1} - l_{1e2} \tag{2-1}$$

where A is the displacement from the unit's longitudinal axle to the unit's rotational center, a positive A indicate the rotational center is on the positive lateral direction of the unit; l_{1e1} and l_{1e2} are displacements from the origin to the equivalent front and rear axles, respectively, a positive l means the axle or coupling is in the positive longitudinal direction of the unit's origin.

The yaw rate of the leading unit is calculated using A as an intermediate variable by:

$$\dot{\Psi}_1 = \frac{v_{1te2x}}{R_{1e2}} = v_{1te2x} \cdot \frac{\cos \delta_{1e2}}{A} = \frac{v_{1te2x}(\tan \delta_{1e1} - \tan \delta_{1e2}) \cos \delta_{1e2}}{l_{1e1} - l_{1e2}} \tag{2-2}$$

where $\dot{\Psi}_1$ is the yaw rate of the leading unit in the global coordinate system, R_{1e2} is the turning radius at the rear equivalent axle of the leading unit.

The translational velocity in the unit's coordinate system at any point on the rigid leading unit can be expressed with v_{1te2x}, δ_{1e2}, $\dot{\Psi}_1$, and related dimensional parameters. The translational velocities of the coupling point (v_{1c2x} and v_{1c2y}) are:

$$\begin{cases} v_{1c2x} = v_{1te2x} \cos \delta_{1e2} \\ v_{1c2y} = v_{1te2x} \sin \delta_{1e2} + \dot{\Psi}_1 (l_{1c2} - l_{1e2}) \end{cases} \tag{2-3}$$

The position and heading of the leading unit in the global coordinate system are given by integrations:

$$\begin{cases} \Psi_1(t) = \int_0^t \dot{\Psi}_1(t) dt + \Psi_1(0) \\ X_{1c2}(t) = \int_0^t \big(v_{1c2x}(t) \cos \Psi_1(t) - v_{1c2y}(t) \sin \Psi_1(t)\big) dt + X_{1c2}(0) \\ Y_{1c2}(t) = \int_0^t \big(v_{1c2x}(t) \sin \Psi_1(t) + v_{1c2y}(t) \cos \Psi_1(t)\big) dt + Y_{1c2}(0) \end{cases} \tag{2-4}$$

where $\Psi_1(t)$, $X_{1c2}(t)$ and $Y_{1c2}(t)$ are the heading angle, longitudinal and lateral position of the coupling at time t, respectively. It is possible to calculate the heading and position of another point of the unit by knowing its relative position to the coupling.

Knowing the motions of the leading unit, the motion of the trailing unit can be calculated in sequence from the first to the last trailer. For the ith unit ($i = 2, 3, \cdots, n-1$) in the articulated vehicle, its front coupling has the same translational velocity as the rear coupling of the i-1th unit in the global coordinate system. This means the front coupling velocity of the ith unit in its global coordinate system (v_{ic1x} and v_{ic1y}) is:

$$\begin{cases} v_{ic1x} = v_{i-1c2x}\cos\theta_{i-1} - v_{i-1c2y}\sin\theta_{i-1} \\ v_{ic1y} = v_{i-1c2x}\sin\theta_{i-1} + v_{i-1c2y}\cos\theta_{i-1} \end{cases} \tag{2-5}$$

where $\theta_{i-1} = \Psi_{i-1} - \Psi_i$ is the articulation angle between the ith and the i-1th units.

Introduce a virtual steering axle that follows the zero-sideslip assumption at the front coupling, meaning the tire's longitudinal direction matches the velocity direction of the front coupling, then its virtual steering angle (β_{ic1}) is:

$$\beta_{ic1} = \text{atan}\frac{v_{ic1y}}{v_{ic1x}} \tag{2-6}$$

By combining the virtual axle and the original equivalent axle on the single-axle trailing unit, the motions of the trailing units can be calculated in the same way as the two-axle leading unit. Equation 2-1 is updated as:

$$A\tan\beta_{ic1} - A\tan\delta_{ie2} = l_{ic1} - l_{ie2} \tag{2-7}$$

where the original equivalent axle on the trailer is treated as the rear axle on a two-axle unit with a subscript $e2$, δ_{ie2} is the steering angle of the equivalent axle, l_{ic1} and l_{ie2} are displacements from the origin to the front coupling and the equivalent axle, respectively.

The turning radius at the equivalent axle (R_{ie2}) can be calculated with the same geometrical relationship that is used in Equation 2-2, which gives:

$$R_{ie2} = \frac{A}{\cos\delta_{ie2}} \tag{2-8}$$

The longitudinal velocity of the equivalent axle in its tire coordinate system (v_{ite2x}) can be expressed with the front coupling velocity under the assumptions of a rigid vehicle body and zero sideslip as:

$$v_{ite2x} = \frac{v_{ic1x}}{\cos\delta_{ie2}} \tag{2-9}$$

Then, the yaw rate of the unit ($\dot{\Psi}_i$) is:

$$\dot{\Psi}_i = \frac{v_{ite2x}}{R_{ie2}} = \frac{v_{ic1x}(\tan\beta_{ic1} - \tan\delta_{ie2})}{l_{ic1} - l_{ie2}} \tag{2-10}$$

With knowing the yaw rate, the rear coupling velocity in the unit's coordinate system (v_{ic2x} and v_{ic2y}) can be calculated from the front coupling velocity:

$$\begin{cases} v_{ic2x} = v_{ic1x} \\ v_{ic2y} = v_{ic1y} + \dot{\Psi}_i(l_{ic2} - l_{ic2}) \end{cases} \tag{2-11}$$

Depending on the purpose, state-space models with different state variables can be built for articulated vehicles using Equations 2-1 to 11. To describe the position and pose of the vehicle in the global coordinate system, the state vector needs to contain at least $n+2$ states for a n-unit articulated vehicle. For example:

$$x = [X_{1c2} \quad Y_{1c2} \quad \Psi_1 \quad \theta_1 \quad \theta_2 \quad \cdots \quad \theta_{n-1}]^T \tag{2-12}$$

where X_{1c2}, Y_{1c2} and Ψ_1 describe the location and orientation of the leading unit, θ_i describe the location and orientation of the trailing unit in relation to its neighboring units.

The focus of Papers A and B is whether the vehicle can reverse without inter-unit clashes under both the steering angle and articulation angle limits; therefore, the analysis is based on articulation angles. The state vector used in these two papers only contains all articulation angles:

$$x_\theta = [\theta_1 \quad \theta_2 \quad \cdots \quad \theta_{n-1}]^T \tag{2-13}$$

Paper C focuses on vehicle paths in the global coordinate system, resulting in additional position states. x_P includes the global position of points linked to vehicle units, such as axle centers, wheels, body corners, and the lookahead point for control. x_P also included unit headings. All states in x_P can be expressed using states from Equation 2-12 with the necessary vehicle dimensional parameters. Therefore, they may also be moved into the output.

$$x_P = [X_{ie2} \quad Y_{ie2} \quad X_{i\dots} \quad Y_{i\dots} \quad \Psi_i]^T, i = 1,2,\cdots,n \tag{2-14}$$

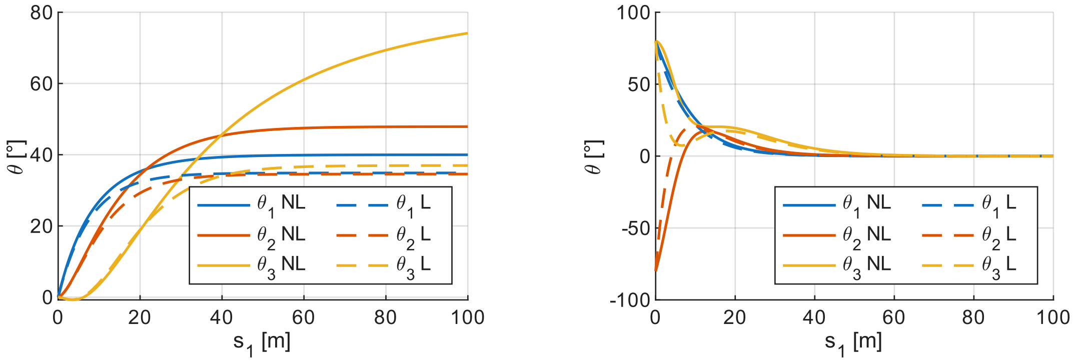
The dynamics of system states given in Equations 2-12 to 14 can be written in terms of time as the independent variable based on Equations 2-1 to 11. Applying the transform given in Equation 2-15, the independent variable of the dynamics can change from time (t) to the vehicle position along the path coordinate of the leading unit's rear axle (s_1). This transform totally removes the leading unit's longitudinal velocity (v_{1te2x}) from expressions of system dynamics because the velocity appears as a scaling factor for all dynamics when using time as the independent variable. The complete removal of velocity means it is possible to develop controllers using a kinematic model without velocity information.

$$\frac{\dot{x}(t)}{v_{1te2x}} = \frac{\mathrm{d}x}{\mathrm{d}t} \div \frac{\mathrm{d}s_1}{\mathrm{d}t} = \frac{\mathrm{d}x}{\mathrm{d}s_1} = \dot{x}(s_1) \tag{2-15}$$

Models built on Equations 2-1 to 11 are nonlinear. Jacobian linearization as shown in Equation 2-16 is commonly applied in existing research for applying linear control design techniques. Most of the linearization is down around the straight pose with zero steering angle and a fixed speed. Some gain-scheduling solutions linearize the model at multiple steady states corresponding to different steering angles. Figure 2-2 shows errors introduced by linearization by comparing articulation angles from a nonlinear model (NL) and its linearized model (L). The latter one is linearized at the straight pose. s_1 is the tractor travel distance. In Figure 2-2.a, the vehicle is driving forward from the straight pose with a fixed steering angle. In Figure 2-2.b, the vehicle drives forward from a non-straight pose with zero steering angle. The errors pose a risk to controller performance when the operating point is away from the linearization point. The vehicle may need to operate in regions far from the chosen linearization point(s) to utilize the flexibility of articulations.

$$\begin{cases} \dot{x} = f(x,u) \\ y = h(x,u) \end{cases} \to \begin{cases} \dot{x} \approx f(x_0,u_0) + \left.\dfrac{\partial f}{\partial x}\right|_{x_0,u_0} \cdot (x - x_0) + \left.\dfrac{\partial f}{\partial u}\right|_{x_0,u_0} \cdot (u - u_0) \\ y \approx h(x_0,u_0) + \left.\dfrac{\partial h}{\partial x}\right|_{x_0,u_0} \cdot (x - x_0) + \left.\dfrac{\partial h}{\partial u}\right|_{x_0,u_0} \cdot (u - u_0) \end{cases} \tag{2-16}$$

Linearization is not the only source of error when applying kinematic models. As mentioned in Paper D, the equivalent axle positions change with different operating conditions, even at low speeds. This means uncertainties in model parameters should also be recognized. Research on off-road vehicles or vehicles operating on non-flat surfaces has observed tire sideslips [57], [99], [100]. One way to address this is to extend the kinematic model to include additional sideslip states for relevant axles. Better path-tracking performance is achieved in existing solutions when accurate estimation of the sideslips is incorporated into the extended kinematic model.



(a) From straight to circular

(b) From non-straight to straight

Figure 2-2 Comparison between nonlinear and linearized kinematic models of an A-double

2.2 Dynamic model

In this study, the term ‘dynamic models’ is associated with force-based vehicle models that mainly concentrate on the dynamics relevant to their specific application field. More detailed and virtual prototype models are referred to as high-fidelity models and are discussed in the following subchapter.

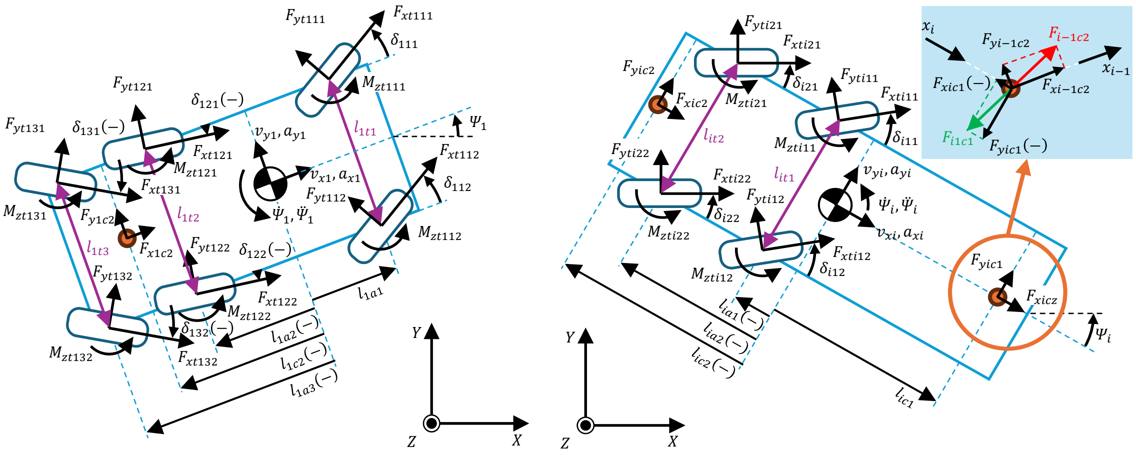
Dynamic models are another type of model found in existing research. In [53], [60], [63], [84], linearized single-track dynamic models are used as control models for control design. In [17], a double-track nonlinear dynamic model is used to validate a kinematic-based control algorithm. The articulation angle limits required for jackknife avoidance with different vehicle-trailer systems are solved with a double-track nonlinear dynamic model in [101]. In [102], a linear single-track dynamic model is used for tire cornering stiffness optimization for a high-fidelity model.

The following dynamic model is a double-track, nonlinear model developed during the preparation of Paper D. It is based on Newtonian mechanics. The dynamic model can also be built through Lagrange mechanics as [103]. Simulations with maneuvers similar to those in Paper D are conducted using this dynamic model and compared with results from the real-vehicle tests and the high-fidelity model in Paper D. The model parameters are not tuned using test measurements from Paper D. The model considers only vehicle motions in the X-Y plane, excluding load transfer. The couplings in the model are assumed to be ideal rotational joints. Notations used in the model are shown in Figure 2-3.

The state vector of the dynamic model of a n-unit articulated vehicle is selected as:

$$x = [\Psi_1 \quad V_{X1} \quad V_{Y1} \quad \dot{\Psi}_1 \quad \theta_1 \quad \cdots \quad \theta_{n-1} \quad \dot{\theta}_1 \quad \cdots \quad \dot{\theta}_{n-1} \quad \delta]^T \tag{2-17}$$

where subscripts denote the unit number, $\dot{}$ denotes time derivative. Ψ, V_X and V_Y are the heading angle, the longitudinal and lateral velocities in the global coordinate system, respectively. δ is the steering angle. All states in x are time dependent. δ can be expanded to a $1 \times n_a$ or a $1 \times 2n_a$ vector to account for individual axle or wheel steering, but here δ is a scalar, as only the front axle on the first unit is considered steerable.



(a) Leading unit

(b) Trailing unit

Figure 2-3 Vehicle units in a double-track dynamic model

The model treats tire forces and moments as external inputs. The steering angle is controlled through the steering angle rate.

$$u = \begin{bmatrix} \boldsymbol{F}_{xt} & \boldsymbol{F}_{yt} & \boldsymbol{M}_{zt} & \dot{\delta} \end{bmatrix}^T \tag{2-18}$$

where $\boldsymbol{F}_{xt}$, $\boldsymbol{F}_{yt}$ and $\boldsymbol{M}_{zt}$ are $1 \times 2n_a$ vectors of tire longitudinal, lateral forces, and aligning moments, n_a is the number of axles in the whole vehicle, $\dot{\delta}$ is the steering rate.

The dynamic of the system, $\dot{x}$, includes several elements that are equal to other system states or inputs. The dynamics of those states can be directly represented by elements of the state or input vectors. The remaining system dynamics that are going to be derived through Newtonian mechanics are given in $\dot{x}_N$:

$$\dot{x}_N = [\dot{V}_{X1} \quad \dot{V}_{Y1} \quad \ddot{\Psi}_1 \quad \ddot{\theta}_1 \quad \cdots \quad \ddot{\theta}_{n-1}]^T \tag{2-19}$$

The primary approach to building the model is to treat each unit individually at first, then combine them into a whole vehicle. For each unit, three equilibria in the longitudinal, lateral, and yaw directions are required for the three corresponding degrees of freedom. Based on Figure 2-3, the three equilibria for the ith unit in its local coordinate system are as follows:

$$\begin{cases} m_i a_{xi} = \sum_{j=1}^{n_{ai}} \sum_{k=1}^{2} \left(F_{xtijk} \cos\delta_{ijk} - F_{ytijk} \sin\delta_{ijk}\right) + F_{xc1i} + F_{xc2i} \\ m_i a_{yi} = \sum_{j=1}^{n_{ai}} \sum_{k=1}^{2} \left(F_{ytijk} \cos\delta_{ijk} + F_{xtijk} \sin\delta_{ijk}\right) + F_{yc1i} + F_{yc2i} \qquad , i = 1, 2, \cdots, n \\ J_i \ddot{\Psi}_i = \sum_{j=1}^{n_{ai}} \sum_{k=1}^{2} \left[F_{xhijk}(-1)^k \frac{l_{itj}}{2} + \left(F_{yhijk}\right) l_{iaj} + M_{ztijk}\right] + F_{yc1i} l_{ic1} + F_{yc2i} l_{ic2} \end{cases} \tag{2-20}$$

$$F_{xhijk} = F_{xtijk} \cos\delta_{ijk} - F_{ytijk} \sin\delta_{ijk}, F_{yhijk} = F_{ytijk} \cos\delta_{ijk} + F_{xtijk} \sin\delta_{ijk}$$

where m_i is the mass of the unit, J_i is the yaw inertial of the unit around its center of gravity. a_{xi}, a_{yi}, and $\ddot{\Psi}_i$ are the longitudinal, lateral, and yaw accelerations, respectively. n_{ai} is the number of axles on the unit. k distinguish the left and right tires on the same axle with values of 1 and 2. F_{xtijk}, F_{ytijk} and M_{ztijk} are the longitudinal, lateral tire forces and the aligning moment at unit i, axle j, and side k, respectively. F_{xc} and F_{yc} are the longitudinal and lateral coupling forces, respectively. $c1$ and $c2$ stand for the front and rear couplings, respectively. l_{itj} is the track width of axle j on unit i. l_{iaj} is the longitudinal displacement from the center of gravity to axle j within unit i. l_{ic1} and l_{ic2} are the longitudinal displacements from the center of gravity to the front and rear couplings within unit i, respectively. Equation 2-20 considered independent steering on all wheels. With the simplified steering setup assumed earlier, δ_{ijk} is assigned as:

$$\delta_{ijk} = \begin{cases} \delta & i = j = 1 \\ 0 & else \end{cases} \tag{2-21}$$

To make Equation 2-20 solvable, unknown variables needed to be expressed with states, state derivatives, and inputs. For the left side of Equation 2-20, this can be done through the kinematics of the vehicle. The yaw acceleration of a trailing unit can be expressed with state derivatives, which include the yaw acceleration of the first unit and articulation angle accelerations, such as:

$$\ddot{\Psi}_i = \ddot{\Psi}_1 - \sum\nolimits_{a=1}^{i-1} \ddot{\theta}_a \, , i = \ 2, \cdots, n \tag{2-22}$$

It is more complex to handle a_{xi} and a_{yi} from Equation 2-20. This is done in two steps. First, the longitudinal and lateral velocities (v_{xi} and v_{yi}) in the unit's local coordinate system are expressed with system states and vehicle dimensional parameters such as:

$$\begin{cases} v_{xi} = f_{v_{xi}}\left(V_{X1}, V_{Y1}, \boldsymbol{\Psi}_i, \dot{\boldsymbol{\Psi}}_{i-1}, \boldsymbol{\theta}_{i-1}, \boldsymbol{l}_{ic1}, \boldsymbol{l}_{i-1c2}\right) \\ v_{yi} = f_{v_{yi}}\left(V_{X1}, V_{Y1}, \boldsymbol{\Psi}_i, \dot{\boldsymbol{\Psi}}_{i-1}, \boldsymbol{\theta}_{i-1}, \boldsymbol{l}_{ic1}, \boldsymbol{l}_{i-1c2}\right) \end{cases}, i = 1, 2, \cdots, n \tag{2-23}$$

where f replaces the nonlinear equations that give two velocities. Bold terms represent a set of variables or parameters. They are empty sets when $i = 1$. $\boldsymbol{\Psi}_i$ contains Ψ_1 to Ψ_i. $\dot{\boldsymbol{\Psi}}_{i-1}$ contains $\dot{\Psi}_1$ to $\dot{\Psi}_{i-1}$. $\boldsymbol{\theta}_{i-1}$ contains θ_1 to $\boldsymbol{\theta}_{i-1}$. $\boldsymbol{l}_{ic1}$ contains the displacements from each unit's center of gravity to its front coupling from the second to the ith unit. $\boldsymbol{l}_{i-1c2}$ contains the displacements from each unit's center of gravity to its rear coupling from the first to the ith unit. All variables used in f are either in or can be expressed with states from Equation 2-17.

The second step is to express a_{xi} and a_{yi} based on Equation 2-23 and Ψ_i. As shown in Equation 2-24, the acceleration includes a time derivative of the local velocity and a product term with the yaw velocity. The time derivative introduces time derivatives of variables from Equation 2-23. All introduced derivatives are within the system states and their directives.

$$\begin{cases} a_{xi} = \dfrac{\mathrm{d}f_{v_{xi}}}{\mathrm{d}t} - v_{yi}\dot{\Psi}_i \\ a_{yi} = \dfrac{\mathrm{d}f_{v_{yi}}}{\mathrm{d}t} + v_{xi}\dot{\Psi}_i \end{cases}, i = 1, 2, \cdots, n \tag{2-24}$$

Equations 2-22 and 24 eliminate unknown variables other than the system states, their derivatives, and the system inputs from the left side of Equation 2-20. Then, move to the right side of Equation 2-20, where the coupling forces (F_{xc1i}, F_{xc2i}, F_{yc1i} and F_{yc2i}) are the unknown variables that need to be expressed with the system states, their derivatives, and the system inputs. This is done by manipulating Equation 2-20 for the first, middle, and last units differently. First, for the first unit that has no front coupling forces, Equation 2-20 can be recognized as:

$$\begin{cases} F_{xc21} = f_{F_{xc21}}(x,\dot{x},u) = m_1 a_{x1} - \sum_{j=1}^{n_{a1}}\sum_{k=1}^{2}\left(F_{xt1jk}\cos\delta_{1jk} - F_{yt1jk}\sin\delta_{1jk}\right) \\ F_{yc21} = f_{F_{yc21}}(x,\dot{x},u) = m_1 a_{y1} - \sum_{j=1}^{n_{a1}}\sum_{k=1}^{2}\left(F_{yt1jk}\cos\delta_{1jk} + F_{xt1jk}\sin\delta_{1jk}\right) \\ f_1(x,\dot{x},u)\colon 0 = J_1\ddot{\Psi}_1 - \sum_{j=1}^{n_{a1}}\sum_{k=1}^{2}\left[F_{xh1jk}(-1)^k\frac{l_{1tj}}{2} + \left(F_{yh1jk}\right)l_{1aj} + M_{zt1jk}\right] - f_{F_{yc21}}(x,\dot{x},u)l_{1c2} \end{cases} \quad (2\text{-}25)$$

where F_{xc21} and F_{yc21} are now expressed as functions of the system states, their derivatives, and the system inputs. Plug $f_{F_{yc21}}$ into the yaw equilibrium result in the first equation, which will be used to solve the system dynamics. The dependence on vehicle parameters is not explicitly stated for the sake of simplification.

