Accepted in ICRA 2019

Trajectory Planning for a Tractor with Multiple Trailers in Extremely Narrow Environments: A Unified Approach *

Bai Li, Youmin Zhang, Tankut Acarman, Qi Kong, and Yue Zhang

Abstract—**Trajectory planning for a tractor-trailer vehicle is challenging because the vehicle kinematics consists of underactuated and nonholonomic constraints that are highly coupled. Prevalent sampling-based or search-based planners suitable for rigid-body vehicles are not capable of handling the tractor-trailer vehicle cases. This work aims to deal with generic *n*-trailer cases in the tiny environments. To this end, an optimal control problem is formulated, which is beneficial in being accurate, straightforward, and unified. An adaptively homotopic warm-starting approach is proposed to facilitate the numerical solution process of the formulated optimal control problem. Compared with the existing sequential warm starting strategies, our proposal can adaptively define the subproblems with the purpose of making the gaps between adjacent subproblems "pleasant" for the solver. Unification and efficiency of the proposed adaptively homotopic warm-starting approach have been investigated in several extremely tiny scenarios. Our planner finds solutions that other existing planners cannot. Online planning opportunities are briefly discussed as well.**

I. Introduction

A. Background and Motivations

A tractor-trailer vehicle refers to a towing tractor hooked up with one or more unpowered trailers [1]. Commonly the trailer wheels are non-steerable, and the steering forces originate from the pivoting joints, which sequentially connect adjacent parts of the entire vehicle [2]. Compared with the same-length rigid-body vehicle, a tractor-trailer vehicle travels more agilely in narrow/cluttered environments, or uneven terrain, thus has been widely applied in many intricate scenarios [3].

Tractor-trailer vehicle trajectory planning refers to generating satisfactory trajectories for the tractor and trailers from their initial configurations to the terminal ones. Herein, the satisfaction requests solution feasibility (e.g. no violation of vehicle kinematics or collision-free constraints) and optimality (i.e. the planned trajectories are expected to be optimal). Trajectory planning for a tractor-trailer vehicle is challenging because i) the underactuated and nonholonomic constraints couple in the planning model [4], thus making most of the existing car-like planners not directly applicable; and ii) the dynamic system is proved to be unstable in the backward motions. This paper focuses on the trajectory planning scheme for the tractor-trailer vehicles.

*Research supported by Natural Sciences and Engineering Research Council of Canada, and the National Natural