Moving to the middle units that have both front and rear coupling forces. There is a coupling force equilibrium at each shared coupling point, that is, the front coupling of one unit and the rear coupling of the unit in front of it. The equilibrium is:

$$\begin{cases} F_{xc1i} = f_{F_{xc1i}}(F_{xc2i-1},\theta_{i-1}\ ) = -\left(F_{xc2i-1}\cos\theta_{i-1} - F_{yc2i-1}\sin\theta_{i-1}\right) \\ F_{yc1i} = f_{F_{yc1i}}\left(F_{yc2i-1},\theta_{i-1}\right) = -\left(F_{xc2i-1}\sin\theta_{i-1} + F_{yc2i-1}\cos\theta_{i-1}\right) \end{cases}, i = 2,3,\cdots,n \quad (2\text{-}26)$$

Reorganize Equation 2-20 with Equation 2-26:

$$\begin{cases} F_{xc2i} = m_i a_{xi} - \sum_{j=1}^{n_{ai}}\sum_{k=1}^{2}\left(F_{xtijk}\cos\delta_{ijk} - F_{ytijk}\sin\delta_{ijk}\right) - f_{F_{xc1i}}(F_{xc2i-1},\theta_{i-1}\ ) \\ F_{yc2i} = m_i a_{yi} - \sum_{j=1}^{n_{ai}}\sum_{k=1}^{2}\left(F_{ytijk}\cos\delta_{ijk} + F_{xtijk}\sin\delta_{ijk}\right) - f_{F_{yc1i}}\left(F_{yc2i-1},\theta_{i-1}\right) \\ f_i(\cdot)\colon 0 = J_i\ddot{\Psi}_i - \sum_{j=1}^{n_{ai}}\sum_{k=1}^{2}\left[F_{xhijk}(-1)^k\frac{l_{itj}}{2} + \left(F_{yhijk}\right)l_{iaj} + M_{ztijk}\right] - f_{F_{yc1i}}\left(F_{yc2i-1},\theta_{i-1}\right)l_{ic1} - F_{yc2i}l_{ic2} \end{cases} \quad (2\text{-}27)$$

$i = 2,3,\cdots,n-1$

Equation 2-27 couples to Equation 2-25 when $i = 2$, which results in F_{xc22}, F_{yc22} and $f_2(\cdot)$ are purely dependent on $(x,\dot{x},u)$. Reuse Equation 2-27 with i increased step by step from 3 to $n-1$, then all the yaw equilibria of the middle units can be recognized as $f_i(x,\dot{x},u)$. For a total $n-2$ units, there are $n-2$ equations ($f_i(x,\dot{x},u), i = 2,3,\cdots,n-1$) will be used in solving the system dynamics.

The last set of equations needed for solving the system dynamics comes from the last unit, which is not exposed to forces from its rear coupling. Therefore, Equation 2-27 is modified to:

$$\begin{cases} f_{n+1}(x,\dot{x},u)\colon 0 = m_n a_{xn} - \sum_{j=1}^{n_{an}}\sum_{k=1}^{2}\left(F_{xtnjk}\cos\delta_{njk} - F_{ytnjk}\sin\delta_{njk}\right) - f_{F_{xc1n}}(F_{xc2n-1},\theta_{n-1}\ ) \\ f_{n+2}(x,\dot{x},u)\colon 0 = m_n a_{yn} - \sum_{j=1}^{n_{an}}\sum_{k=1}^{2}\left(F_{ytnjk}\cos\delta_{njk} + F_{xtnjk}\sin\delta_{njk}\right) - f_{F_{yc1n}}\left(F_{yc2n-1},\theta_{n-1}\right) \\ f_n(x,\dot{x},u)\colon 0 = J_n\ddot{\Psi}_n - \sum_{j=1}^{n_{an}}\sum_{k=1}^{2}\left[F_{xhnjk}(-1)^k\frac{l_{ntj}}{2} + \left(F_{yhnjk}\right)l_{naj} + M_{ztnjk}\right] - f_{F_{yc1n}}\left(F_{yc2n-1},\theta_{n-1}\right)l_{nc1} \end{cases} \quad (2\text{-}28)$$

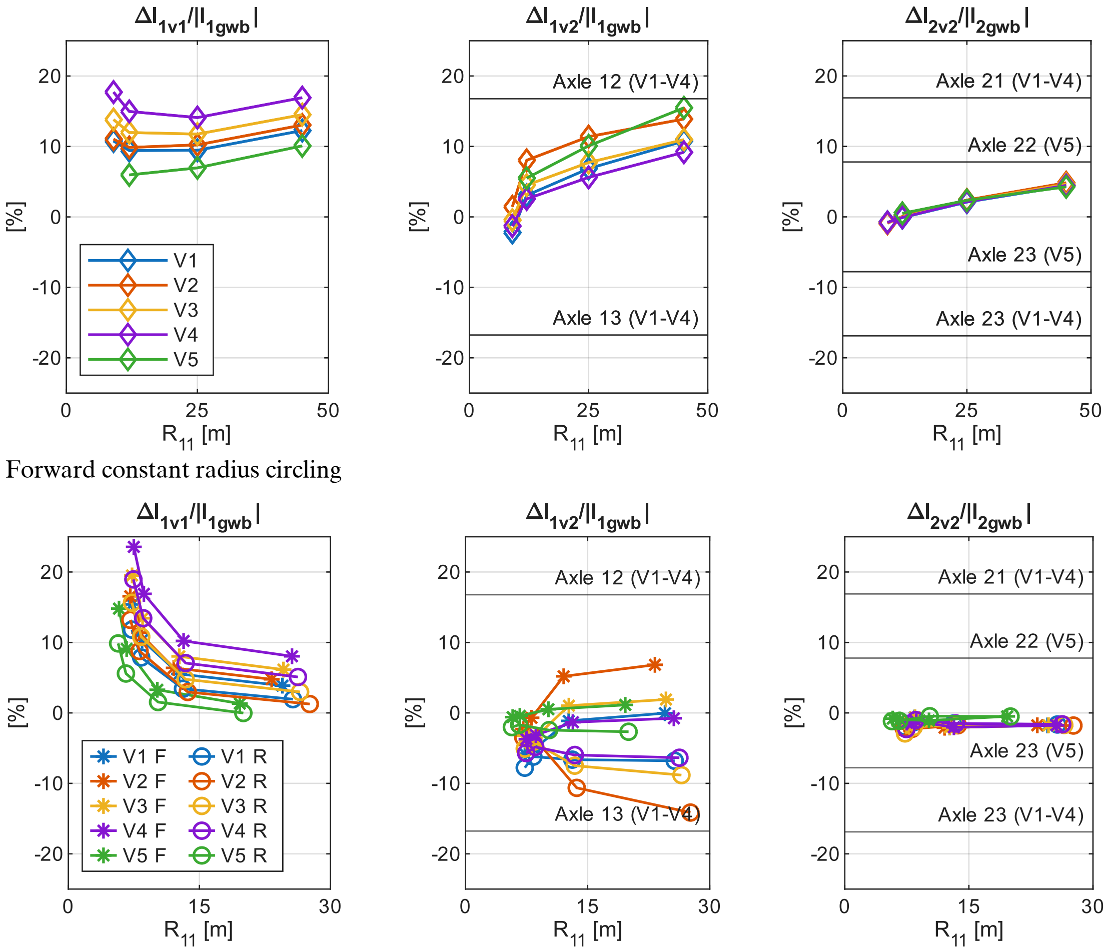
Up to this point, Equations 2-25 to 28 have provided the $n+2$ equations ($f_1(x,\dot{x},u)$ to $f_{n+2}(x,\dot{x},u)$) that are needed for solving the $n+2$ elements in $\dot{x}_N$. Include the $n+1$ integral states in x, there are $2n+3$ equations for $2n+3$ states in x. The $2n+3$ equations create a set of nonlinear implicit ordinary differential equations (ODEs). The system dynamics are solved with ode15i [104] in MATLAB. Linearization is a way to get an explicit solution to the system, but it is not used here because the goal is to achieve more accurate simulations.

The wheel hub velocities are computed from system states and input into an external tire model. This tire model is considered external because it is not solved simultaneously with the system dynamics. The used tire model is based on the Magic Formula model [105], and its parameters are extracted from Volvo Transport Models (VTM) [106]. The parallel Magic Formula model from [107] is included for the turn slip. The dynamic model is built in Simulink, where the tire model uses the solution of the ODEs, and the forces and moments generated by the tire model are fed back into the ODEs. An algebraic loop is created in the model, and a delay block between the solution of the ODEs and the tire model solves it. This is a source of error for simulations. It is theoretically possible to expand the ODEs to include the tire model; however, in practice, formulating and solving them within a reasonable time can be challenging.

Compared to the kinematic model, the presented dynamic model should have the additional ability to represent the vehicle dynamics under conditions that can create or change tire sideslip angles. To verify this ability, simulations similar to those in Paper D conducted with VTM are also performed using this dynamic model. As mentioned in Paper D, some vehicle parameters, such as mass and yaw inertia, used in the model are based on rough estimates of the test vehicle. Due to differences in model setups, the kinematic model and VTM also don't share the same parameters. Therefore, any comparison between models or between a model and the real vehicle is at the quantitative level. The results presented in Figure 2-4 are based on the same maneuvers and the same method presented in Paper D. Recap from Paper D, the normalized virtual axle offsets (Δ/l) indicate the normalized displacement between the virtual axle position, which has zero sideslip, and its corresponding physical axle (geometrical center for a bogie) position. A higher magnitude in Δ/l means a higher error in the kinematic vehicle parameter, if the physical axle positions or the bogie centers are selected as the positions for equivalent axles.

Figure 2-4.a shows results from steady-state circling maneuvers, which are comparable to those in Fig. 5 of Paper D. Some trends observed in the real vehicle tests are also present in the dynamic model simulations. For vehicles tracking the same radius, the orders of the offsets between different vehicle setups are the same or similar. The offset trends when the tracking radius changes differ between the real-vehicle test and the two models, namely the dynamic model and the VTM. Meanwhile, simulations from the dynamic model and VTM show similarities. Figure 2-4.b is comparable to Fig. 11 in Paper D. Since the dynamic model is symmetric, only left turn simulations are performed. Both the real vehicle tests and the dynamic model simulations show that when the change direction changes from forward to reverse, the tractor's axle offsets ($\Delta_{1v1}/|l_{1gwb}|$ and $\Delta_{1v2}/|l_{1gwb}|$) are moving towards the negative direction. However, as with the constant-radius tests, the trend in offsets changes when the path radius changes are not uniform between tests and simulations. One possible error source is that the non-stable tractor velocity in the test is not followed in the simulations. Due to the simplifications made in the presented dynamic model, it cannot handle load transfer.

In summary, a dynamic model can capture motion behaviors that the kinematic model cannot. The complexity of the dynamic model can vary depending on the level of detail included. When using a dynamic model as the control model, it is essential to balance its accuracy with its complexity. Additionally, the accessibility and uncertainty of the parameters required by the dynamic model should be considered when evaluating the model.



(a) Forward constant radius circling

(b) Forward (F) and reverse (R) constant steer

Figure 2-4 Virtual axle offsets in dynamic model simulations

2.3 High-fidelity model

Two high-fidelity models used in the present thesis are the Volvo Transport Model (VTM) [106] and TruckMaker [108]. Both models are built on multi-body dynamics (MBD) and include significantly more degrees of freedom than the dynamic models discussed in the previous subchapter, enabling them to capture more detailed dynamics in simulations. Suspended cab, vehicle bodies, suspended axles, and tires are included as basic subsystems. Additional subsystems can be added. In addition to the basic subsystems included in VTM, TruckMaker also includes powertrain and brake systems. TruckMaker supports co-simulation with Simulink. The compatibility of both VTM and TruckMaker with Simulink is beneficial in control design. It provides a quick way to implement and validate novel control algorithms in a relatively realistic virtual environment. Consequently, these high-fidelity models serve as an effective bridge between theoretical control design and practical vehicle implementation, enabling robust performance evaluation before experimental testing on physical prototypes.

2.4 Summary and discussion

Among the three types of models introduced in this chapter, the kinematic model is the simplest both in terms of its required parameters and the equations describing its dynamics. The dimensional parameters of vehicles can be easily obtained from existing databases or measured on-site. The explicit solutions of dynamics enable computationally efficient simulation and prediction using explicit methods, which can significantly benefit path-planning algorithms that involve random sampling and simulation. It is expected that the simple kinematic model will not be able to capture some detailed behaviors observed in real vehicles and in high-fidelity models, as shown in Papers B and D. Some existing research extends kinematic models with additional states to account for lateral slip at axles, which gives chances to increase the accuracy of kinematic models in more scenarios.

The complexity of dynamic models can vary widely depending on how detailed the vehicle and its subsystems are modeled. Most of the existing research does not use dynamic models. A few existing studies employ a highly simplified dynamic model with combined axles, linear tire models, and the entire model is linearized. This is a compromise between the model details and the control approaches, including the application of linear control techniques and real-time performance of model predictive control. It is worth further investigation to determine how the performance of model-based controllers differs between using a kinematic model and a dynamic model with acceptable complexity across a wider operating range. The difficulties in obtaining accurate parameters for dynamic models should not be forgotten, as this can be an obstacle to their performance in practice.

High-fidelity, high-detail dynamic models can still be used in control design, but they are unlikely to be applied in real time. Precalculated results, such as swept areas and slip angle ranges, can be used in control algorithms. High-fidelity models are also an essential means of functional validation. To be a reliable validation environment, the high-fidelity model is expected to capture as many behaviors as observed in tests for Paper D and other tests. Even though no high-fidelity model is perfect, and real-vehicle tests will still be more convincing in many ways, validation with high-fidelity models will be a key part of the current development stage, as simulations are easier than real-vehicle experiments for exploring a wide range of scenarios.

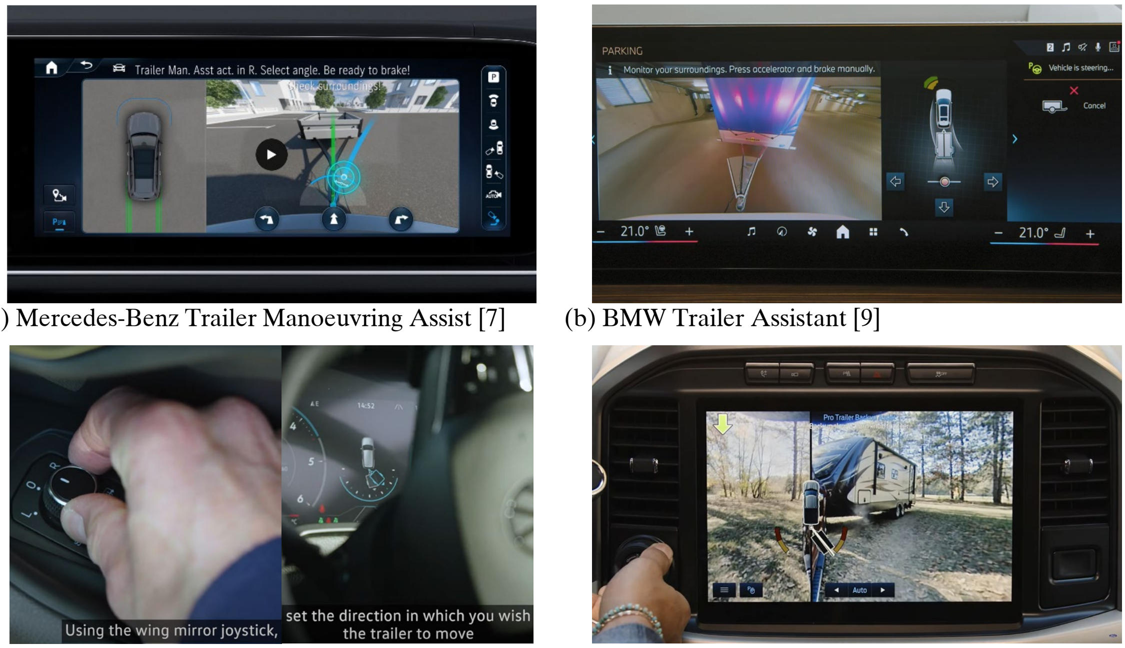
Chapter 3
Reversing control

Research on the control of reversing articulated vehicles can be categorized into three main categories: articulation stabilization, path following, and path planning. Articulation stabilization primarily functions as a driver-assistance feature, since the driver must supply a reference articulation angle or path radius. It is sometimes used as a fallback or auxiliary controller for path-following controllers to prevent the articulation angle from exceeding a limit that would cause a jackknife. Path-following aims to help vehicles follow given reference paths, either by autonomously controlling steering or by providing steering guidance through a human-machine interface. Path-planning provides feasible reference paths for path-following functions. The external environment, the vehicle, and the path-following solutions jointly determine the paths' feasibility. This chapter introduces existing research in each category and the connections between categories. There is no direct comparison in control performance because it is difficult to fairly compare different controllers with tunable parameters without a unified tuning method [77]. The discussion instead focuses on methodologies.

3.1 Articulation stabilization

Reversing an articulated vehicle involves controlling an unstable system. Continuous steering corrections are necessary even when the vehicle needs to maintain a steady state while reversing in a straight line or along a circular path with a constant radius. The articulation stabilization aims to automate steering corrections based on a driver's input. Usually, the driver's input is the target articulation angle for single-articulated vehicles, or the last trailer's turning radius for multi-articulated vehicles. The controllers need to be designed to reject infeasible targets, such as articulation angles exceeding certain limits and reference radii that are too small. Besides the difference in the driver's input between single- and multi-articulated vehicles, the target for the articulation stabilization is the same, that is, to stabilize vehicles to the steady state determined by the driver's input. Under the assumption of single steered front axle group on the leading unit, there is a difference in the control-based articulation stabilization between single- and multi-articulated vehicles. For a single articulated vehicle, the controlled system is SISO (Single-Input Single-Output), with the input being the steering and the output being the single articulation angle. On the other hand, a multi-articulated vehicle is a SIMO (Single-Input Multi-Output) system, as multiple articulation angles must be controlled through a single steerable axle group.

The first step is to investigate existing articulation stabilization from the four industrialized driver aids shown in Figure 3-1. A common function for them is to maintain the articulation angle according to the driver's input. A shortcut is also available in all four aids to quickly set the target articulation angle to zero. The human-machine interface is different across the driver aids: the target articulation angle is specified via the touchscreen in [7], [9], and via a physical knob or joystick in [9], [10], [11]. Two more advanced functions are available in [7]. It can turn the trailer 90 degrees to the left or right with respect to its initial heading and reach zero articulation angle at the end of the maneuver. Additionally, it can control the vehicle to maintain zero articulation angle while keeping the trailer's heading. The two advanced functions involve controlling the trailer's heading in the global coordinate system, which is more than just controlling the articulation angle. However, [7] is not considered a path-following solution because the path used for turning the trailer 90 degrees is not presented to the driver.


Trailer Man. Asst act. in R. Select angle. Be ready to brake!

Check surroundings!

PARKING

Monitor your surroundings. Press accelerator and brake manually.

Vehicle is steering...

Cancel

Using the wing mirror joystick, set the direction in which you wish the trailer to move

(a) Mercedes-Benz Trailer Manoeuvring Assist [7]

(b) BMW Trailer Assistant [9]

(c) Volkswagen Trailer Assist [10]

(d) Ford Pro Trailer Backup Assist [11]

Figure 3-1 Articulation stabilization solutions in passenger vehicles

None of the industrialized functions discussed here provides drivers with predictive information about the trailer's path. The relationship between a specific articulation angle and the vehicle's steady-state path is not trivial to all drivers. What's maybe even more confusing to drivers is that the path taken by the vehicle from the current position to the steady state corresponds to the targeted articulation angle. As shown by the kinematic model in chapter 2.1, the articulation angle rate is proportional to the velocity of the leading vehicle, which means changing the articulation angle requires travel distance. This means the driver's desired state always comes with a delay. In general, the driver still needs to be familiar with the dynamics, including the assistance system, to perform precise parking with a trailer.

In Table 1-1, most of the research on articulation stabilization for single-articulated vehicles may be considered as variants of the form given by Equation 3-1. The steering angle (δ) in Equation 3-1 consists of a feedforward part based on the reference articulation angle ($\delta_{ff}(\theta_r)$) and a feedback part based on the articulation angle error ($\delta_{fb}(\theta - \theta_r)$).

$$\delta = \delta_{ff}(\theta_r) + \delta_{fb}(\theta - \theta_r) \tag{3-1}$$

where δ is the steering angle, δ_{ff} is the feedforward steering angle based on the reference articulation angle θ_r, δ_{fb} is the feedback steering angle based on the articulation angle error $\theta - \theta_r$, the mapping for both δ_{ff} and δ_{fb} can be dynamic.

The feedforward part ensures the steering angle closely matches the necessary angle to maintain the unstable steady state associated with the articulation angle, even when the articulation angle error is near zero. The feedforward steering angle can be determined using a kinematic model. The reference input is not limited to the articulation angle alone but also includes other quantities that may be transferred to the articulation angle. These may include the trailer's turning radius or a virtual steering angle on its virtual axle. The feedback part can be implemented using various control approaches. A straightforward approach is a manually tuned proportional control that uses the articulation angle error as the processing error and directly controls the steering angle. Advanced control theories, such as linear–quadratic regulator (LQR) and model predictive control (MPC), can also be utilized in the feedback part. The feedforward and feedback parts can also be designed jointly. One approach is to link the articulation angle error to a desired articulation angle rate, then calculate the steering angle using a kinematic model.

The articulation stabilization is treated differently in [12], where no controller is designed for steering. The system in [12] aims to help the driver maintain or change between steady states while reversing. Two predicted paths at the trailer's middle axle, based on kinematics, are displayed to the driver on a screen. One path is based on the actual articulation angle with an assumed steering angle that results in the vehicle being in a steady state; the other path is based on the actual articulation angle and the actual steering angle. If the driver matches the two predicted paths by changing the steering angle, then the vehicle can maintain a steady state corresponding to the current articulation angle. By diverging the two predicted paths, the relative positions of the paths are used to guide the driver in changing the steady-state curvature along the trailer's path.

Not all research treats the articulation stabilization problem as an articulation angle tracking problem. The articulation stabilization part in [22] acts as a fallback controller for a path-following controller. It overwrites the path-following controller with a steering command that controls the vehicle towards the straight pose when there is a risk of a jackknife. A path-following controller designed for a non-articulated vehicle is combined with an articulation stabilization control in [23]. The stabilization part continuously adjusts the steering request from the path-following part to avoid a jackknife.

For research papers on articulation stabilization that are listed in Table 1-1, more research is conducted on single-articulated vehicles than on multi-articulated vehicles. The categorization of suitable vehicles is mainly based on the information provided in the publications, which means it may be under- or over-bounded. A controller is called underbounded if it claims to be suitable for vehicles with more articulation joints but does not achieve similar performance, or is practically difficult to extend. For example, some research first defines a target motion for the last trailer and then calculates the required motion for the first unit using the kinematic model. Those approaches are valid with tests in publications, but are often not used with trucks or similar vehicles. A required motion that is possible for a differential-wheeled robot can be impossible for a tractor, as it can require a steering angle that exceeds the mechanical limit of the tractor, which is not only due to the difference in the leading unit but also the dimensions of all trailing units. Paper C analyzed the dynamics of several typical LCVs while reversing and demonstrated why the inverse kinematic method is limited for these LCVs. The dynamics of added articulations are coupled with those of existing articulations, meaning that for single-axle steered articulated vehicles, controlling the added articulation angles requires manipulating the dynamics of the existing articulation angles. This means that a controller validated for certain articulated vehicles may have a reduced feasible operating range when more trailers are added. This effect may not seem trivial because, in the kinematic model, additional articulations merely involve appending a set of equations and do not change the dynamics of existing articulation angles. A controller is called overbounded, usually because its feasibility on vehicles with few articulations is not explicitly stated in the publication, which needs to be reminded when using Table 1-1. It is essential to understand the fundamental limitations of articulated vehicles to evaluate the performance of an articulation stabilization and determine whether it is underbounded. The fundamental limitations are studied through Papers A and B, which aim to define the boundaries of the articulation angles within which vehicles can be reversed into a straight pose, considering both steering angle and inter-unit clash constraints. An ideal articulation stabilization controller should be able to stabilize a vehicle from any pose within the vehicle's fundamental limitations.

At the conceptual level, articulation stabilization for a multi-articulated vehicle is similar to that for a single-articulated vehicle, as the controller has a similar structure. The SIMO controller structure in Equation 3-2 is an expansion of the SISO version in Equation 3-1.

$$\delta = \delta_{ff}(\theta_{n-1r}) + \delta_{fb}(\boldsymbol{\theta} - \boldsymbol{\theta}_r) \tag{3-2}$$

where θ_{n-1r} is the reference articulation angle for the last articulation angle of the n-unit articulated vehicle, $\boldsymbol{\theta}$ is the measured articulation angle vector with size $n-1$, $\boldsymbol{\theta}_r$ is the θ_{n-1r} dependent reference articulation angle vector.

For practice reasons, the reference for a multi-articulated vehicle is usually given as the turning radius of the last trailing unit and then transferred to θ_{n-1r} and $\boldsymbol{\theta}_r$ through a kinematic model. The multiple articulations do not make calculating the feedforward part for steady state tracking more involved. The feedback component usually becomes more difficult to tune depending on how it is designed. Some papers implemented the feedback part in a chain-like manner, such as placing a controller at each articulation to control the motion of the unit close to the leading unit based on the desired motion of the unit close to the trailing unit. This can be done either by a series of proportional controllers or with the backstepping technique [13]. Finally, the desired motion of the leading unit is realized by controlling its steering. Kinematics are typically used to define the relative motion between neighboring units. Each controller introduces at least one independent control parameter. Increased tunable parameters and articulations make the tuning challenge more complex. A workaround seen in existing research for this is to use control approaches that treat all articulations in a coordinated way, such as full-state feedback (FSF), LQR, MPC, etc. There are still parameters that need to be tuned, but they might be easier to interpret. The vehicle's nonlinearity should also be considered when using linear control approaches.

Results shown in Figure 3-2 are generated using an LQR-based articulation stabilization controller to show a drawback that needs to be overcome by drivers when using articulation stabilization as an assistant function. The controller is designed in the same way as the one used in Paper B. System matrices are adjusted for the vehicle, and the state cost matrix is maintained as an identity matrix of the correct dimension, yielding acceptable performance. The drawback shown in Figure 3-2 is the transition distance required for articulated vehicles to reach the desired steady state, which increases as the vehicle becomes longer and has more articulations. The initial state of each vehicle is shown in the bottom-right corner of each subfigure, with a straight pose and the last trailer axle at the origin. The blue initial reference represents a path with a 5-meter radius that the driver would like to follow by the last unit. The paths of different units show how the vehicle moves if the 5-meter radius is used as the reference for the articulation stabilization controller. Obviously, when vehicles reach the desired steady state, the centers of the actual paths at the last units differ from the initial reference. This means that if drivers want to control the precise position of the vehicle, they need to be familiar with this kind of transient dynamics and actively and predictively adjust the control reference. Further assistance in this leads to path following solutions.

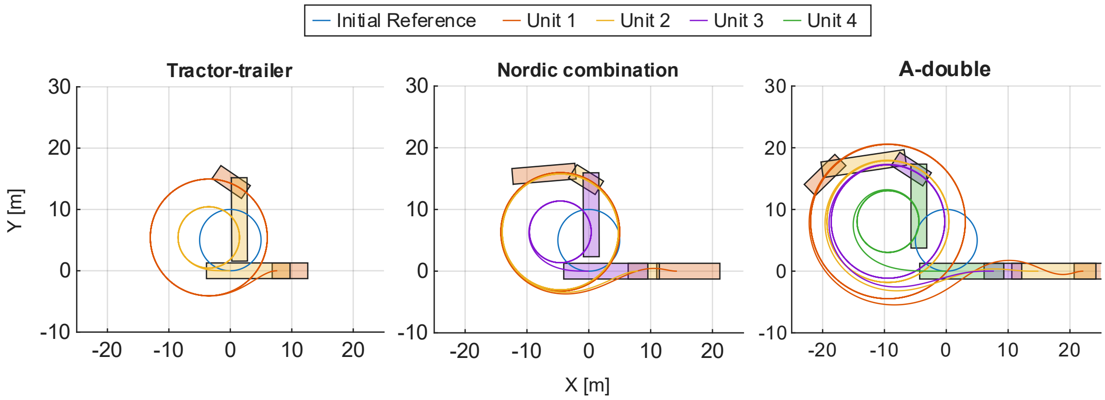


Figure 3-2 Reverse articulation vehicles into a steady state

3.2 Path following

Path-following controls the steering or provides steering guidance to the driver for articulated vehicles following predetermined paths. The high number of studies in Table 1-1 indicate this is an attractive research area. Single articulated vehicles are the most common research objects in references, potentially because they are common across many vehicle types, including heavy, passenger, and agricultural vehicles. The number of studies focusing on multi-articulated vehicles is lower than that on single-articulated vehicles. It is more common for studies on multi-articulated vehicles to originate from robotics. This chapter aims to provide a high-level overview of existing path-following research in three key aspects: objectives, control structures, and controller types. The following discussion is based on path following for the reversing vehicle case. With slight modifications, the three key aspects are also applicable to controlling an articulated vehicle in another unstable mode: path-following with the trailing unit, usually the last one, while driving forward. That is the operation mode needed when the path planning uses cusp technology, which will be introduced in the following subchapter.

3.2.1 Objectives

The expression of controlling a vehicle to follow a predetermined path is blurry because it does not specify the vehicle state vector and the corresponding reference vector needed for control design. An exact definition of the path-following objectives requires definitions of two types of points: control points and reference points.

The control points are points fixed in the vehicle units' local coordinate systems. Some common choices of control points are shown in Figure 3-3, C_{nl} is a lookahead point fixed on the last trailing unit's coordinates that moved away from the axle center in the reverse direction, with a lookahead distance l_{la} ([84]); C_n is a point on the last trailing unit ([33], [35], [39], …); C_1 is a point on the leading unit ([60], [63]). Besides being located at the bogie centers as shown in Figure 3-3, C_1 and C_n can also be located at other places on the vehicle units. Most commonly, only one control point is used in path following, but controllers can also use more control points simultaneously ([60], [63]). The states linked to the control points are positions and units' headings in the global coordinate systems. Those states are part of the states in Equation 2-12, or are linear or nonlinear combinations of those states. Even if dynamic models with more states are used in some studies ([53], [63], [84], …), the states linked to Equation 2-12 are still used for control, because the positions and headings of all units in articulated vehicles can be derived from them.

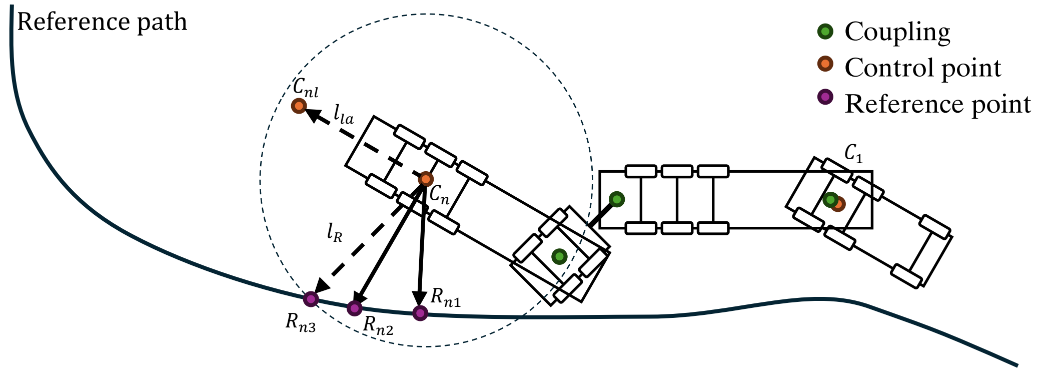


Figure 3-3 Control and reference points in path following

The reference points are points on the predefined reference path. The objective of the controller is to control the vehicle so that the states at the control points converge to the corresponding states at the reference points. It is straightforward when the controlled states are positions, as they can be directly extracted from a point on the reference path ([64], [65], [67], …). The reference heading is typically defined as the tangential direction of the reference path at the reference point ([49], [65], [72], …), which requires the path to have differentiable curvature. However, the actual path of control points on articulated vehicles can lack differentiable curvature, as discussed in detail in the next chapter. Sometimes, the curvature itself is also used as a control state from reference points ([40], [48], [57], …). Figure 3-3 shows three common methods of assigning reference points that move with control points, where control point C_n is selected as an example. For many methods, the reference points are moving with the control point. First, R_{n1} is selected as the point on the reference path that is closest to the corresponding control point C_n ([63], [65], [71], …). Second, R_{n2} is selected as the point on the reference path that is on the unit's lateral direction of C_n ([41], [48], [56], …). Third, R_{n3} is selected at a point that has a certain distance (lookahead distance l_R) from the reference path to C_n ([39], [70], [84], …). There can be more than one solution for R_{n3}, then, additional algorithms are needed to determine which solution to use. As discussed earlier, sometimes the lookahead distance may already be implied at the control point. For those setups, it is reasonable to select a nearby point on the reference path. The reference point can also be provided externally, without an explicit relation to the control point ([49], [50]).

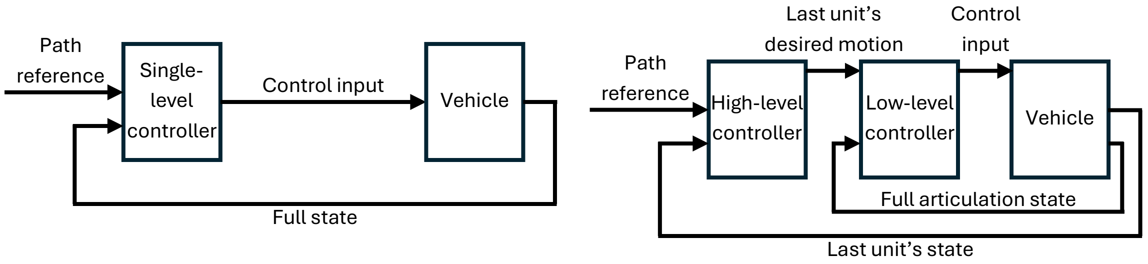
The general objective of the path-following controller is to control the vehicle so that the control points have the same states as the corresponding reference points, which means the control errors are the state differences between the control and reference points. Reducing the lateral error between the control and reference points is typically considered the most crucial target when reversing articulated vehicles. Depending on the choice of the control point and control methods, the lateral velocity of the control point may need to be described in different coordinate systems. For example, if an LQR is applied, when the control point is on the last trailer's axle, its lateral motion should not be described in the trailer's local coordinate system, because the local lateral velocity in the kinematic model is always zero. Instead, the lateral velocity should be described in the path-dependent Frenet frame with the longitudinal velocity in the unit's local coordinate system and heading angle in the Frenet frame. The Frenet frame is a right-handed orthonormal frame whose origin is attached to the reference point that moves along the reference path and whose longitudinal direction is tangential to the path. On the other hand, when the control point is away from the axle, the lateral velocity can be described in both the unit's local coordinate system and the Frenet frame. Some controllers are designed only to control the lateral tracking error directly, such as [53] only uses non-zero weights in the cost function on the lateral error and steering angle. Only controlling the lateral error can also achieve a good heading tracking [84]. The heading can also be tracked actively [65]. The longitudinal error in the last trailer's coordinate system is often ignored in most studies, as tracking position along the path is less important than maintaining lateral stability in the reversing task ([27], [63], [68], …).

The objectives mentioned are also applicable to non-articulated vehicles. What makes articulated vehicles special is their articulation angles, which become unstable during reverse. It is necessary to control them to avoid inter-unit clashes or to have the desired final pose required by the parking place. The most common choice of reference for articulation angles is the steady-state articulation angle based on the desired last trailer's instantaneous motion ([65], [67], [68], …), which is the same as that commonly used for articulation angle stabilization. The desired motion depends on how path-following controllers are designed. The articulation angles are not always actively controlled while path following. However, they can still converge to steady states if the path-following controller can follow a path with constant curvature without oscillations. In [84], the cost function for the LQR controller doesn't include articulation angles; however, by injecting the desired steady-state articulation angle into the feedback states, articulation angles can converge to their steady states. In [40], the motion of the leading unit is always calculated from the desired motion of the last unit with inverse kinematics, and the vehicle can follow constant curvature paths in steady states without using reference articulation angles.

Path-following controllers reach the objectives by giving the target steering angle [27] or steering angle rate [68] to the steering system. As mentioned before, the longitudinal velocity of the last trailer is usually not considered in the path-following control. However, in studies where the trailer velocity is also controlled, such as [57], a target velocity must be provided to the leading unit.

3.2.2 Control structures

Two common control structures used in path-following controllers for reversing articulated vehicles are shown in Figure 3-4. The single-level controllers ([27], [56], [67], …) treat the path-following problem as a whole and calculate the control input by simultaneously considering all included control objectives. Naturally, such single-level controllers are generally linked to optimal control theory. The two-level controllers ([40], [41], [65], …) consist of a higher-level control that handles path-following for the last trailing unit and a lower-level control that provides control inputs based on the desired motion from the higher-level control. Path-following controllers designed for non-articulated vehicles are common choices for the higher-level control ([39], [44], [65], …). The higher-level control's output is the desired motion of the last trailing unit, typically the desired curvature, which serves as the reference for the lower-level control. The articulation stabilization solutions mentioned earlier can use this reference and serve as the lower-level control. In other words, the lower-level controller in a two-level path-following can be used independently as an articulation stabilization controller.



(a) Single-level controller

(b) Two-level controller

Figure 3-4 Common control structures in path-following controllers for reversing articulated vehicles

The control methods used in the two control structures are briefly introduced later and divided into standard and specialized control methods. Standard methods are either similar methods used across studies or methods that can be adopted straightforwardly for different vehicles. Specialized methods rely heavily on understanding system dynamics, which makes them more challenging to adopt across different vehicles than standard methods.

3.2.3 Standard control methods

Control techniques used in standard single-level controllers include linear–quadratic regulator, model predictive control, full state feedback, feedback linearization, and reinforcement learning. For standard two-level controllers, the high-level controllers are divided into three categories based on their connections to the geometric properties of the path-following problem; the low-level controllers can use control techniques used for single-level controllers or the inverse kinematics. In the following part, the control techniques used in single-level controllers are introduced first. Their usages in two-level controllers are introduced simultaneously. The second part will focus on control methods used only as the low-level part of two-level controllers. The high-level control techniques used in two-level controllers are discussed at the end.

The following control methods are primarily used as single-level controllers, though some are also used in two-level controllers.

LQR (linear–quadratic regulator). Already in 1986, the problem of reversing lane change for multi-articulated vehicles was formulated as a linear optimization problem in [27], where the vehicle states include the tractor heading and lateral position in the global coordinate system, all articulation angles, and the input is the steering angle. All states and the input are included in the cost function, with weights tuned to yield a response similar to that of human drivers. Some basic LQRs ([27], [28], [69], [78], [83]) were designed without feedforward steering angle, and nonzero articulation angle references aim for a straight pose for the vehicle when the lateral error and, sometimes, the heading error are reduced to zero. These basic LQRs can provide good performance when the vehicle does not need to be stabilized around a non-straight steady state, for example, when the reference path is a straight line or the lateral error changes more quickly than the vehicle. Improvements are achieved by introducing a feedforward steering angle, with corresponding steady-state articulation angles used as references ([53], [61], [80], [84], [86]). Therefore, the vehicle can reach a steady circling state corresponding to the feedforward steering angle when the LQR successfully brings all states to their references. The feedforward part introduces a parallel level in a single-level control. Still, it is not considered a two-level control because the two levels are defined to be series-connected. The LQRs are still handling the path-following and the articulation stabilization simultaneously. The feedforward inputs and reference states can be hard to calculate for a multi-actuator vehicle, since a steady state can be reached by different input combinations. The problem is solved in [80] by introducing simplified ghost vehicles to facilitate reference calculation.

A particularly interesting LQR and feedforward formulation is observed in [53], where the LQR cost function only consists of the weighted lateral error from the last unit and the weighted steering angle. Meanwhile, articulation stabilization is achieved indirectly, as the input will reach zero if all modelled states, including the articulation angles, are driven to their references. In [60], lateral offsets of the last and the first units are included in the LQR cost function to formulate a minimum swept path control.

LQR is also used as the low-level control in the two-level control ([65], [67], [68], [81], [82]), which can be seen as an independent articulation stabilization control. Some techniques are used in existing studies to mitigate the negative effects of linearization. The articulation angle weight in [67] depends on the lateral error: lower weights are used for higher errors to improve stability, and higher weights are used for lower errors to accelerate convergence. The LQR gain is then calculated based on the latest linearized model. Similarly, gain-scheduled LQRs are developed in [81], [82], where the steady state, corresponding to the desired motion from the high-level control, is the operating point for the linearization and subsequently affects the LQR gain calculated with the same cost function.

MPC (model predictive control). One major difference between LQR and MPC is the ability to handle constraints. Therefore, problem formulations used for LQR design can easily move to MPC as [56] and add additional constraints on states and inputs. As extensions of the LQR-based minimum swept path control in [60], two MPC-based lane-bounded controllers are presented in [63], one with fixed weights and one with adaptive weights on lateral offsets. These two MPC controllers are not considered path planning because the path boundaries still guide the vehicle. In [43], the MPC uses a tractor-trailer modeled by a mixed logical dynamical model to capture the nonlinearity in a linear-like way.

FSF (full state feedback). Instead of using optimal theory in LQR, FSF stabilizes the closed-loop system using a linearized vehicle model with assigned eigenvalues. FSF is combined with a feedforward part in [67] to be a single-level control. The closed-loop system's stability is only local due to the linearization of the vehicle.

Feedback linearization. As the articulated vehicle is a nonlinear system, it is natural to try nonlinear control techniques. One common choice is feedback linearization, which includes both input-state and input-output linearization. Nonlinear control techniques generally impose stricter requirements on the structure of the controlled system, as is also the case for articulated vehicles. For feedback linearization, the requirements are directly connected to the coupling positions and the number of trailing units. A coupling is on-axle if it is directly on a trailer's axle or on the real axle of the two-axle leading unit ($l_{ic2} = l_{ie2}, i = 1, 2, \cdots$ in Figure 2-1); otherwise, it is an off-axle coupling that has a nonzero longitudinal distance between the coupling and the axle in the unit's local coordinate system ($l_{ic2} \neq l_{ie2}, i = 1, 2, \cdots$ in Figure 2-1).

Feedback linearization is more commonly used in single-level controllers. Start with single articulated vehicles. Initially, the feedback linearization was thought not to be directly applicable for reversing in [54]. This is because the initial transformation of the system is designed so that the transformed system is stabilized only when the trailer is moving in the desired path's positive direction, which requires the vehicle to drive forward in the paper. The workaround in [54] is to introduce a further time scale transformation, equivalent to transforming the linearized model into a scaled time domain. Then linear control techniques can be applied to the transformed system. [33] shows that if the system dynamics are directly modelled in the time domain, then the standard feedback linearization procedure is sufficient. For single articulated vehicles, feedback linearization is feasible for both on-axle coupling ([30], [54]) and off-axle coupling ([33], [48]). However, multi-articulated vehicles with off-axle couplings cannot be feedback linearized [54]. Recall from the kinematic model, the steering angle input appears in the dynamics of all articulation angles when there is no on-axle coupling, which is linked to unstable zero dynamics. If the multi-articulated vehicle only contains on-axle couplings, then the feedback linearized system does not have any zero dynamics if the last unit's lateral offset is the output. Therefore, [32] introduces a ghost vehicle with only on-axle couplings to represent the real vehicle that has off-axle couplings. The ghost vehicle is controlled via feedback linearization to follow a ghost path, so that at steady state with the same steering angle, the ghost and real vehicles follow the ghost and real paths, respectively. The ghost vehicle states are determined by the real vehicle via a transformation that ensures steady-state compatibility.

Feedback linearization is also used in two-level controllers. It is used as a high-level controller in [34], which controls the last trailing unit as a two-axle unit with a virtual steering axle at its front coupling for lateral tracking via the virtual steering angle. As discussed in the kinematic model, the desired virtual steering direction of the virtual steering axle is the same as the desired velocity direction at that coupling. Then, the virtual steering angle is realized by determining the actual steering angle on the leading unit from the desired velocity direction or the last trailer's coupling through inverse kinematics. The feedback linearization can also work as a low-level controller, such as articulation angle tracking in a single articulated vehicle [34].

RL (Reinforcement Learning). An RL approach is used to control a single-articulated vehicle in [61]. Readers are referred to the original thesis [61] for more details.

Some of the control methods used as low-level control in a two-level control, or the articulation stabilization control, have already been introduced in the previous subchapter about articulation stabilization from a higher perspective as a combination of feedforward and feedback control. Specific control methods shared with various single-level controls were also introduced earlier. The inverse kinematics method is another popular low-level control technique that has been mentioned before and will be introduced in detail later. The main idea is to show the limit of the inverse kinematics method and to see how existing studies have to overcome its drawbacks.

Inverse kinematics. The inverse kinematics means the kinematic chain used for vehicle modeling is used in an inverse direction. Recall from the kinematic model presented in chapter 2.1 and assume no trailer steering, then the motion of a unit can be calculated from the motion of its front unit with a transform matrix T as Equation 3-3. The motion of a unit includes its longitudinal velocity in its local coordinate system and its yaw rate. Trailer steering can be included in T, but if the steering is actively controlled, then additional controllers are needed to determine T.

$$\begin{bmatrix} v_{xi+1} \\ \dot{\Psi}_{i+1} \end{bmatrix} = \underbrace{\begin{bmatrix} \cos\theta_i & -\sin\theta_i\,(l_{ic2}-l_{ie2}) \\ \dfrac{\sin\theta_i}{l_{i+1c1}-l_{i+1e2}} & \dfrac{\cos\theta_i\,(l_{ic2}-l_{ie2})}{l_{i+1c1}-l_{i+1e2}} \end{bmatrix}}_{T} \begin{bmatrix} v_{xi} \\ \dot{\Psi}_i \end{bmatrix} \qquad (3\text{-}3)$$

To see the inverse kinematics, $T(\theta_i)^{-1}$ is introduced in Equation 3-4, which gives a way to calculate the motion of a unit from its rear unit. Using the first equation in Equation 3-4 recursively can transform the desired motion on the last trailing unit from the high-level control to the desired motion on the leading unit. Then the steering angle of the two-axle leading unit can be calculated by the second equation in Equation 3-4. For the above process to be feasible, T must be invertible and δ_{1e1} must be within the mechanical steering limit.

$$\begin{cases} \begin{bmatrix} v_{xi} \\ \dot{\varPsi}_i \end{bmatrix} = T(\theta_i)^{-1} \begin{bmatrix} v_{xi+1} \\ \dot{\varPsi}_{i+1} \end{bmatrix}, i = 1, 2, \cdots, n-1 \\ \delta_{1e1} = \mathrm{atan}\dfrac{\dot{\varPsi}_1(l_{1e1} - l_{1e2})}{v_{1x}} \end{cases} \tag{3-4}$$

The invertibility of T requires all coupling to be off-axle, which is equal to $l_{ic2} \neq l_{ie2}$. When differential-wheeled robots are used as leading units ([40], [44], [46], …), the yaw rate and longitudinal velocity can be controlled in a decoupled way, which means motion in the same direction as desired by $[v_{x1} \quad \dot{\varPsi}_1]^T$ is always realizable. However, when applying this method to car-like steered vehicles ([57], [59], [74], …), the feasibility of δ_{1e1} need to be guaranteed through the high-level control. As shown in Paper C, this may result in limited motion on the last unit due to the current articulation angles. To realize the last unit motions that the inverse kinematics cannot instantaneously realize, various studies have improved low-level controllers. There is no well-established category name for these controllers; in this study, they are called inverse kinematics guided solutions because they aim to achieve an effect similar to the inverse kinematic method.

Inverse kinematics guided. The main idea behind inverse kinematics guided control is that, even though the desired motion of the last trailer cannot be achieved instantaneously with the exact inverse kinematics, it can be approached with modified inverse matrices or add-ons. The inaccuracy of T^{-1} is considered in [41], [42]. A tunable scale is multiplied by T^{-1} to account for parameter inaccuracies within T^{-1}. In [42], estimated articulation angle offsets are added in T^{-1} for vehicles reversing on a banked surface to compensate for the mismatch between the kinematic model and the actual vehicle. To avoid an infeasible steering angle, a quasi-static curvature is introduced in [57] to be the reference of the low-level control, which avoids a fast change in the reference. The above improvements still rely on T^{-1} in the low-level control. To create a workable solution for on-axle coupling, T^{-1} need to be eliminated from the solution.

A non-invertible T only means we can't fully determine the motion of a unit based on the desired motion of its later unit. However, if it is assumed that the desired motion of the latter unit is also maintained in a steady state with zero relative yaw rate, then the motion of the unit can be determined. The yaw rate of the two units needs to be the same, and the heading direction of the unit with the on-axle coupling needs to be the same as the desired front coupling velocity direction of the latter unit. The assumption that the desired motion can be reached at a steady state is also helpful for units with off-axle couplings when the desired motion requires a steering angle exceeding the steering limit via inverse kinematics. Suppose the steady state is reachable within all mechanical limits. In that case, the lower-level controller can be retargeted to stabilize the vehicle at that state and realize the desired motion of the last unit when the steady state is reached. This idea of tracking steady states is used in various studies. For single articulated vehicles, the tractor yaw rate is controlled by a feedforward controller and a proportional (P) controller based on the articulation angle error [70]; proportional–integral (PI) controllers based on the articulation angle error are used to control the motion of the leading unit in [38], [47], [72], where the integral part enables the error to go to zero without a feedforward part; a proportional–integral–derivative (PID) controller is used in [64]. The same idea is also used with multi-articulated vehicles: for a double-articulated vehicle in [84], the steering angle is the sum of a feedforward steering angle corresponding to the desired steady state and two proportional parts that depend on two articulation angle errors; for an n-trailer system, the yaw rates of on-axle units are controlled by a feedforward controller based on the desired steady state and a P-controller based on the articulation angle error ([62], [75]) or the yaw rate error ([18]), and it also works on off-axle units. A problem with multi-articulated vehicles is that the number of independent controllers and tunable parameters increases with the number of articulation joints, making them difficult to tune. A workaround given in [18] is to use an optimization method with an introduced cost function. Other methods that treat the system as a whole and use SIMO or MIMO control methods have been previously presented and can also simplify the tuning. There are still many tunable parameters, but they can be more interpretable.

The last piece in the two-level control is the high-level control, which determines the desired motion of the last trailing unit based on the tracking errors. These control methods are initially designed for non-articulated vehicles, including differential wheeled vehicles and car-like two-axle vehicles. They are divided into three categories based on how the desired motion is calculated from the tracking error: geometric, geometric-influenced, and non-geometric methods.

Geometric methods. Geometric methods create a desired path between the control point linked to the vehicle and a reference point on the reference path. One popular geometric method is the pure pursuit method, which is used for single articulated vehicles ([39],[67]) and multi-articulated vehicles ([68], [81], [82]). The control point is at the axle center of the last trailing unit, and for a trailer with a bogie, the center of the equivalent axle is used. The reference point is selected as the point with a given lookahead distance from the control point. The lookahead distance is the only tunable parameter in the pure pursuit method. When there are multiple solutions for the lookahead point, the one ahead in the direction of travel should be chosen. The desired path is a constant-radius arc connecting the control point and the reference point, which is tangential to the longitudinal direction of the last unit at the control point. The tangential criteria are based on the kinematic assumption that there is no sideslip at the axle. The desired path corresponds to a desired velocity direction or a virtual steering angle at the front coupling of the last unit and is realized by a low-level controller. Finding the reference point by calculating distances from the control point to various points along the reference path can be computationally intensive. A simplified and modified pure pursuit method is used in [84], where the control point is away from the last unit's axle with a lookahead distance in its negative longitudinal direction. The reference point is the point of the reference path that is on the lateral direction of the last unit from the control point. The desired path is a constant-radius arc connecting the reference point and the center point of the last unit's axle instead of the control point.

Geometric-influenced methods. Unlike geometric methods, which assume a specific path that leads the last unit towards the reference path, geometric-influenced methods do not provide a path but instead yield a desired motion from geometric arguments. The vector fields orientation (VFO) method is one of the geometric-influenced methods that appear in many existing solutions ([44], [49], [62], [70], [75]). VFO was initially designed for differential wheeled robots, and its control inputs are the yaw rate and the longitudinal velocity at the axle center. VFO is applied at the last trailing unit in articulated vehicles; therefore, the control point is the axle center of the last unit. There is no special rule for selecting the reference point, other than that it should move along the reference path. The desired longitudinal velocity is the longitudinal component of a convergence translational velocity vector in the unit's local coordinate system. The convergence translational velocity vector consists of a feedforward component that equals the velocity vector of the moving reference point and a feedback component proportional to the vector from the control point to the reference point. Similarly, the desired yaw rate is a combination of a feedforward component that equals the yaw rate of the convergence translational velocity vector and a feedback component proportional to the heading difference between the last trailing unit and the direction of the convergence translational velocity vector. With additional smooth requirements on the reference path, local asymptotic stability can be reached in path-following if the desired motion can be realized exactly. Depending on the vehicle configuration and the low-level controller, VFO stability may not be maintained if the desired motion is not exactly realized. A VFO-based control is developed in [62] to solve the average path-following problem, which aims to minimize the swept area around the reference math. The control point in the high-level control is replaced by a virtual point with position and heading based on a weighted average of all units. Then, the VFO is applied to the new control point. The low-level control is adjusted coordinately to act as a weighted inverse kinematic, where the weight of the inverse kinematic from a given unit to the leading unit equals the weight of its position and heading at the new control point.

VFO has only one reference point in each instance. Similarly, other higher-level controllers use only a single reference point. Those controllers usually use a linear combination of different tracking errors between the control point and the reference point to determine the desired motion of the last trailer. Start with single articulated vehicles. The sum of the scaled lateral, heading, and curvature errors determines the desired articulation angle in [38], [51], [38], [72], and the desired articulation angle rate in [47]. The desired curvature equals the sum of the scaled heading and lateral errors in [50], additional feedforward curvature from the reference point is added in [41], [42]. The desired yaw rate of the trailer is a combination of a feedforward term proportional to the reference curvature and a feedback term based on a PID controller using the lateral and heading errors in [64]. The lookahead concept is explicitly stated only in [51], where a lookahead distance is used between the control point and the trailer's axle to avoid aggressive responses. The lack of lookahead in single articulated vehicles is likely because the desired motion of the trailer can be realized instantaneously or with an accepted delay. For the double-articulated vehicle in [59], the lookahead distance between the control point and the last trailer's axle is the product of the last trailer's local longitudinal velocity and a time constant. The reference point is in the local lateral direction from the control point. The desired motion of the last trailer in [59] is either its front coupling velocity direction or its virtual steering angle, which is proportional to the induced angle between a line from the control point to the axle center and a line from the reference point to the axle center. Lookahead is also applied with the multi-articulated vehicle in [73] by moving the control point away from the last trailer's axle in the negative longitudinal direction. Then, the desired longitudinal velocity on the last trailer is proportional to the longitudinal position error in the local coordinate system, and the desired yaw rate is proportional to the lateral position error in the local coordinate system.

Many driver models use more than one reference or preview point, which can also be seen in high-level controllers. The three control points in [85] are one point on the last trailer's axle center and two points that are away from the axle center in the local negative longitudinal directions with two different lookahead distances. Each control point has a corresponding reference point on the reference path, located in the local lateral direction from the control point. The correction angle is the angle between the line from the axle center to a control point and the line from the axle center to the reference point corresponding to the control point. Two correction angles exist in [85] with two pairs of control-reference points. The desired motion of the last trailer is defined as the virtual steering angle at its front coupling, which is the sum of the weighted average of the two correction angles with a proportional gain and a weighted integral of the lateral error between the control point at the axle center and its reference point. The same setup of control and reference points is used in [74]. An additional element is added to the calculation of the desired trailer virtual steering angle, proportional to the heading difference between the last trailer's current heading and the path heading at the middle reference point. A detailed tuning approach is provided in [74]. It is worth mentioning a well-known path-following controller developed in [109] for non-articulated vehicles, which also uses multiple preview points. The controller calculates the steering angle as a weighted sum of a single heading error and multiple lateral errors with saturations on three different levels. The controller's tracking performance and robustness across different vehicle configurations make it a potential choice for a high-level controller for reversing articulated vehicles.

Non-geometric methods. Non-geometric methods are control methods used in high-level control that do not originate from the geometric interpretation of the path-following problem. The expression of control input from a non-geometric method may have a similar structure to that of a geometric or geometric-influenced controller; for example, the desired motion is a sum of scaled errors between the control point and the reference point. However, the control parameters from a non-geometric method are not designed for interpreting geometric properties. High-level control is used in [31] uses linear and nonlinear state feedback control, where the controller's inputs are position and heading errors between the control point on the last trailer's axle and an externally assigned reference point on the reference path. Differential dynamic programming applied in [65] is another non-geometric method. The control point is located at the center of the last trailer's axle, and the reference point is a point on the reference path that is oriented in its local lateral direction. The control input to the trailer is its virtual steering angle, which, together with the known trailer longitudinal velocity, defines the desired motion of the trailer. The cost function included weighted lateral position, heading errors, and weighted input and input rate.

3.2.4 Specialized control methods

The following control methods are considered specialized control methods because, unlike the standard control methods above, they may lack well-defined recipes and are difficult to generalize across different articulated vehicles. In standard control methods, there are also vague parts of the design methods, but these mainly concern parameter tuning, which can result in the designed controllers performing poorly. The specialized control methods used in existing studies include expert systems, fuzzy logic, backstepping, and Lyapunov-based control design.

Expert system. An expert system is a control technique that solves a control problem based on human decision-making. A controller designed with this technique successfully solved parking problems for a single-articulated vehicle in [29]. Six system states are used for decision making, including tractor and trailer headings, articulation angle, tractor and trailer lateral offsets, and tractor longitudinal offset. Each state is described with two to four linguistic values. The control logic is designed based on those states with linguistic values to determine the tractor's drive and steering directions. It can be difficult to expand this expert system to multi-articulated vehicles, as both the required number of states and the resolutions of the linguistic values are expected to grow, leading to significantly more complex decision logic.

Fuzzy logic. Fuzzy control is based on fuzzy logic, in which the controller's output is a weighted combination of outputs from multiple controllers, each of which has a weight determined by specific system states. A fuzzy logic controller is developed in [110] for reverse direction control of a single articulated vehicle, and used as a low-level control in [39]. Two control rules are set up, both of which control the curvature of the tractor to be equal to a linear weighted combination of the heading angle errors of the tractor and the trailer. The weights in one control rule are tuned for zero articulation angle and for the articulation angle limit in another control rule. The fuzzy sets are dependent on the articulation angle. When tracking the zero heading, the stability of this fuzzy controller is proved with the Lyapunov direct method, and the ability to avoid jackknife is also guaranteed. However, stability and jackknife avoidance are no longer guaranteed when tracking a dynamic reference. A triple-articulated vehicle in [87] is modeled using a fuzzy model comprising two linear sub-models, with rules that depend on the last articulation angle and the last trailer's heading error relative to the reference path. A linear state-feedback controller is designed for each sub-model. The performance-related problems, including stability, decay rate, input and output constraints, and disturbance rejection, are formulated into linear matrix inequalities. Similar to [39], the performance is not guaranteed globally. Meanwhile, the rules in fuzzy logic might not be universal across different vehicles, as articulation angles beyond the last one can also significantly affect system dynamics.

Backstepping. This is a nonlinear control technique used in [68], [79], [111] as single-level controllers. Compared with feedback linearization in standard control methods, backstepping also works with on-axle trailers in [79]. The stability of backstepping control is usually proved with Lyapunov's direct method, which is valid if a Lyapunov function exists. No constraints on states and inputs are considered in [68], [79], [111]. This is generally acceptable for differential-wheeled robots with trailers because the yaw and longitudinal motions of the leading unit are decoupled, and they are usually designed without imposing an articulation angle limit. However, the steering and articulation angle limits on actual articulated trucks can make the stability conclusion invalid. Besides mechanical limits, backstepping is sensitive to model uncertainties and disturbances, so theoretical stability may not be achieved in real operations. The recursive structure of backstepping makes the desired control input more complex as the number of articulations increases. The drawbacks of backstepping make it a less popular choice in literature.

Lyapunov's direct method. This method evaluates the stability of a system based on a Lyapunov function. Lyapunov's direct method itself is not a control method, but it can be used to find a controller that makes the closed-loop system satisfy the stability criteria. The process involves designing both the Lyapunov function and the controller. Backstepping controllers introduced earlier can also be listed under this category. Single-level controllers that are asymptotically stable based on

Lyapunov's direct method are developed for single-articulated vehicles in [35], [36]. Two separate controllers are developed for the longitudinal velocity and the yaw rate of the differential-wheeled leading unit in [35]. The longitudinal velocity control is designed to drive the trailer's position and heading angle errors to zero, and its stability is proved with a nonquadratic Lyapunov function. The yaw rate control is designed with a quadratic Lyapunov function for the heading error of the leading unit. Articulation angle is not actively controlled [35], but stabilized through reference-heading tracking across two units. The Lyapunov-based controller developed in [36] stabilizes the tractor's lateral, heading, and articulation errors, considering both steering and articulation angle limits. Lyapunov functions are used for low-level control design in [39], [51]. Heading errors of two units are controlled in [39], and the quadratic Lyapunov function used for stability proof has a state-dependent Lyapunov matrix. The state-independent quadratic Lyapunov in [51] only includes the articulation angle error; its negative derivative is maintained by requiring an articulation angle rate proportional to the articulation angle error. The sign of the proportional gain is determined by Lyapunov's direct method. Still, the magnitude needs to be tuned so the articulation angle rate can be realized within the steering angle limit. A Lyapunov-based high-level controller is used for a multi-articulated vehicle in [46], which was initially developed for non-articulated vehicles in [112]. The stability of position and heading tracking in [112] can be proven using a nonquadratic Lyapunov function if the reference path is sufficiently smooth. Even though Lyapunov's direct method offers a way to help design asymptotically stable closed-loop systems, it is not trivial to define a desired Lyapunov function and related controllers, especially when the number of controlled states is large. Designing a Lyapunov-based high-level or low-level controller may be easier because there are fewer controlled states. Still, the stability might not be preserved when combined with the other part in a two-level controller.

3.3 Path planning

For the various path-following solutions discussed before, a reference is one of their essential inputs. The reference that guides a vehicle to its destination can be a setpoint or a continuous path on the road plane. The complexity of the path-planning problem depends on both the vehicle's dynamics and its external environment. This subchapter will first present the objectives of path planning, starting from the simplest case of moving an omnidirectional wheeled robot in an obstacle-free space to parking an articulated vehicle in a loading dock with obstacle avoidance. Then, summaries of the planning algorithms used in the existing research from Table 1-1 are given.

3.3.1 Objectives

The discussions are limited to path planning in a 2-dimensional (2D) plane. Under this limit, the simplest task in path planning is to plan a path for a holonomic system in an obstacle-free environment from its current position to a target position, which is also the only objective. A holonomic vehicle, like an omnidirectional wheeled robot in 2D, can move in all directions instantly. Therefore, if only the translational positions of the start and end points are taken as the constraints, then any path shown in Figure 3-5.a connecting the start and end points can be followed by the holonomic vehicle. There is no need for any smooth requirement on the path for it to be exactly realized by the vehicle. Even if the start and heading directions are constrained, all paths remain feasible for holonomic vehicles, as they can control their yaw motion independently of their translational motion. All paths in Figure 3-5.a are also feasible for nonholonomic differential wheeled robots if they are allowed to stop at points on the path where the path heading is discontinued. However, as discussed earlier, the differential wheeled robots differ from car-like tractors or rigid trucks. The effects of car-like nonholonomic vehicles on path planning will be discussed later. In real applications, it is natural to select the optimal solution when multiple solutions are available. The solution can be trivial if the objective is to find the shortest path in Figure 3-5.a, but it can be a more complex problem if it is to find the fastest path. The fastest solution can depend on the vehicles and may not be the same as the shortest path. An inspired reference for this is the micro mouse competition [113], but there were obstacles. For reversing articulated vehicles, it is vital to find a feasible solution with minimal computational time rather than an optimal one. However, optimization is also used to find a feasible solution, which will be discussed in the context of planning algorithms.

Car-like steered nonholonomic vehicles introduce a feasibility objective for the path-planning problem. A planned path is feasible for a nonholonomic vehicle if it can be exactly followed without violating the vehicle's nonholonomic constraints. Paths in Figure 3-5.b are Dubins paths, which are a set of feasible solutions that the rear axle center of a car-like vehicle can exactly follow. Details about the Dubins path are given later. A higher requirement, linked to the smoothness of the path, is that the planned path be followed by the vehicle with smooth velocity and steering, which is addressed in [92] for both non-articulated and articulated vehicles. For nonarticulated vehicles, the path should have continuous curvature. For a path planned for the nth trailer in an articulated vehicle, the path curvature should be continuous at least for its nth derivatives. A path that the trailer's axle center of a single-articulated vehicle can follow between a start and end point, with position and heading constraints, must be described by a ninth-order polynomial with eight variables. Details on how to determine those variables in optimized or simplified ways are given in [92].

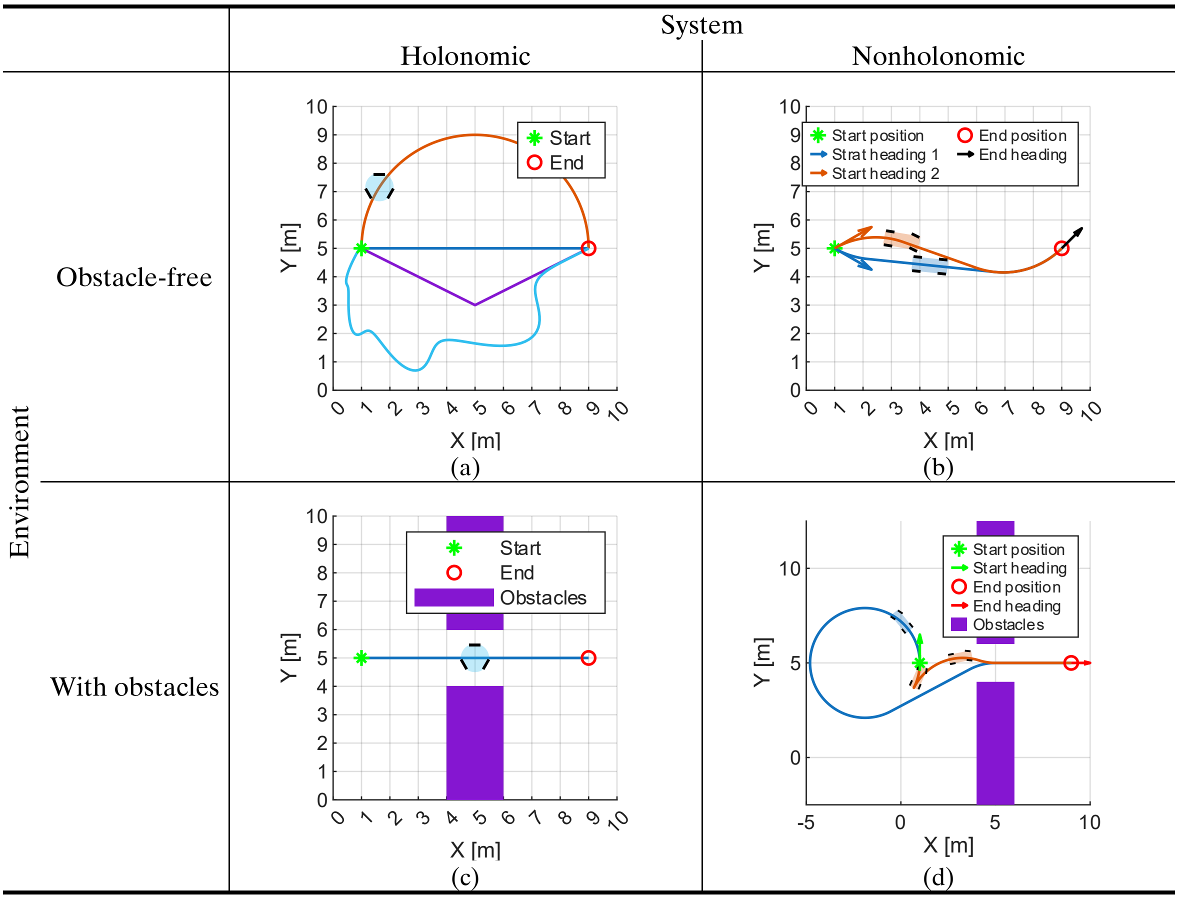


Figure 3-5 Path planning for different systems in different environments

Obstacle avoidance is another objective that must be achieved through path planning. The obstacles in this study are mainly parked vehicles, stationary structures within the shunting area, and the shunting area boundary. Moving obstacle avoidance through time-based planning is not considered in existing solutions. The presence of obstacles complicates path planning for both holonomic and nonholonomic systems. Besides the examples shown in Figure 3-5.c, no further discussion is done on holonomic systems in this study. Figure 3-5.d shows that a shorter path may be reached in the car-like nonholonomic system that can change its driving direction. The change in travel direction creates a cusp in the path, which is why bi-directional planning is referred to as cusp technology in some studies. The cusp technology has also been proven to be useful for articulated vehicles in existing solutions, which will be introduced later. Besides Figure 3-5.d, it is worth noticing that for articulated vehicles, when the last trailer of the leading unit is following a main path, the paths of the rest units might have large offsets relative to the main path, which must be considered while path planning. Therefore, depending on the path-following design, the planned path may need to guide the articulated vehicle to reach a desired pose at the end point, such as the straight pose when reversing into a loading dock.

In summary, for shunting an articulated vehicle, the path-planning function needs to provide a feasible, collision-free path that allows the vehicle to drive from its initial state at the start point to its desired state at the end point. Shunting means the travel direction is not limited to forward or reverse; changing direction is allowed if a feasible path requires it. Feasible means that the vehicle's nonholonomic constraints are obeyed. Collision-free not only includes avoiding collisions with external obstacles but also avoiding inter-unit clashes. The current state can be considered determined, which in turn determines both the vehicle's position and pose. The position also includes the heading of one unit, and all articulation angles define the pose. The desired state has more freedom. For parking maneuvers, both the position and pose are usually given in the desired state. For coupling and decoupling, it may not be necessary to always include the pose in the desired state.

3.3.2 Planning algorithms

Path planning algorithms used in Table 1-1 are divided into six categories. The first two categories, Dubins path and set-point tracking, are primarily used as local planners that generate feasible, inter-unit clash-free paths between two points without considering the external environment. Third, scenario-based planning provides full paths, but only for predefined scenarios. The fourth and fifth categories are based on the lattice and the rapidly-exploring random tree (RRT), respectively. They are designed as global planners for finding a path between two points with obstacles in the workspace. Depending on the design of a specific algorithm, the global planner's planned path may need to be refined by a local controller to make it feasible for the vehicle. Some algorithms have become less popular in recent years, but inspired alternatives have been introduced in the sixth category, called others.

Dubins path. Dubins path is used for path planning with a single-articulated vehicle in an obstacle-free environment in [61]. It is used in multiple stages within an RRT-based path planning for a single-articulated vehicle in environments with obstacles [51]. The original Dubins path is designed for a specific two-axle vehicle, called the Dubins car, whose steering angle can only be zero or turn left or right with the same magnitude. The steering magnitude is usually set to a value close to the maximum steering angle, with a certain margin for practical reasons. Then, the car's rear axle center can only create three possible path sections: a straight line (S) and two constant-radius circles for left (L) and right (R) turns, respectively. Then, the path with the minimum length to move a Dubin car from one point to another point with assigned initial and ending headings in a Euclidean plane is always formed by one of the following six types of paths formed by the three possible path sections: RSR, RSL, LSR, LSL, RLR, LRL. The shortest path is called the Dubins path.

In [51], [61], the reference path is planned for the last trailer's axle center. Both use a modified Dubins path to plan the last section before the target, which includes a long straight segment at the end of the original Dubins path. In this way, the articulated vehicle approaches a straight pose when the trailer arrives at the target. Since the last trailer cannot drive exactly like a Dubins car, the Dubins path will not be exactly realized by the vehicle. The radius of the circular section needs to be determined so that the closed-loop system formed by the vehicle and its path-following controller is stable while tracking the Dubins path. The original Dubins path is used in [51] within and after the RRT process. Within RRT, the distances from a new state to existing states are computed as the lengths of Dubins paths, providing more realistic distances. The state here comprises the vehicle's position in an Euclidean plane and its heading. Then, the new state and the nearest existing state are connected with a Dubins path in the Euclidean plane. Details about RRT will be discussed later. After RRT finds a solution, the Dubins path is used again for path optimization. The optimization algorithm attempts to skip route points in the RRT solution. If a Dubins path connecting non-neighboring route points is still collision-free, then intermediate route points can be removed, resulting in a shorter path.

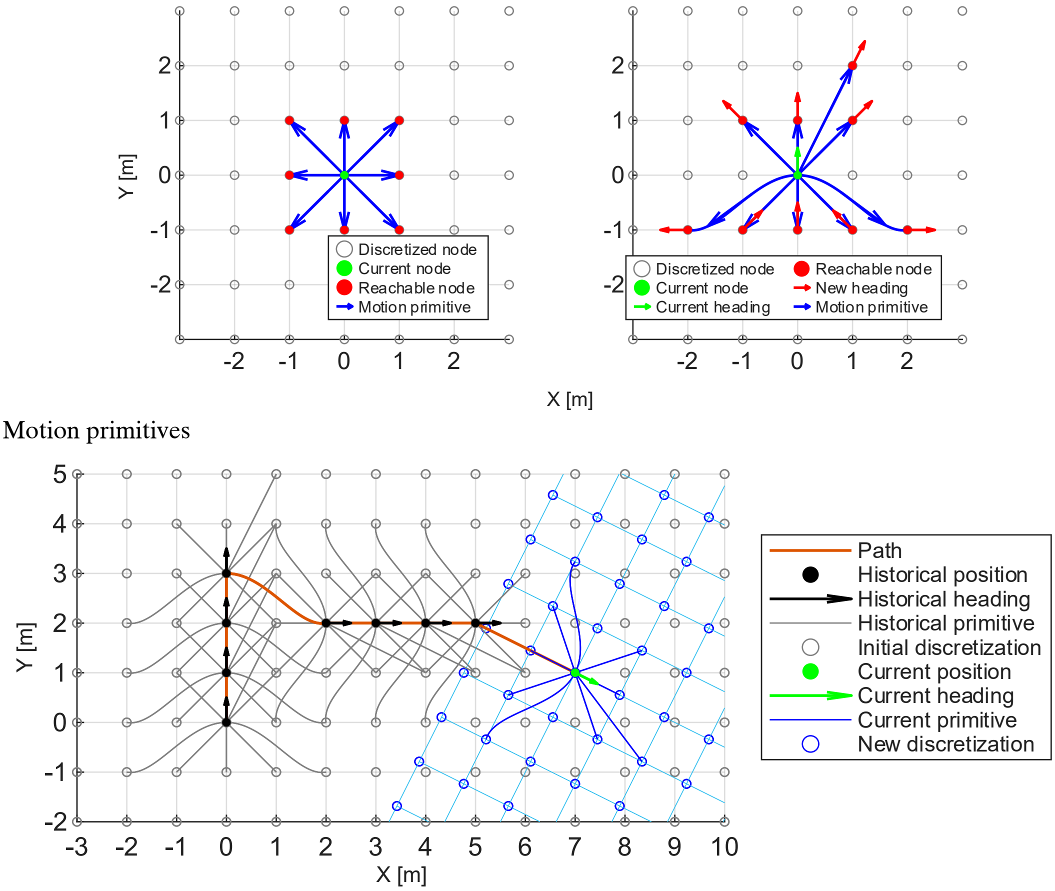
Setpoint tracking. Setpoint tracking is a general concept in control theory that refers to controlling a system to reach a desired state. Given a setpoint that defines the desired vehicle position, or even includes heading and pose, if a path-following controller can control the vehicle to reach the setpoint, then a feasible path is created. This means that a simulation of the closed-loop system formed by a vehicle model and a path-following controller, with a setpoint as input, is also a path planner. The expert system used in [29] can be the controller, but since its behavior is complex to tune, it can be troublesome if the initial plan is not collision-free with external obstacles. VFO-based path-following controllers ([44], [45] [49], …) can be better choices as there is a clearer link between control parameters and path shapes. However, they are still not capable of actively planning collision-free paths, which limits them mainly to act as local planners. Path-following controllers that handle obstacle constraints may serve as global planners. Such as several MPC-based controllers developed in [63]: lane-bounded reversing control (LBRC), adaptive lane-bounded reversing control (ALBRC), and adaptive bi-directional control (ABC). LBRC implements the shunting area boundary as a constraint to ensure the entire vehicle remains within the given area. ALBRC introduces the self-tuning concept into LBRC to improve solvability. Both LBRC and ALBRC limited the vehicle to reverse motions. A rule-based outer layer is added to ALBRC, forming ABC, which allows the vehicle to reposition itself by driving forward if the setpoint cannot be reached via pure reverse. However, when faced with complex external environments, they may not be able to provide a feasible solution.

Scenario-based planning. Scenario-based planning focuses on specific scenarios, such as parallel ([69], [90], [95]) and perpendicular ([58], [74], [95]) parking. Compared to later, more general planning algorithms, they have limited applications but are usually more computationally efficient in their intended use scenarios. Many of those scenario-based planners are based on the open-loop kinematic behavior of articulated vehicles. The planner that uses only a modified Dubins path in [61] can be seen as a very simple scenario-based planner, whose application scenario is to park the vehicle in a straight pose at a given position, with no obstacles in the operating area. [58], [74] focus on a more typical perpendicular parking scenario in freight terminals, where articulated trucks usually drive forward and perpendicular to the loading docks when they enter the shunting area, then reverse into a narrow loading dock, which is likely to have parked vehicles on its left and right sides. Therefore, the planned paths from [58], [74] include a forward section and a reverse section. The reverse section is designed with forward driving in mind, as a vehicle leaves the loading dock. Therefore, there is a straight section from the final position along the loading dock's longitudinal direction to prevent collisions with parked neighboring vehicles. After the last trailer reaches the end of the straight section, a near-constant steering angle, including transient behavior, is applied while driving forward. Then, the curved path becomes another section of the reverse path. The ideal of using a constant steering angle is that the need to change steering while reversing will also be reduced. The forward section is created by a Dubin path that connects the initial position of the last trailer to a selected point on the curve portion of the reverse section. This aims to guarantee the designed path for the last trailer has the same heading at the switch or cusp point. However, the vehicle pose is not guaranteed at the switch point for the same vehicle that reached from different path sections. Therefore, a vehicle that comes from the forward section may not be able to follow the curved portion of the reverse section with a constant steering angle. This is handled by a path-following controller in the two original studies.

Scenario-based planners can also include controllers to make them more adaptive to changes in similar scenarios. Parallel parking paths are planned for the leading unit of a single-articulated vehicle with a planner that includes fuzz control in [90]. The planned path contains three sections. In the first section, the vehicle is driving forward along the side of a parallel parking slot. The last section is followed during reverse, but is planned in the forward driving direction, ensuring the vehicle can exit the parking slot without colliding. The first and last sections are designed to intersect at a point where the tractor has the same heading angle, but not for the trailer. An explicit solution is provided for the middle section, so the tractor can drive forward from and reverse to the intersection via a cusp point, with two different paths, to create the desired trailer heading difference. The fuzzy logic controls the position of the intersection point based on parking size and external space constraints, enabling the planner to handle different parallel parking scenarios. MPC is used for path planning in [95] for both parallel and perpendicular parking of a single-articulated vehicle. Unlike tracking a single setpoint at the destination, each scenario is divided into multiple stages with multiple target areas. Target areas for connect stages have an overlap, and the MPC is designed to move the vehicle from one area to the next via the overlap. The multiple sub-targets make it easier to modify the entire planned path for given scenarios.

Lattice-based planning. The general idea of lattice-based planning is to discretize the entire planning space into lattice points, and the controlled system can move between them according to specific rules. The planner's objective is to find a route between the initial and the target lattice points. Due to discretization, the initial and target states may not lie exactly on lattice points, so it is usually acceptable to choose a nearby lattice point. Before introducing various applications of lattice-based planning in the literature, several points that may easily be misunderstood are clarified. First, the lattice can have more than two dimensions even if the path is planned in a 2D Euclidean plane. For example, if the heading angle of a two-axle nonarticulated vehicle needs to be controlled during path planning, the system must be described using a 3D lattice. Vehicles with the same position in a 2D Euclidean plane with different headings do not have the same state. The dimension of the lattice will continue to grow for articulated vehicles when their pose needs to be controlled; an additional dimension is needed for each articulation angle. This means the number of lattice points and the number of possible moves between them can grow exponentially. Second, the possible moves from a lattice point under specific rules are not limited to nearby points. Possible moves are often called motion primitives in robotics. Specific rules include constraints that limit the system's dynamics, such as nonholonomic and mechanical constraints. Assume positions on a 2D Euclidean plane are discretized into nodes on a square grid. When introducing examples for motion primitives, it is very common to assume there is an omnidirectional robot that can travel in eight directions and reach nearby nodes, as shown in the left part of Figure 3-6.a. The gray points are discretized position nodes, the robot's current nodes are in green, and nodes that can be reached with one step are marked with red. However, if the robot is directional and has a special dynamic constraint, then the motion primitives on the right of Figure 3-6.a may apply. Arrows represent the robot's current heading direction and heading directions when it reaches new nodes. The robot's directionality is represented by its heading-dependent motion primitives. The special dynamic constraint is that when the robot is moving towards its left or right, it will not reach the nearest point on the position grid; instead, it will reach the second-nearest point. This example is not intended to make the problem more complex, but to show how the motion primitives become more complex due to additional dimensions and constraints, such as those from articulated vehicles. Third, a static lattice and motion primitives may not be sufficient for the planning, especially when the controlled system has some directional behaviors. This is explained by Figure 3-6.b, when the directional robot moves along the orange path to the green node on the grey position grid, none of the motion primitives from the blue node are able to reach a node on the grey grid. Either the position grid or the motion primitives need to be replanned to ensure there are feasible connections between lattice points. In Figure 3-6.b, a new blue position grid is introduced from the blue node to solve the problem. It is not ideal to create a new search space. Approximating a new point generated by a motion primitive to the nearest existing lattice point is another approach; however, an additional feasibility check may be needed, which can be computationally intensive for the nonlinear dynamics of articulated vehicles.

The motion primitives are generated in different ways in reversing control research. Most of the studies limit the number of motion primitives to reduce computational complexity for subsequent search algorithms. For a single-articulated vehicle, the motion primitives used in [66], [94] are generated by moving the vehicle between selected points in a discretized 4D lattice with optimal control. The 4D lattice includes the tractor's longitudinal and lateral positions, the tractor's heading, and the articulation angle. The optimal control has considered constraints of the steering system and the path curvature. [93] limits the number of motion primitives of a double-articulated vehicle by assigning all primitives to have zero steering and articulation angles as the initial and end states. The motion primitives include only a limited set of direction change and lane change maneuvers. The paths of all motion primitives are planned to have continuous steering angles. The planning lattice is reduced to 3D, which only includes the tractor's longitudinal and lateral positions and heading. The swept area of each motion primitive is also precalculated for collision check while searching. Similarly, [68] uses limited-motion primitives with a triple-articulated vehicle that includes heading changes in eight directions. The motion primitives are planned using an optimization framework with costs and constraints on the steering angle, rate, acceleration, and articulation angles. In [86], the motion primitives of a double-articulated vehicle are designed in its complete 5D lattice, but with additional rules to limit the number of motion primitives for the searching algorithm. The remaining primitives are designed using optimization techniques to reduce steering angle, rate, and acceleration. Articulation angles are included in the cost function for the optimization only when the vehicle is in reverse. A dynamic lattice with changeable dimensions and resolutions is used in [96] for a double-articulated vehicle. The 5D system is reduced to 3D, which only includes the last trailer's position and the last articulation angle for simple reverse cases. High dimensions and high resolutions are used when there are high requirements on the vehicle position and pose. All the above motion primitives are designed without considering external obstacles. External obstacles are considered in the later searching process from each point to extend. Based on the state of the selected point and the external environment, certain motion primitives might become infeasible due to collisions with external obstacles.



(a) Motion primitives

(b) Path-planning solution

Figure 3-6 Lattice-based planning in 2D-space for a two-axle nonarticulated vehicle

The motion primitives give feasible connections between lattice points. Then the task is to find a complete route from the initial lattice point to the destination point. Two algorithms used in existing solutions to solve lattice-based problems are the A* algorithm, used in [66], [68], [86], [94], [96], and Dijkstra's algorithm used in [93]. To briefly introduce these algorithms, assume all points on a lattice have infinite cost and are marked as unexplored. Both algorithms start searching from the initial point in the lattice. Costs are calculated for all points that can be directly reached from the initial point. For Dijkstra's algorithm, the cost at each point is the actual cost of moving from the initial point to this point. For A*, the cost is the cost from Dijkstra's algorithm plus the estimated cost from this point to the destination. The initial point is then marked as explored. The point to extend is the unexplored one at minimum cost. If the minimum cost occurs at multiple unexplored points, select one of them. Then, costs are calculated for all unexplored points reachable from the point to extend; meanwhile, the point to extend is marked as explored. Iteratively, a new point to extend is selected and marked as explored as before, and costs are calculated for all reachable unexplored points from it until the destination point is reached. Before a point is marked as explored, its cost may be updated several times. This occurs when the cost to this point is lower from a new point than from any previous point. When a point receives a cost update, it also needs to record which explored point caused it. Finally, the complete route can be traced back from the destination to the initial point. For further details and their application, see the reference above.

As seen in the introduction above to A* and Dijkstra's algorithm, the cost influences the search direction and determines which aspect of the final solution is optimal. The cost of travel for Dijkstra's algorithm applied in [93] is a weighted sum of the absolute heading change and the path length of the last trailer. For A* used in [96], the cost to move is designed to promote driving forward in a straight line. Therefore, the cost to move increases with changes in the vehicle's heading and driving direction. The estimated cost to the destination is a weighted sum of the last trailer's position, heading, and articulation angle errors. [86] uses a variant of A*, which still uses costs to move and to the destination. The cost to move for a motion primitive is derived from the same cost function used when optimizing the primitive. The estimated cost to the destination from a lattice point is the maximum value between the Euclidean distance and a value from precomputed heuristic look-up tables (HLUTs). HLUTs compute costs based on planning without considering obstacles with Dijkstra's algorithm [114].

Preference paths from previous searches or predefined by experiences are used to increase the efficiency of lattice-based planning in [96]. Instead of planning the whole route from the initial point to the destination, the planning is divided into three sections. The first section guides the vehicle onto a preferred path. The vehicle then follows the preference path. The last section determines an exit point for the vehicle from the preference path and plans the final path to the destination. In this way, existing knowledge is utilized, reducing the complexity of new planning.

Rapidly-exploring random tree (RRT). As indicated by the name, randomness is one of the basic concepts of RRT. RRT is non-deterministic, which sets it apart from the previously discussed planning algorithms. Before going through the applications of RRT and its variants on articulated vehicles, a short introduction to the original RRT is given based on [115]. Assume a path is planned for a car-like nonarticulated vehicle in a bounded space with stationary obstacles. The car's state is defined in the global coordinate system, including longitudinal and lateral positions and heading. The initial state is known and is in a vertex list. RRT includes six steps. First, select a random state from the planning space, which is not necessary in the obstacle-free space. Second, the nearest state to the random state is found from the vertex list. A customized distance metric in the state space is needed to determine the distance between two states. Third, the car's input is determined by the random and the nearest states. Depending on how the problem is formulated, the car's input can be multidimensional and dependent on the state. The input should minimize the distance between the nearest and random states. The input itself and its controlled car must not violate any constraints, such as the maximum steering angle for the input and the car's collision-free condition. Fourth, a new state is generated by simulating the car from the closest state using the input from the previous step and a fixed time step. Depending on the time step selection, the new state may reach the random state, but not necessarily. Fifth, the new state is added to the vertex list and will be used to find the nearest state for later random states. Sixth, the nearest and new states, along with the control input, are added to an edge list. By this step, a loop in RRT is complete. Then, select a new random state and repeat the loop above until a new state is close enough to the target state or the maximum number of iterations is reached. If a state in the vertex list is close enough to the target state, a feasible path back to the initial state can be established using information from the edge list. Otherwise, RRT failed to provide a solution. There are many freedoms within the first four steps of the original RRT loop. How they are modified in the context of reversing control research will be introduced next.

Several modifications have been made in [51] on RRT for reversing a single-articulated vehicle. During planning, the vehicle is reduced to a 3D system including the trailer's lateral and longitudinal positions and heading. The distance metric uses the length of the Dubins path between two states. Even though the trailer cannot precisely follow the Dubins path, it still provides a better estimate than the Euclidean distance between two positions. The Dubins path is also used to connect a random state with its nearest state. Then the planned path between the nearest state and a new state is a part of the Dubins path. The collision check for the planned path is performed in two steps. First, checking whether the path itself has a certain margin from the boundaries of obstacles, thereby rejecting a new path that has a high likelihood of collision without simulating the whole vehicle. If the first check passes, the entire vehicle is simulated following the new path in the second step. The swept area still needs a certain margin from obstacles for the new path to be classified as collision-free, to account for mismatches between the model and the real vehicle. Not all new states reachable via collision-free paths will be added to the vertex list. A new state will be kept only if it is not too close to any existing states in the vertex list. This promotes allocating resources to exploring a larger area in the planning space to increase the chance of finding a feasible solution between the initial and final states.

A variant of RRT, Closed-loop RRT (CL-RRT), is used in [81], [97] to control double-articulated vehicles. The difference between the original RRT and the CL-RRT is that, instead of directly sampling from the planning space in the original RRT, the CL-RRT samples from the input space of a stable closed-loop system. The random state CL-RRT is the simulated system state based on the sampled input. In these two studies, the closed-loop system consists of a double articulated vehicle and a two-level path-following controller. The randomly sampled inputs are slightly different between [81], [97]. Both include the longitudinal and lateral position errors of the last trailer; additionally, the input of [81] includes a desired velocity, and that of [97] includes the heading error of the last trailer.

Besides using CL-RRT, [81], [97] also include other modifications to improve the searching performance. One of them is that the customized distance metric between two states is precalculated into lookup tables. Lookup tables in [81] are based on position differences and drive directions. Simulations are performed using the closed-loop system to bring the vehicle from a given initial state to the positions specified in the lookup tables, with the corresponding drive direction. If a position is not reachable, it can be considered as having an infinite distance. The distance for the reverse cases includes the path length, weighted changes in both articulation and heading angles. For forward driving, only the path length is used to compute the distance between two states. The distance used in [97] also accounted for the heading angle error when the trailer reached the target position. The random state does not always try to connect to the nearest state in [81]. For half of the time, it will try to connect to the state that has the lowest cost from the initial state. For both [81], [97], after each new state is added, it will try to connect with the target state. If the extension can reach a predefined range around the target, a solution is found. However, new random input will continue to be sampled until a given time limit is reached. Since multiple solutions can be found with this setup, the lowest-cost solution is selected using a cost function that considers the total travel distance and the weighted error between the target state and the final state within the acceptance range around the target state. The above modifications can also be applied to the original RRT.

Others. In this category, some path-planning algorithms for reversing articulated vehicles that do not fit into the other categories are presented. Start with the local planner developed for a single-articulated vehicle in [31]. The local planner receives a path from a global planner that doesn't account for the vehicle's nonholonomic constraints. The local planner follows the nonholonomic constraints by introducing cusp points when two reference points are too close to each other, given their curvature differences. A cusp point means the vehicle will follow the reference path in different directions before and after the cusp point. According to the topological property given in [31], from a given reference point with a given curvature, the feasible path from this point must stay in a specific cone with its vertex located at the reference point. If the next reference point is not within the cone, an additional cusp point is needed to make the path feasible. A circle that has the same curvature as the next reference point can be created with tangential to the global path. The cusp point is placed on the part of the circle that is within the cone. The advantage of this method is that it can turn an infeasible path for trailers into a feasible one, thereby improving the trailer's path-tracking performance. However, there is a risk that the vehicle will arrive at the destination in the wrong direction. An additional layer in the planner is needed to ensure the vehicle arrives in the desired direction.

There is a group of global planners based on artificial potential fields. One of them is the global planner developed in [116], which is used for a single-articulated vehicle in [31]. The planner involves two types of artificial potential fields. The first type of field pushes the vehicle away from obstacles and settles to a specific set of middle points determined by the surrounding obstacles, or the skeleton, as described in the original study. The second type of field attracts the vehicle to its destination. In [116], the second type of field is designed with the first type of field in mind. This ensures nonzero gradients along the skeleton of the first field and a global minimum potential at the destination. Multiple controller points are selected on the controlled vehicle for controlling multiple degrees of freedom. Since each control point will have a different position at the destination, it will also have a different set of fields of the second type. To solve the planning problem, a potential function is needed to combine the potentials from the second type of fields across different control points and to find their global minimum. Several different potential functions are presented in [116], but all of them can have multiple local minima. Two methods are presented in [116] to solve this problem. Exhaustively exploring the discretized configuration space is recommended for systems with fewer DOFs. For systems with many DOFs, the Monte Carlo method is recommended, which involves random sampling. The potential field can be formed in different ways. In [98], the potential field is created with two-dimensional Gaussian attractors placed at obstacles and a two-dimensional Gaussian repulsor at the destination. Then, an optimization problem is formulated with nonholonomic constraints imposed by the vehicle. The cost function includes vehicle states, inputs, and potential. [98] is designed for forward driving. The above solutions do not ensure collision avoidance along the planned path. If a designed potential field and cost function result in an infeasible path, it might not be straightforward to redesign them to find a feasible solution, if it exists.

The last group of planning algorithms in this category includes two multi-stage planners. The path-planning problem is considered as a nonlinear root-finding problem in [88]. The objective is to find a control sequence that guides the vehicle to the destination from its initial state, subject to multiple constraints. It can be complex for a solver to handle multiple constraints in an extensive search area. Therefore, in the first stage, the Newton-Raphson-based solver only manages equality nonholonomic constraints, and the leading unit's inputs (longitudinal velocity and steering) are assumed to be in the form of Fourier series expansions. The inequality constraints, such as the steering angle, articulation angle, and obstacles, are applied later. These inequality constraints are converted into equality constraints in [88]. For each stage, the solver uses the solution from the previous stage as the initial guess. The steering and articulation angle limits are applied in the second stage. Constraints from external obstacles are applied in the third stage. Another multi-stage planner is presented in [89]. The basic concept is that, first, a collision-free path is planned by following only the nonholonomic constraints of a car-like leading unit; then, in each iteration, the path from the previous iteration is modified to obey the additional nonholonomic constraints imposed by an additional trailer while maintaining collision-freeness. There is no limit on the planner used in the initial stage; any path planner that can plan a collision-free path for the car-like leading unit is acceptable. The first method introduced in [89] for path iteration that accounts for an additional nonholonomic constraint is called pick-and-link (PL). PL starts by picking the start and end points on the path from the previous iteration and tries to connect them with a collision-free path that obeys the new constraint set. If this is not possible, additional middle points will be added until all neighboring points can be linked by collision-free path segments obeying the new constraints. The second method is tube-PPP (Probabilistic Path Planner), which applies the PPP from [117] to a tube area around the path from the previous iteration. A sinusoidal planner is used in both methods as a local planner that can plan path segments with the new constraint set. More details are referred to the original publication [89]. For both multi-stage planners, the final solution is closely related to the initial solution from the first stage. The risk is that no feasible solution will be found for the full constraints due to a poor initial solution. As stated in [89], it is worth spending more effort on the initial solution to consider more nonholonomic constraints, especially for vehicles with many nonholonomic constraints.

3.4 Summary and discussion

An important takeaway from this survey of path planning algorithms is the importance of articulation angles. For all existing algorithms in articulation stabilization and path following, the articulation angles are used as feedback states. Almost all approaches under those two categories require the feedback states in quantitative form. Only a few use articulation angles qualitatively, and they are not intuitive to extend to vehicles with multiple articulations. The reason for requiring articulation angles is simple: they're the unstable states that need stabilization, and the kinematic model may become unobservable if they are not measured. Even though articulation angle measurement and estimation are outside the scope of this survey, it is worth emphasizing that if a driver aid is expected to make reversing a multi-articulated vehicle as easy as reversing a non-articulated vehicle, or even a vehicle with fewer articulations, then articulation angles are states the aid must know.

Articulation stabilization controllers are relatively simple compared to path planning and path following, and they are a fundamental part of many path planning algorithms. They are expected to be easy to use if the task is only about maintaining the straight pose and adjusting between steady states close to the straight pose with larger turning radii. As shown in Figure 3-2 and Paper C, moving between steady states with a larger turning radius difference requires drivers to have a good understanding of the closed-loop system's behavior and to act while ahead, especially for longer combination vehicles with multiple articulations. The performance of an articulation stabilization controller can be evaluated using the controllable range determined in Paper B. An ideal solution should be able to stabilize a vehicle from any pose within its controllable range to any steady state within the controllable range. The function developed by Mercedes-Benz for passenger vehicles, which stabilizes articulation angles while maintaining or tracking the trailer's heading, might significantly improve the usability; however, it still requires the driver to issue commands to the system at the right time. This is why it's better to develop additional assistance systems that include path-following and path-planning features.

Model-based path-following approaches are seen to have greater potential to adapt to vehicles with different dimensions and numbers of articulations. However, the model-based inverse kinematic method is not preferred for articulated trucks because, as shown in Paper C, its feasible range can be quite limited due to vehicle dimensions and steering limits. Compare model-based single-level and two-level control approaches; the former can be better at controlling the vehicle pose while tracking the reference path, such as reducing the swept area or following references that include both positions and articulation angles. On the other hand, a two-level approach is typically structured with a high-level control that applies path-following algorithms designed for nonarticulated vehicles to the last trailing unit, and a low-level control that realizes the desired motion specified by the high-level control. The low-level control is designed either to focus solely on achieving the desired motion or to achieve it by controlling the vehicle toward steady states; neither is designed to control the vehicle to a non-steady-state pose actively. A point mentioned in the research and strongly related to the practical implementation of a path-following controller is that multiple sets of control parameters are used, enabling the controller to maintain stability over a wide operating range without compromising performance.

A preliminary but important initial step for implementing path-planning algorithms is to consider those scenario-based approaches. They are targeted at some of the most common shunting scenarios for articulated trucks and can be predesigned and reused for those same scenarios. Their results can also help guide the design of new freight terminals to be more accommodating to long combination vehicles. Both lattice-based and sampling-based path planners offer significant potential for path planning in various scenarios. Model-based motion primitives and sampling play a critical role in those planners to ensure the planned path is feasible. Therefore, it is worth continuing the study in Paper D to obtain a quantitative understanding of the mismatch between the model and the real vehicle across different scenarios, thereby providing an opportunity to make the planning less conservative. Further study on the uncontrollable range from Paper A can also contribute to the existing path planning literature. A special set of terminal motion primitives can be designed for the vehicle to drive from controllable to uncontrollable poses, with uncontrollable poses placed at the destination. Those terminal motion primitives can expand the goal area and might facilitate the search for a feasible solution.

For most trucks on the road today, running a path-planning algorithm onboard is impractical. Additionally, very few trucks can accept external steering requests. The existing realistic solution is to handle planning outside the truck and to combine path-following with a human-machine interface to let the driver control the vehicle. Given the rapid development of autonomous driving in passenger vehicles, it is not unreasonable to assume that, in the future, drivers will no longer need to worry about shunting articulated trucks.

Chapter 4
Summary and discussion of included papers

This chapter summarizes the four included papers and discusses connections to previous chapters. The problem of reversing long combination vehicles is proposed to be divided into two operation ranges, as described in Paper A. Among those two ranges, the general favorable range is where jackknifes can be avoided, whose boundary can be defined by the innovative method given in Paper B. Paper C emphasizes the importance of path planning and following functions in solving the reverse challenge of long combination vehicles because the dynamic guideline, a well-established parking support for nonarticulated vehicles, cannot offer the same level of assistance for articulated vehicles. An initial study examines the mismatch between a real tractor-trailer and the most commonly used low-speed vehicle model, the kinematic model, as preparation for similar studies in long combination vehicles and control design in Paper D.

4.1 Paper A

Long combination vehicles reverse strategies based on articulation angle gradient

From drivers experienced in reversing a passenger car with a trailer to researchers working with multi-articulated vehicles, it is well known that specific articulation angle limits must be respected, as exceeding them results in an unavoidable jackknife during continuous reversing. Articulation angle sets that make an articulated vehicle within such a limit are named controllable articulation angles. A common starting point for identifying these limits is the articulation angle set corresponding to the steady-state with the smallest turning radius while remaining free of inter-unit collisions, referred to as the steady-state circling limitation (SSCL) in the paper. The controlled articulated vehicle is then tested for transferring between the two corners in SSCL, which are steady states of maximum left and right turns, in reverse. The steering angle magnitude that produces the steady articulation angle set is reduced until the transfer is successful. Therefore, the resulting articulation angle limit for jackknife avoidance, as determined by this method, is always equal to or tighter than the SSCL. However, based on the visualized articulation angle gradients, articulated vehicles with articulation angles outside SSCLs can still reverse into straight poses, indicating they remain controllable, and SSCLs are conservative. The potential for vehicles outside their controllable range is shown by the distance over which they can be reversed before an inter-unit clash with fixed steering angles.

This paper is inspired by different reverse demands in real operations. Commonly, vehicle poses outside the controllable limitation are considered unwanted poses for reverse, as they pose a risk of inter-unit clashes, and the only way to leave them is to drive forward. For tasks such as reversing along a path or along a narrow loading dock, it is reasonable for the vehicle to remain within the controllable limitation. Most existing research on articulation angle stabilization aims to keep vehicles within their controllable limitations. As discussed before, some nonlinear control methods can offer global asymptotic stability in theory, but achieving it in practice can be difficult or even impossible, especially with dynamic references. A more practical approach is to define the feasible operating range through simulations of a closed-loop system. If the controller is poorly designed, the operating range can be limited, limiting the flexibility of articulations during shunting maneuvers. Some vehicles drive into their uncontrollable range and even jackknife intentionally. For example, when a multi-articulated dump truck empties its first container without decoupling, some articulation angles can approach their mechanical limits. Functions that stabilize articulation angles naturally avoid these uncontrollable poses. This also includes path planners that use information from closed-loop systems with articulation angle stabilization control. Controllers designed for setpoint tracking might be able to bring a vehicle to a desired pose within its uncontrolled range. To complete the dump truck's desired task, the setpoint needs to include not only the position and heading of a single unit but also several or all articulation angles.

4.2 Paper B

Determining the controllable range of reversing multi-articulated vehicles using a geometrical method

An unsolved problem in Paper A is how to define the limit for the controllable range, which is partially solved in Paper B. The controllable range of reversing a multi-articulated vehicle is defined as a set in the articulation angle space, within which the vehicle can reverse into the straight pose while satisfying steering angle and articulation angle constraints. The boundary of the controllable range is the fundamental limitation for the control problem of reversing an articulated vehicle into its straight pose, subject to state and input constraints. No control design can solve the problem if the vehicle's articulation angles are outside the controllable range. In other words, an ideal controller should be able to straighten the vehicle from any pose within the controllable range. Two assumptions are made to simplify the searching problem of determining the controllable range. First, articulated vehicles are assumed to be anti-causal in their articulation angle dynamics. Second, there exists an initial closed controllable range from the straight pose in the articulation angle space. The first assumption shifts the problem from searching for vehicle poses that are possible to reverse into the straight pose to searching for vehicle poses that can be reached from the straight pose by driving forward. Then, the range in the second assumption can be generated by a set of articulation angle paths that start from the straight pose with different fixed steering angles and end at steady states or articulation angle limits. The first assumption is a built-in property in the kinematic model, but not for real vehicles. However, by introducing an LQR-based articulation angle tracking controller, the articulation angle path generated by forward driving can be tracked during reversing, with relatively good performance in a high-fidelity simulation environment. The geometrical method is introduced for vehicles with up to three articulations due to limitations in visualizing articulation angle paths within articulation angle spaces. Based on the geometrical method, the controllable range is iteratively determined by defining the current controllable range's boundary, extending it using new articulation angle paths, and then extending the current controllable range. By examining the geometrical properties of articulation angle paths through visualization, the boundary of an expanded range can be more easily identified. The innovative method shows that articulated vehicles can operate over a broader range while retaining the controllability of reversing into the straight pose, beyond the limits set by existing studies.

The problem of determining a clear boundary between controllable and uncontrollable ranges is considered partially solved for two reasons. First, the geometrical method is challenging to apply in higher dimensions. For triple-articulated vehicles, volume expansion is stopped before the maximum volume is reached because the boundaries lack explicit solutions and rely on simulations. Continuing with the very limited expanded volumes in later stages may not be worthwhile, as they require many simulations, increasing the effort required for expansion. Second, the controllable range determined by a kinematic model may not be directly applied to a real vehicle due to potential model mismatch. As shown by the example of a double-articulated vehicle in Paper B, approximately 89% of the controllable range determined with a kinematic model is still the controllable range within a high-fidelity simulation environment. The question of how to determine an appropriate margin between the kinematic-based controllable range and the controllable range for real operations is one of the reasons for initiating the study in Paper D.

The research literature for controller performance has typically provided feasible operating ranges for their controllers. For path following, their operating ranges are not only determined by the initial articulation angles but also by the initial tracking errors in position and heading. The position and heading as external states are not considered in Paper B. However, the controllable range found by the method described in Paper B is still capable of showing whether an existing controller has the potential to improve performance in maintaining a vehicle within its controllable range, i.e., the ability to reverse continuously without any inter-unit clash. The controllable range is the fundamental limitation of the continuous inter-unit clash-free reversing problem. Suppose a controller can stabilize the articulation angles from any initial pose within the fundamental limitation. In that case, no effort should be spent improving the controller's coverage within the articulation angle space, as this is impossible. A clear boundary for the controllable range can also serve as guidance to drivers or controllers about whether to reverse from the current vehicle pose. Suppose the pose is outside the controllable range. In that case, the vehicle should move forward to reposition itself within the controllable range, unless the desired position can be reached through a limited reverse distance within the uncontrollable range. Most control approaches are designed to stabilize articulation angles towards steady states. These steady states are commonly based on the desired motion of the last trailer, since the planned path lacks information about the vehicle's pose. This may lead to rapid changes in the reference articulation angles, resulting in unstable behavior. For example, when a short lookahead distance is used within pure pursuit. The articulation angle tracking controller in Paper B uses non-steady-state references during transitions between steady states and shows a good tracking performance. For path planners based on closed-loop simulation, besides the planned path in the Euclidean plane, other states have also already appeared in the planning process. Keeping all states and adapting the path-tracking accordingly might help the vehicle transition through non-steady states. Especially in a space-constrained environment, the objective is not only for the last trailer to follow the planned path, but also for the entire vehicle to follow it with specific poses, given how collision checking was handled during planning.

4.3 Paper C

Challenges and opportunities of using rearview dynamic guidelines to assist articulated vehicles in reversing

Nowadays, when drivers shift a modern passenger car into reverse, it is common to see live video from the rearview camera on the car's infotainment screen. This has already been a mandatory safety requirement for some countries. Advanced rearview systems can show a predicted vehicle path based on the current steering angle, which the paper calls the rearview dynamic guideline. Dynamic guidelines are also frequently used on passenger cars equipped with top-view camera systems. The concept behind the rearview dynamic guideline is to show the predicted vehicle motion based on the current driver's input. Many drivers would agree that the dynamic guideline is a helpful function for parking their non-articulated passenger cars. However, by evaluating the dynamics of reversing articulated trucks, it was found that the dynamic guideline cannot provide the same level of assistance as for non-articulated vehicles. For non-articulated vehicles, a one-to-one relationship between steering angle and kinematic motion indicates that the driver can rely on the same set of guidelines for the same vehicle at all positions. This simplifies the planning problem for human drivers. However, this is not the case for articulated vehicles. The kinematic motion of an articulated vehicle depends heavily on all its articulation angles. This makes it very difficult for drivers to predict the vehicle's possible kinematic motion at a future position, because they need to estimate all articulation angles when the vehicle arrives there, which is not easy. Using the same set of guidelines to help drivers in planning is not feasible. Therefore, the planning challenge of reversing articulated vehicles cannot be solved by the rearview dynamic guideline. This doesn't mean the rearview dynamic guideline, or, in a broader sense, the concept of predicting and presenting the path of a vehicle based on the vehicle's state and input, is useless for articulated vehicles. Since articulation angles make it difficult for people to estimate vehicle motions, direct motion prediction can help in multiple ways. Predictive motion can help vehicles avoid inter-unit collisions. With information about the external environment, collisions with external obstacles can be avoided. The predicted motion may also be communicated to other vehicles and workers in the same operating area.

Paper C does not directly focus on any path-following or planning algorithms, but the rearview dynamic guidelines, calculated for nine typical articulated trucks, are suitable for discussion alongside the inverse kinematic approach used in many path-following solutions. The last unit's guidelines for a vehicle are based on given initial vehicle poses, and the leading unit's steering angle remains at different constant angles within its limit. The inverse kinematic approach works inversely, calculating the desired steering angle based on the desired motion of the last unit, which usually acts as a low-level control in two-level path following controllers. Since Paper C shows that the range of the last unit's feasible motion is limited and depends on articulation angles, there is a high chance that a desired motion may not result in a feasible steering angle with inverse kinematics. The non-minimum phase behaviors observed in Paper C make it even harder to handle an infeasible steering angle from the inverse kinematics. To ensure that the desired last unit's motion always yields a feasible steering angle with inverse kinematics, the path planner and the high-level part of the path-following controller need to be designed accordingly. Either the path should be designed so the last trailer can follow it exactly, or the high-level part of the path-following controller should prevent overly rapid dynamics in the desired unit motion.

Another point of Paper C is that to eliminate the challenge of reversing articulated vehicles, path-planning and following solutions are needed. The planning challenge mentioned in Paper C is not only about planning a path to the destination, but also about planning the steering sequence to follow a given path. An example is given in Paper C to show the required steering angle sequences for vehicles with zero to three articulations to reverse through a 90-degree corner. Curves of the last articulation angle with respect to reverse distance show a similar pattern to the steering angle curve of the non-articulated vehicle. To maintain that pattern, the dynamics of steering and articulation angles become increasingly complex as the number of articulations increases. It is not reasonable to expect all drivers to enjoy handling such tasks. The rearview dynamic guideline with an external view, or other solutions that only aim to present current information, will still require drivers to perform the complex planning work for vehicle control.

4.4 Paper D

Loading effects on low-speed motions of a tractor-trailer vehicle

The kinematic model is the most used vehicle model in model-based controllers for low-speed scenarios, such as shunting of articulated vehicles. In the kinematic model, an articulated truck must be simplified into a two-axle leading unit and multiple single-axle trailing units. This means each multi-axle bogie is simplified to a single axle. The equivalent axle positions used in the kinematic model can either be determined only using the vehicle's geometric parameters or based on a force-based approach to find the exact axle position with zero sideslip. Even when the force-based approach is used, it typically only computes the equivalent axle positions for a given operating point and does not update the axle positions used in the kinematic model during operation. This preliminary study investigates how the zero-sideslip axle positions change for a tractor-trailer vehicle under different load conditions during various low-speed maneuvers. Based on real vehicle tests, the offset distance between the actual zero-sideslip position and the corresponding equivalent axle determined by the geometrical method can exceed 20% of the tractor's geometric wheelbase. Differences in load conditions and maneuvers can cause the axle offsets to change by similar magnitudes. This indicates that, across various low-speed conditions, identifying axle positions for the kinematic model using a force-based method at a single operating point provides limited benefit. Controllers designed with kinematic models must account for uncertainties in the model parameters to define their feasible operating range.

Driver feedback is one of the reasons for initiating the study in Paper D. They noticed that when the load condition is changed on an A-double combination, they need to develop different steering strategies while reversing to cope with the change in the vehicle's response. This is a fundamental limitation of the kinematic model, which does not depend on the loading case. When shunting with loading and unloading, the total load and load distribution of a long combination vehicle may change substantially. A mismatch between the kinematic and high-fidelity models is also observed in Paper B. The root cause of the mismatch is that, at the axle positions used in the kinematic model, the actual vehicle no longer has zero sideslip. Besides considering the mismatch as a length variation in Paper D, there are other ways to compensate for it. There are two approaches based on the kinematic model found in the literature. One approach is to expand the kinematic model to include sideslip at each axle as new states. This approach is more popular within off-road applications. Another approach is to introduce articulation angle offsets between the actual measurements and the feedback states, allowing the actual vehicle to stabilize at steady states that cannot be described by the kinematic model. One such application mentioned is reversing on a surface with a side slope, where the articulation angle is nonzero when the entire vehicle is reversed in a straight line. Some of the control algorithms in research studies are inherently robust to certain levels of model mismatch, and validation shows that the controllers still work as expected despite parameter errors in the control model. A few controllers use force-based dynamic models, but they also involve significant simplifications, such as combining multiple tires and ignoring the vertical dimension. The balance between model details and computational complexity, along with the difficulty of obtaining accurate vehicle parameters, makes it challenging to build an accurate dynamic model. Therefore, it is hard to say that a simple dynamic model will consistently outperform the kinematic model. Paper D also observed that the dynamics of the steering and powertrain systems contribute to the mismatch between the vehicle and the kinematic model. These two systems are the systems that finally realize the desired motion. Their effects are also noticed in studies with tests on real vehicles or downscaled models. The general approach in path planning and following is to maintain smooth, low-dynamic steering and velocity requests, but this may also not be easy for heavy vehicles. The shunting challenge involves not just path planning and following, but also all upstream and downstream systems. Improving collaboration across all systems on the chain is an unresolved problem.

Chapter 5
Conclusion and outlook

The general conclusion based on existing research is that some tasks in reversing heavy articulated vehicles can be solved by developed assistance functions. For solvable scenarios, the assistance functions can plan a collision-free path through an environment with obstacles and then control the vehicle to follow it stably. However, there is no clear limit to describing the capabilities of those functions in relation to complex real-world scenarios. Among existing functions, model-based methods play a critical role in solutions that show good adaptability in vehicle and scenario variety. Meanwhile, the present thesis, along with the included papers, identifies several aspects that can lead to improved assistance functions.

The most used vehicle model is the kinematic model, which is the simplest model capable of capturing the dynamics of articulation angles. A reasonable estimate of the length parameters needed for the kinematic model can be obtained using only the vehicle's geometrical dimensions. However, the kinematic model becomes inaccurate when sideslips occur at axle positions, which are influenced by vehicle maneuvers and the external environment. Tests with a tractor-trailer in Paper D show that the zero-sideslip axle positions are influenced by load conditions, driving speed, direction, and steering maneuvers. The influence of these factors is not captured by the kinematic model, leading to performance degradation for many existing controllers. A feasible range is essential for industrializing a driver aid, meaning that, for kinematic model-based driver aids, factors that lead to the kinematic model being inaccurate in vehicle operations need to be understood. Some attempts are made to extend the kinematic model for passenger and agricultural vehicles by adding additional states to account for sideslip effects. Similar studies on multi-articulated trucks during typical maneuvers are yet to be found. In some applications, the kinematic model is replaced with highly simplified dynamic models while maintaining real-time performance. Detailed dynamic models are challenging for real-time applications but can still be useful in offline applications, such as precalculating motion primitives and heuristic functions. Detailed dynamic models and high-fidelity models are preferred for functional validation. Accessing accurate model parameters for those two types of models can be challenging.

The need for driver aids in reversing articulated vehicles is repeatedly mentioned in the literature. As shown in Paper C, it is difficult for drivers to predict the motion of a reversing articulated vehicle, especially a multi-articulated one. Displaying dynamic guidelines for an articulated vehicle can show drivers the vehicle's predicted motion based on its current state, helping them avoid both internal and external collisions. However, the guidelines offer almost no assistance to drivers in stabilizing articulation angles, which is a major challenge they face because of the nature of the articulation angles are unstable while reversing. The guidelines also cannot provide the same level of support for articulated vehicles in path planning and following as they do for non-articulated vehicles, because changing articulation angles affects vehicle motion. Paper C demonstrates the importance of driver aids with control capabilities to eliminate the challenge of reversing articulated vehicles.

The simplest assistant function that involves control is articulation angle stabilization. This function only requires knowledge of articulation angles, which are the vehicle's internal states. It is useful when the vehicle needs to reverse around a straight line or a circular steady state. The controllable range and the method for defining it, as given in Papers A and B, provide a way to evaluate whether an articulation stabilization controller covers all controllable vehicle poses. Controllable poses not covered by an existing controller indicate either that the controller needs tuning or that a new controller is required. Precise parking and turning with articulation angle stabilization requires the driver to have a good understanding of the vehicle's dynamics controlled by this system, so they can issue precise commands ahead to compensate for the unavoidable delay in the closed-loop system.

Path following can solve the drawback of articulation stabilization. With additional knowledge of the reference path and the vehicle's position and heading relative to it, a path-following controller designed for a non-articulated vehicle can be combined with an articulation angle stabilization controller to form a two-level path-following controller for articulated vehicles. An articulation stabilization controller with good coverage, as evaluated using the method described in Papers A and B, can improve the performance of two-level path-following controllers. A two-level control architecture generally achieves good tracking performance at the unit where the high-level controller is located, but the low-level controller cannot track any external reference for the vehicle's pose. The single-level control architecture, typically formulated as an optimization problem considering the entire vehicle, can achieve more complex objectives. Depending on the cost function design, the vehicle can track the position of a single unit, the entire vehicle's pose, or multiple units' positions. For single-level approaches, tuning weights across multiple objectives is challenging; these objectives can also lead to contradictory demands on weights.

The performance of path-following solutions is closely tied to the reference path. That is why model-based path-planning solutions that respect the vehicle's nonholonomic constraints are considered more attractive. Scenario-based planning is limited to known scenarios and requires manual effort to understand and plan for them. This method is worth it for repeated scenarios, such as parking in freight terminals. The planned path can be quite optimal if the scenario is well understood. For more general and novel scenarios, path-planning algorithms such as lattice-based and CL-RRT-based search are needed. The former one is deterministic, and the latter one is non-deterministic. They can systematically search for feasible solutions in a complex planning space where manual planning can be impractical. The lattice-based planning relies on predetermined model-based motion primitives. Due to computational constraints, the primitive set cannot include all feasible vehicle motions, potentially affecting the solvability of the planning problem. Similarly, the CL-RRT relies on a predetermined path-tracking controller, and the planned path is constrained by its performance. As suggested by Paper A, allowing uncontrollable poses as end poses can increase the feasibility of the planning problem by relieving the terminal constraints. The uncontrollable poses will surely cause an inter-unit clash only if the vehicle continues to reverse, which is not a problem with the end pose, as the vehicle can leave the uncontrollable pose by driving forward.

5.1 Outlook

Existing research creates a concrete foundation for future study. It has already been shown that the possibility of fully automating specific tasks in reversing articulated heavy vehicles. To further facilitate the development of an industrialized driver aid from existing research, it is necessary to determine the feasible scenarios for the current research and expand them during development. All approaches have their own limitations, and they need to be known before they are delivered.

One area for future research is the vehicle model, since model-based designs are very attractive and their performance is connected to model quality. The kinematic model is considered a priority choice due to its simplicity. The main direction is to identify under which scenarios the level of presented sideslips will create a model mismatch that cannot be handled by the control parts in current research. There are two potential sub-directions for solving this problem. One direction is to extend the kinematic model to account for sideslip, as well as more complex models that account for forces. However, even with a new model, it is not expected that all real vehicle behaviors will be covered. Extreme scenarios that cause vehicles to deviate significantly from control models need to be identified and either excluded from the assistant aid or handled separately. The other direction is to investigate how the vehicle can be controlled to avoid excessive sideslips, so that it behaves as closely as possible to its kinematic model. Depending on the level of automation, it may also be necessary to consider driver behavior if the driver remains in the loop.

Articulation stabilization solutions will be evaluated with the controllable range developed in this study. A new controllable range may be established in a future study by including additional requirements for maneuvers, such as maneuvering space and heading changes, to account for vehicle states in the global coordinate system. If the whole controllable range is not covered, effort can be spent on expanding the feasible range of control approaches. Otherwise, development should focus on control performance. An improved articulation stabilization solution can further improve the performance of the two-level path-following control based on it. A hybrid controller that combines multiple existing path-following approaches and multiple control parameter sets for the same controller is expected to cover a larger operating range for path-following. In the area of path planning, improvements can be made by allowing uncontrollable poses as the final pose and by implementing improved path-following solutions to facilitate finding solutions.

References

[1] International Transport Forum, “Passenger and freight transport trends compared around the world.” Accessed: Sep. 08, 2025. [Online]. Available: https://www.itf-oecd.org/passenger-freight-transport-trends

[2] Trafikverket, “Vägnät för fordonståg upp till 34,5 meter (Road network for vehicle trains up to 34.5 meters).” Accessed: Jun. 14, 2024. [Online]. Available: https://bransch.trafikverket.se/for-dig-i-branschen/vag/langa-lastbilar-pa-det-svenska-vagnatet/

[3] N. Fröjd, E. Pettersson, and L. Larsson, “Svenska HCT typfordonskombinationer utvärderade mot år 2020 gällande regelverk för BK4 (Swedish HCT typical vehicle combinations evaluation based regulations of year 2020 for BK4),” Apr. 2021. [Online]. Available: https://nvfnorden.org/wp-content/uploads/2021/04/2021-04-15_Svenska_HCT_Typfordon.pdf

[4] Vinnova, “AUTOFREIGHT 2 - Efficient Transport Systems for Regional Container Transports.” Accessed: Jan. 18, 2025. [Online]. Available: https://www.vinnova.se/en/p/autofreight-2---efficient-transport-systems-for-regional-container-transports/

[5] M. F. Mortensen, “EMS2, Trial including double-trailers in Denmark,” Gothenburg, Oct. 2023. [Online]. Available: https://closer.lindholmen.se/sites/default/files/2023-10/5b-denmark-martin-f-m.pdf

[6] A. Vesmes, “Drivers’ needs while reversing with a high capacity transport combination,” Master’s thesis, Chalmers University of Technology, Gothenburg, 2023. [Online]. Available: https://odr.chalmers.se/items/49880d3e-ac72-49ea-ba37-a98cd6c512b6

[7] “Trailer Manoeuvring Assist | GLS SUV March 2025 X167 MBUX | Videos | Owner’s Manual.” Accessed: Sep. 09, 2025. [Online]. Available: https://www.mercedes-benz.co.uk/passengercars/services/manuals.html/gls-suv-2025-03-x167-mbux/videos/trailer-manoeuvring-assist

[8] “Autona/earth.” Accessed: Sep. 16, 2025. [Online]. Available: https://www.volvoautonomoussolutions.com/en-en/autona-earth.html

[9] “How-To Use the BMW Trailer Assistant. - YouTube.” Accessed: Sep. 09, 2025. [Online]. Available: https://www.youtube.com/watch?v=zHgXv-xL8b8

[10] “How to use Volkswagen Trailer Assist on the 2023 Touareg - YouTube.” Accessed: Sep. 09, 2025. [Online]. Available: https://www.youtube.com/watch?v=DeiXrkTQSOY

[11] “Using Pro Trailer Backup AssistTM and Trailer Reverse Guidance | A Ford Towing Video Guide | Ford - YouTube.” Accessed: Sep. 09, 2025. [Online]. Available: https://www.youtube.com/watch?v=fr6GQRBbPsg

[12] D. Zöble, P. David, and W. Philipp, “Steering assistance for backing up articulated vehicles,” *Systemics, Cybernetics and Informatics*, vol. 1, no. 5, pp. 101–106, 2003.

[13] K. Matsushita and T. Murakami, “Backward motion control for articulated vehicles with double trailers considering driver’s input,” in *IECON 2006 - 32nd Annual Conference on IEEE Industrial Electronics*, Paris, France: IEEE, 2006, pp. 3052–3057. doi: 10.1109/IECON.2006.347530.

[14] C. Lundquist, W. Reinelt, and O. Enqvist, “Back driving assistant for passenger cars with trailer,” *SAE Technical Papers*, 2006, doi: 10.4271/2006-01-0940.

[15] J. Stergiopoulos and S. Manesis, “Anti-jackknife state feedback control law for nonholonomic vehicles with trailer sliding mechanism,” *International Journal of Systems, Control and Communications*, vol. 1, no. 3, pp. 297–311, 2009, doi: 10.1504/IJSCC.2009.024557.

[16] J. Il Roh and W. Chung, “Reversing control of a car with a trailer using the driver assistance system,” *Int. J. Adv. Robot. Syst.*, vol. 8, no. 2, pp. 114–121, 2011, doi: 10.5772/10578.

[17] J. Chiu and A. Goswami, “Driver Assist for Backing-Up a Vehicle with a Long-Wheelbase Dual-Axle Trailer,” in *The 11th International Symposium on Advanced Vehicle Control, AVEC12*, Seoul, 2012. [Online]. Available: http://www.ambarish.com/paper/Chiu_Goswami_AVEC12_final.pdf

[18] J. Morales, J. L. Martinez, A. Mandow, and A. J. Garcia-Cerezo, “Steering the last trailer as a virtual tractor for reversing vehicles with passive on-and off-axle hitches,” *IEEE Transactions on Industrial Electronics*, vol. 60, no. 12, pp. 5729–5736, 2013, doi: 10.1109/TIE.2013.2240631.

[19] D. S. Rhode *et al.*, “Trailer path curvature control for trailer backup assist,” US 8,909,426 B2, Dec. 09, 2014

[20] V. Josef, “Trailer parking assistant: Upgrade for current parking assistant control units,” in *Proceedings of the 16th International Conference on Mechatronics - Mechatronika 2014*, Brno, Czech Republic: IEEE, Dec. 2014, pp. 677–682. doi: 10.1109/MECHATRONIKA.2014.7018342.

[21] J. Cheng, B. Wang, Y. Zhang, and Z. Wang, “Backward orientation tracking control of mobile robot with N trailers,” *Int. J. Control Autom. Syst.*, vol. 15, no. 2, pp. 867–874, 2017, doi: 10.1007/s12555-015-0382-7.

[22] M. Hejase, J. Jing, J. M. Maroli, Y. Bin Salamah, L. Fiorentini, and Ü. Özgüner, “Constrained Backward Path Tracking Control using a Plug-in Jackknife Prevention System for Autonomous Tractor-Trailers,” in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, Maui, HI, USA: IEEE, Nov. 2018, pp. 2012–2017. doi: 10.1109/ITSC.2018.8569262.

[23] M. Beglini, L. Lanari, and G. Oriolo, “Anti-Jackknifing Control of Tractor-Trailer Vehicles via Intrinsically Stable MPC,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France: IEEE, May 2020, pp. 8806–8812. doi: 10.1109/ICRA40945.2020.9197012.

[24] J. Roempke and C. Liu, “Trailer backup assist using steer-by-wire,” Master’s thesis, KTH Royal Institute of Technology, Stockholm, 2024.

[25] C. Liu, J. Roempke, M. Klomp, and L. Drugge, “Trailer reversing supported by steer-by-wire,” in *16th International Symposium on Advanced Vehicle Control. AVEC 2024. Lecture Notes in Mechanical Engineering*, G. Mastinu, F. Braghin, F. Cheli, M. Corno, and S. M. Savaresi, Eds., Springer, Oct. 2024, pp. 585–596. doi: 10.1007/978-3-031-70392-8_83.

[26] N. A. Ashok *et al.*, “Development of test environments for reverse assist functions as applied to an A-double vehicle combination,” in *16th International Symposium on Heavy Vehicle Transportation Technology (HVTT16)*, Sep. 2021. [Online]. Available: https://hvttforum.org/wp-content/uploads/2021/10/Ashok-Development-Of-Test-Environments-For-Reverse-Assist-Functions-As-Applied-To-An-A-Double-Vehicle-Combination.pdf

[27] I. Kageyama and Y. Saito, “On the backward maneuverability of articulated vehicles.,” *Vehicle System Dynamics,* vol. 15, no. sup1, pp. 222–232, 1986, doi: 10.1080/00423118608969138.

[28] U. Larsson, C. Zell, K. Hyyppa, and A. Wernersson, “Navigating an articulated vehicle and reversing with a trailer,” in *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, San Diego, CA, USA: IEEE, May 1994, pp. 2398–2404. doi: 10.1109/robot.1994.351152.

[29] R. Parra-Loera and D. J. Corelis, “Expert system controller for backing-up a truck-trailer system in a constrained space,” in *Proceedings of 1994 37th Midwest Symposium on Circuits and Systems*, Lafayette, LA, USA: IEEE, Aug. 1994, pp. 1357–1361. doi: 10.1109/mwscas.1994.519060.

[30] M. Sampei, T. Tamura, T. Kobayashi, and N. Shibui, “Arbitrary Path Tracking Control of Articulated Vehicles Using Nonlinear Control Theory,” *IEEE Transactions on Control Systems Technology*, vol. 3, no. 1, pp. 125–131, Mar. 1995, doi: 10.1109/87.370718.

[31] S. Sekhavat, F. Lamiraux, J. P. Laumond, G. Bauzil, and A. Ferrand, “Motion planning and control for Hilare pulling a trailer: Experimental issues,” in *Proceedings of International Conference on Robotics and Automation*, Albuquerque, NM, USA: IEEE, 1997, pp. 3306–3311. doi: 10.1109/robot.1997.606793.

[32] P. Bolzern, R. M. DeSantis, A. Locatelli, and D. Masciocchi, “Path-tracking for articulated vehicles with off-axle hitching,” *IEEE Transactions on Control Systems Technology*, vol. 6, no. 4, pp. 515–523, 1998, doi: 10.1109/87.701346.

[33] R. M. DeSantis, “Path-tracking for articulated vehicles via exact and Jacobian linearization,” *IFAC Proceedings Volumes*, vol. 31, no. 3, pp. 159–164, 1998, doi: 10.1016/s1474-6670(17)44078-x.

[34] P. Bolzern, R. M. DeSantis, and A. Locatelli, “An input-output linearization approach to the control of an n-body articulated vehicle,” *Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME*, vol. 123, no. 3, pp. 309–316, Jan. 2001, doi: 10.1115/1.1387010.

[35] D.-H. Kim and J.-H. Oh, “Globally asymptotically stable tracking control for a trailer system,” *J. Robot. Syst.*, vol. 19, no. 5, pp. 199–205, 2002, doi: 10.1002/rob.10034.

[36] A. Astolfi, P. Bolzern, and A. Locatelli, “Path-tracking of a tractor-trailer vehicle along rectilinear and circular paths: A Lyapunov-based approach,” *IEEE Transactions on Robotics and Automation*, vol. 20, no. 1, pp. 154–160, 2004, doi: 10.1109/TRA.2003.820928.

[37] M. Park, W. Chung, and M. Kim, “Experimental research of a passive multiple trailer system for backward motion control,” in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, Barcelona, Spain: IEEE, Apr. 2005, pp. 105–110. doi: 10.1109/ROBOT.2005.1570104.

[38] C. Pradalier and K. Usher, “Experiments in autonomous reversing of a tractor-trailer system,” in *Field and Service Robotics. Springer Tracts in Advanced Robotics*, vol. 42, C. Laugier and R. Siegwart, Eds., Berlin, Heidelberg: Springer, 2008, pp. 475–484. doi: 10.1007/978-3-540-75404-6_45.

[39] A. González-Cantos and A. Ollero, “Backing-up maneuvers of autonomous tractor-trailer vehicles using the qualitative theory of nonlinear dynamical systems,” *Int. J. Rob. Res.*, vol. 28, no. 1, pp. 49–65, 2009, doi: 10.1177/0278364908097588.

[40] J. Morales, J. L. Martínez, A. Mandow, and I. J. Medina, “Virtual steering limitations for reversing an articulated vehicle with off-axle passive trailers,” in *2009 35th Annual Conference of IEEE Industrial Electronics*, Porto, Portugal: IEEE, Nov. 2009, pp. 2385–2390. doi: 10.1109/IECON.2009.5415436.

[41] Z. Leng and M. Minor, “A simple tractor-trailer backing control law for path following,” in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, Oct. 2010, pp. 5538–5542. doi: 10.1109/IROS.2010.5650489.

[42] Z. Leng and M. A. Minor, “A simple tractor-trailer backing control law for path following with side-slope compensation,” in *2011 IEEE International Conference on Robotics and Automation*, Shanghai, China: IEEE, May 2011, pp. 2386–2391. doi: 10.1109/ICRA.2011.5979918.

[43] Y. Bin, T. Shim, N. Feng, and D. Zhou, “Path tracking control for backing-up tractor-trailer system via model predictive control,” in *2012 24th Chinese Control and Decision Conference (CCDC)*, Taiyuan, China: IEEE, May 2012, pp. 198–203. doi: 10.1109/CCDC.2012.6242930.

[44] M. Michaek, “Application of the VFO method to set-point control for the N-trailer vehicle with off-axle hitching,” *Int. J. Control*, vol. 85, no. 5, pp. 502–521, 2012, doi: 10.1080/00207179.2012.658524.

[45] M. Michalek and M. Kielczewski, “Helping a driver in backward docking with N-trailer vehicles by the passive control-assistance system,” in *16th International IEEE Conference on Intelligent*

Transportation Systems (ITSC 2013), The Hague, Netherlands: IEEE, Oct. 2013, pp. 1993–1999. doi: 10.1109/ITSC.2013.6728522.

[46] M. M. Michałek, “A highly scalable path-following controller for N-trailers with off-axle hitching,” *Control Eng. Pract.*, vol. 29, pp. 61–73, 2014, doi: 10.1016/j.conengprac.2014.04.001.

[47] C. Pradalier and K. Usher, “Robust Trajectory Tracking for a Reversing Tractor Trailer,” *J. Field Robot.*, vol. 25, no. 6–7, pp. 378–399, 2008, doi: 10.1002/rob.20241.

[48] M. Werling, P. Reinisch, M. Heidingsfeld, and K. Gresser, “Reversing the general one-trailer system: Asymptotic curvature stabilization and path tracking,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 2, pp. 627–636, 2014, doi: 10.1109/TITS.2013.2285602.

[49] M. Michalek and K. Kozlowski, “VFO tracking and set-point control for an articulated vehicle with off-axle hitched trailer,” in *2009 European Control Conference (ECC)*, Budapest, Hungary: IEEE, Aug. 2009, pp. 266–271. doi: 10.23919/ecc.2009.7074415.

[50] O. Enqvist, “AFS-Assisted Trailer Reversing,” Master’s thesis, Linköping University, Linköping, 2006. [Online]. Available: http://www.vehicular.isy.liu.se/Publications/MSc/09_EX_4227_JL.pdf

[51] A. Elhassan, “Autonomous driving system for reversing an articulated vehicle,” Master’s thesis, KTH Royal Institute of Technology, 2015.

[52] M. M. Michalek and M. Kielczewski, “The concept of passive control assistance for docking maneuvers with n-trailer vehicles,” *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 5, pp. 2075–2084, 2015, doi: 10.1109/TMECH.2014.2362354.

[53] A. J. Rimmer and D. Cebon, “Theory and practice of reversing control on multiply-articulated vehicles,” *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 230, no. 7, pp. 899–913, 2016, doi: 10.1177/0954407015596918.

[54] M. Sampei, T. Tamura, T. Itoh, and M. Nakamichi, “Path Tracking Control of Trailer-Like Mobile Robot,” in *Proceedings IROS ‘91:IEEE/RSJ International Workshop on Intelligent Robots and Systems ‘91*, Osaka, Japan: IEEE, Nov. 1991, p. 128. doi: 10.1109/IROS.1991.174448.

[55] K. Kural, I. Besselink, Y. Xu, A. Tomar, and H. Nijmeijer, “A driver support system for improved maneuvering of articulated vehicles using an unmanned aerial vehicle,” in *14th International Symposium on Heavy Vehicle Transportation Technology (HVTT14)*, 2016. [Online]. Available: https://hvttforum.org/wp-content/uploads/2019/11/Kural-A-driver-support-system-for-improved-maneuvering-of-articulated-vehicles-using-an-unmanned-aerial-vehicle.pdf

[56] T. Wu and J. Y. Hung, “Path following for a tractor-trailer system using model predictive control,” in *SoutheastCon 2017*, Concord, NC, USA: IEEE, Mar. 2017, pp. 1–5. doi: 10.1109/SECON.2017.7925337.

[57] Z. Leng and M. A. Minor, “Curvature-based ground vehicle control of trailer path following considering sideslip and limited steering actuation,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 2, pp. 332–348, Feb. 2017, doi: 10.1109/TITS.2016.2572208.

[58] L. Buning *et al.*, “INTRALOG - Intelligent Autonomous Truck Applications in Logistics; Autonomous rearward docking maneuvering on distribution centers,” in *13th ITS European Congress*, Strasbourg, Jun. 2017.

[59] R. Kusumakar, “Autonomous Parking for Articulated Vehicles,” Master’s thesis, HAN University of Applied Sciences, 2017. doi: 10.13140/RG.2.2.30533.76009.

[60] X. Liu and D. Cebon, “A Minimum Swept Path Control Strategy for Reversing Articulated Vehicles,” in *2018 IEEE Intelligent Vehicles Symposium (IV)*, Changshu, China: IEEE, Jun. 2018, pp. 1962–1967. doi: 10.1109/IVS.2018.8500417.

[61] J. McDowell, “Comparison of Modern Controls and Reinforcement Learning for Robust Control of Autonomously Backing Up Tractor-Trailers to Loading Docks,” Master’s thesis, California Polytechnic State University, 2019.

[62] M. M. Michałek, “Agile maneuvering with intelligent articulated vehicles: A control perspective,” *IFAC-PapersOnLine*, vol. 52, no. 8, pp. 458–473, 2019, doi: 10.1016/j.ifacol.2019.08.092.

[63] X. Liu, “Space-constrained autonomous reversing of articulated vehicles,” PhD thesis, University of Cambridge, 2020. doi: 10.17863/CAM.64466.

[64] Z. Li, H. Cheng, J. Ma, and H. Zhou, “Research on parking control of semi-trailer truck,” in *2020 4th CAA International Conference on Vehicular Control and Intelligence, CVCI 2020*, Hangzhou, China, Dec. 2020, pp. 424–429. doi: 10.1109/CVCI51460.2020.9338617.

[65] M. Kumar, A. C. Hildebrandt, P. Strauss, S. Kraus, C. Stiller, and A. Zimmermann, “Lateral Trajectory Stabilization of an Articulated Truck during Reverse Driving Maneuvers,” in *2020 IEEE Intelligent Vehicles Symposium (IV)*, Las Vegas, NV, USA, Oct. 2020, pp. 744–751. doi: 10.1109/IV47402.2020.9304691.

[66] P. Ribeiro *et al.*, “A VR truck docking simulator platform for developing personalized driver assistance,” *Applied Sciences*, vol. 11, no. 19, 2021, doi: 10.3390/app11198911.

[67] A. C. Manav, I. Lazoglu, and E. Aydemir, “Adaptive path-following control for autonomous semi-trailer docking,” *IEEE Trans. Veh. Technol.*, vol. 71, no. 1, pp. 69–85, Jan. 2022, doi: 10.1109/TVT.2021.3125131.

[68] E. P. Biever, “A nonlinear path tracking approach for reversing an A-double truck,” Master’s thesis, Eindhoven University of Technology, 2023. [Online]. Available: https://research.tue.nl/en/studentTheses/a-nonlinear-path-tracking-approach-for-reversing-an-a-double-truc

[69] J. Kim and S. Yim, “Automated Parking System for Tractor-Trailer Vehicle,” in *16th International Symposium on Advanced Vehicle Control. AVEC 2024. Lecture Notes in Mechanical Engineering*, Springer, 2024, pp. 171–177. doi: 10.1007/978-3-031-70392-8_25.

[70] S. Rahman, T. Gordon, L. Henderson, Y. Gao, S. Coleman, and D. Kerr, “Model-Free Automated Reversing of Articulated Heavy Goods Vehicles,” in *16th International Symposium on Advanced Vehicle Control. AVEC 2024. Lecture Notes in Mechanical Engineering*, G. Mastinu, F. Braghin, F. Cheli, M. Corno, and S. M. Savaresi, Eds., Springer, 2024, pp. 116–122. doi: 10.1007/978-3-031-70392-8_17.

[71] V. V. Gosar, M. Alirezaei, I. J. M. Besselink, and H. Nijmeijer, “Performance-aware control design for reverse docking of tractor semi-trailer combinations,” in *Advances in Dynamics of Vehicles on Roads and Tracks III. IAVSD 2023. Lecture Notes in Mechanical Engineering*, W. Huang and M. Ahmadian, Eds., Springer, 2024, pp. 739–748. doi: 10.1007/978-3-031-66968-2_73.

[72] C. Pradalier and K. Usher, “A simple and efficient control scheme to reverse a tractor-trailer system on a trajectory,” in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, IEEE, Apr. 2007, pp. 2208–2214. doi: 10.1109/ROBOT.2007.363648.

[73] M. Park, W. Chung, M. Kim, and J. Song, “Control of a mobile robot with passive multiple trailers,” in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA ‘04. 2004*, New Orleans, LA, USA: IEEE, Apr. 2004, pp. 4367–4374. doi: 10.1109/ROBOT.2004.1302405.

[74] K. Kural, “Analysis of High Capacity Vehicles for Europe: application of Performance Based Standards and improving Manoeuvrability,” PhD thesis, Technische Universiteit Eindhoven, Eindhoven, 2019.

[75] M. Michałek, “Geometrically motivated set-point control strategy for the standard N-trailer vehicle,” in *2011 IEEE Intelligent Vehicles Symposium (IV)*, Baden-Baden, Germany: IEEE, Jun. 2011, pp. 138–143. doi: 10.1109/IVS.2011.5940436.

[76] P. Rouchon, M. Fliess, J. Lévine, and P. Martin, “Flatness and motion planning: the car with n trailers,” in *Proc. ECC’93*, Groningen, 1993, pp. 1–6. [Online]. Available: http://cas.ensmp.fr/~rouchon/publications/PR1993/ECC93.pdf

[77] F. Bullo and R. M. Murray, "Experimental comparison of trajectory trackers for a car with trailers," *IFAC Proceedings Volumes*, vol. 29, no. 1, pp. 2804–2809, 1996, doi: 10.1016/s1474-6670(17)58101-x.

[78] C. Altafini, A. Speranzon, and B. Wahlberg, "A feedback control scheme for reversing a truck and trailer vehicle," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 6, pp. 915–922, Dec. 2001, doi: 10.1109/70.976025.

[79] J. Yuan and Y. Huang, "Path Following Control for Tractor-Trailer Mobile Robots with Two Kinds of Connection Structures," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing, China: IEEE, Oct. 2006, pp. 2533–2538. doi: 10.1109/IROS.2006.281702.

[80] C.-J. Hoel and P. Falcone, "Low speed maneuvering assistance for long vehicle combinations," in *2013 IEEE Intelligent Vehicles Symposium (IV)*, Gold Coast, QLD, Australia: IEEE, Jun. 2013, pp. 598–604. doi: 10.1109/IVS.2013.6629532.

[81] O. Ljungqvist, "Motion Planning and Stabilization for a Reversing Truck and Trailer System," Master's thesis, Linköping University, 2015.

[82] N. Evestedt, O. Ljungqvist, and D. Axehill, "Path tracking and stabilization for a reversing general 2-trailer configuration using a cascaded control approach," in *2016 IEEE Intelligent Vehicles Symposium (IV)*, Gothenburg, Sweden: IEEE, Jun. 2016, pp. 1156–1161. doi: 10.1109/IVS.2016.7535535.

[83] O. Ljungqvist, D. Axehill, and A. Helmersson, "Path following control for a reversing general 2-trailer system," in *2016 IEEE 55th Conference on Decision and Control (CDC)*, Las Vegas, NV, USA: IEEE, Dec. 2016, pp. 2455–2461. doi: 10.1109/CDC.2016.7798630.

[84] A. J. Rimmer and D. Cebon, "Implementation of reversing control on a doubly articulated vehicle," *Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME*, vol. 139, no. 6, 2017, doi: 10.1115/1.4035456.

[85] A. S. Tomar, K. Kural, R. Kusumakar, and A. Naren, "Path Following Bi-Directional Controller for Articulated Vehicles," in *15th International Symposium on Heavy Vehicle Transportation Technology (HVTT15)*, 2018.

[86] O. Ljungqvist, N. Evestedt, D. Axehill, M. Cirillo, and H. Pettersson, "A path planning and path-following control framework for a general 2-trailer with a car-like tractor," *J. Field Robot.*, vol. 36, no. 8, pp. 1345–1377, 2019, doi: 10.1002/rob.21908.

[87] K. Tanaka, T. Taniguchi, and H. O. Wang, "Trajectory control of an articulated vehicle with triple trailers," in *Proceedings of the 1999 IEEE International Conference on Control Applications (Cat. No.99CH36328)*, Kohala Coast, HI, USA, Aug. 1999, pp. 1673–1678. doi: 10.1109/cca.1999.801223.

[88] A. W. Divelbiss and J. T. Wen, "A path space approach to nonholonomic motion planning in the presence of obstacles," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 3, pp. 443–451, Jun. 1997, doi: 10.1109/70.585905.

[89] S. Sekhavat, P. Švestka, J. P. Laumond, and M. H. Overmars, "Multilevel path planning for nonholonomic robots using semiholonomic subsystems," *Int. J. Rob. Res.*, vol. 17, no. 8, pp. 840–857, 1998, doi: 10.1177/027836499801700803.

[90] F. Cuesta, F. Gómez-Bravo, and A. Ollero, "Parking maneuvers of industrial-like electrical vehicles with and without trailer," *IEEE Transactions on Industrial Electronics*, vol. 51, no. 2, pp. 257–269, 2004, doi: 10.1109/TIE.2004.824855.

[91] T. Nayl, G. Nikolakopoulos, and T. Gustafsson, "On-Line path planning for an articulated vehicle based on Model Predictive Control," in *2013 IEEE International Conference on Control Applications (CCA)*, Hyderabad, India: IEEE, Aug. 2013, pp. 772–777. doi: 10.1109/CCA.2013.6662843.

[92] F. Ghilardelli, G. Lini, and A. Piazzi, "Path generation using η4-Splines for a truck and trailer vehicle," *IEEE Transactions on Automation Science and Engineering*, vol. 11, no. 1, pp. 187–203, Jan. 2014, doi: 10.1109/TASE.2013.2266962.

[93] A. J. Rimmer and D. Cebon, “Planning collision-free trajectories for reversing multiply-articulated vehicles,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 7, pp. 1998–2007, Jul. 2016, doi: 10.1109/TITS.2015.2511880.

[94] K. Kural *et al.*, “VISTA: Vision supported truck docking assistant,” in *16th International Symposium on Heavy Vehicle Transportation Technology (HVTT16)*, 2021.

[95] M. Bos, B. Vandewal, W. Decré, and J. Swevers, “MPC-based Motion Planning for Autonomous Truck-Trailer Maneuvering,” *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 4877–4882, Jul. 2023, doi: 10.1016/j.ifacol.2023.10.1258.

[96] R. Stahn, T. Stark, and A. Stopp, “Laser scanner-based navigation and motion planning for truck-trailer combinations,” in *2007 IEEE/ASME international conference on advanced intelligent mechatronics*, Zurich, Switzerland: IEEE, Sep. 2007. doi: 10.1109/AIM.2007.4412422.

[97] N. Evestedt, O. Ljungqvist, and D. Axehill, “Motion planning for a reversing general 2-trailer configuration using Closed-Loop RRT,” in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Daejeon, Korea (South): IEEE, Oct. 2016, pp. 3690–3697. doi: 10.1109/IROS.2016.7759544.

[98] A. Mohamed, J. Ren, H. Lang, and M. El-Gindy, “Optimal collision free path planning for an autonomous articulated vehicle with two trailers,” in *2017 IEEE International Conference on Industrial Technology (ICIT)*, Toronto, ON, Canada: IEEE, Mar. 2017, pp. 860–865. doi: 10.1109/ICIT.2017.7915472.

[99] X. Wang, J. Taghia, and J. Katupitiya, “Robust Model Predictive Control for Path Tracking of a Tracked Vehicle with a Steerable Trailer in the Presence of Slip,” *IFAC-PapersOnLine*, vol. 49, no. 16, pp. 469–474, 2016, doi: 10.1016/j.ifacol.2016.10.085.

[100] R. Lenain, B. Thuilot, C. Cariou, and P. Martinet, “Sideslip angles observer for vehicle guidance in sliding conditions: Application to agricultural path tracking tasks,” in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, Orlando, FL, USA: IEEE, May 2006, pp. 3183–3188. doi: 10.1109/ROBOT.2006.1642186.

[101] J. Chiu and A. Goswami, “The critical hitch angle for jackknife avoidance during slow backing up of vehicle-trailer systems,” *Vehicle System Dynamics*, vol. 52, no. 7, pp. 992–1015, 2014, doi: 10.1080/00423114.2014.909944.

[102] K. Kural, A. Prati, I. Besselink, J. Pauwelussen, and H. Nijmeijer, “Validation of Longer and Heavier Vehicle Combination Simulation Models,” *SAE Int. J. Commer. Veh.*, vol. 6, no. 2, pp. 340–352, 2013, doi: 10.4271/2013-01-2369.

[103] T. Ghandriz, B. Jacobson, P. Nilsson, L. Laine, and N. Fröjd, “Computationally efficient nonlinear one- And two-track models for multitrailer road vehicles,” *IEEE Access*, vol. 8, pp. 203854–203875, 2020, doi: 10.1109/ACCESS.2020.3037035.

[104] Inc. The MathWorks, “ode15i - Solve fully implicit differential equations — variable order method - MATLAB.” Accessed: Sep. 23, 2025. [Online]. Available: https://se.mathworks.com/help/matlab/ref/ode15i.html

[105] E. Kuiper and J. J. M. Van Oosten, “The PAC2002 advanced handling tire model,” *Vehicle System Dynamics*, vol. 45, no. sup1, pp. 153–167, 2007, doi: 10.1080/00423110701773893.

[106] N. Fröjd, “Handling Analysis and Control Development of Commercial Trucks with Volvo Transport Models - MATLAB & Simulink.” Accessed: Jun. 15, 2025. [Online]. Available: https://se.mathworks.com/videos/handling-analysis-and-control-development-of-commercial-trucks-with-volvo-transport-models-1622035211192.html

[107] I. J. M. Besselink, M. H. M. Baart, and H. Nijmeijer, “Simplified Turn Slip Modeling by a Parallel Magic Formula Model,” in *Advances in Dynamics of Vehicles on Roads and Tracks II. IAVSD 2021. Lecture Notes in Mechanical Engineering*, A. Orlova and D. Cole, Eds., Springer, Aug. 2022, pp. 966–973. doi: 10.1007/978-3-031-07305-2_89.

[108] “TruckMaker | IPG Automotive.” Accessed: Aug. 26, 2025. [Online]. Available: https://www.ipg-automotive.com/en/products-solutions/software/truckmaker/

[109] R. S. Sharp, D. Casanova, and P. Symonds, “A mathematical model for driver steering control, with design, tuning and performance results,” *Vehicle System Dynamics*, vol. 33, no. 5, pp. 289–326, 2000, doi: 10.1076/0042-3114(200005)33:5;1-q;ft289.

[110] A. Gonzblez-Cantos, J. I. Maza, and A. Ollero, “Design of a Stable Backing Up Fuzzy Control of Autonomous Articulated Vehicles for Factory Automation,” in *ETFA 2001. 8th International Conference on Emerging Technologies and Factory Automation. Proceedings (Cat. No.01TH8597)*, Antibes-Juan les Pins, France: IEEE, Oct. 2001, pp. 447–451. doi: 10.1109/ETFA.2001.996401.

[111] M. Hosein and A. Mohammadion, “Trajectory Tracking Wheeled Mobile Robot Using Backstepping Method with Connection off Axle Trailer,” *International Journal of Smart Electrical Engineering*, vol. 7, no. 4, pp. 177–187, 2018.

[112] A. Morro, A. Sgorbissa, and R. Zaccaria, “Path following for unicycle robots with an arbitrary path curvature,” *IEEE Transactions on Robotics*, vol. 27, no. 5, pp. 1016–1023, Oct. 2011, doi: 10.1109/TRO.2011.2148250.

[113] “The Fastest Maze-Solving Competition On Earth - YouTube.” Accessed: Nov. 11, 2025. [Online]. Available: https://www.youtube.com/watch?v=ZMQbHMgK2rw

[114] R. A. Knepper and A. Kelly, “High Performance State Lattice Planning Using Heuristic Look-Up Tables,” in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing, China: IEEE, Oct. 2006, pp. 3375–3380. doi: 10.1109/IROS.2006.282515.

[115] S. M. LaValle, “Rapidly exploring random trees: A new tool for path planning,” Ames, Jun. 1998. [Online]. Available: https://lavalle.pl/papers/Lav98c.pdf

[116] J. Barraquand and J. C. Latombe, “Robot Motion Planning: A Distributed Representation Approach,” *Int. J. Rob. Res.*, vol. 10, no. 6, pp. 628–649, 1991, doi: 10.1177/027836499101000604.

[117] L. E. Kavralu, P. Svestka, J.-C. Latombe, and M. H. Overmars, “Probabilistic Roadmaps for Path Planning in High-Dimensional Configuration Spaces,” *IEEE Transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566–580, Aug. 1996, doi: 10.1109/70.508439.

List of licentiate theses from Chalmers Vehicle Dynamics group

Year	**Author**	**Thesis title**
2007	Matthijs Klomp	On Drive Force Distribution and Road Vehicle Handling - a Study of Understeer and Lateral Grip
2008	Sogol Kharrazi	On Lateral Stability of Heavy Vehicle Combinations - Linking Accident Analysis and Vehicle Dynamic
2011	Derong Yang	Post Impact Vehicle Path Control in Multiple Event Accident
2014	Kristoffer Tagesson	Truck Steering System and Driver Interaction
2015	Peter Nilsson	On Traffic Situation Predictions for Automated Driving of Long Vehicle Combinations
2015	Adithya Arikere	Vehicle Dynamic Opportunities in Electrified Vehicles for Active Safety Interventions
2015	Anton Albinsson	Online State Estimation in Electrified Vehicles Linked to Vehicle Dynamics
2016	Artem Kusachov	Motion perception and tire models for winter conditions in driving simulators
2017	Pär Pettersson	On numerical descriptions of road transport missions
2018	Ulrich Sander	Towards Crash-Free Driving - Opportunities and Limitations of Automated Emergency Braking in Intersections
2018	Toheed Ghandriz	Transportation mission based optimization of heavy vehicle fleets including propulsion tailoring
2019	Tushar Chugh	Haptic feedback control methods for steering systems
2019	Weitao Chen	Co-simulation in virtual verification of vehicles with mechatronic systems
2020	Juliette Torinsson	Power loss minimization in electric cars by wheel force allocation
2021	Luigi Romano	Mathematical modelling of operating cycles for road vehicles
2022	Yansong Huang	Target-driven road vehicle suspension design
2023	Sachin Janardhanan	On power loss minimisation for heavy vehicles with axle-wise and modular electrical propulsion and friction braking
2024	Umur Erdinc	Safe distributed control allocation for articulated heavy vehicles
2026	Samira Deylaghian	Modelling and Analysis of Longitudinal Vehicle Dynamics Near Standstill with Brake Friction
2026	Carl Emvin	Road vehicle energy demand predictions under uncertain operating conditions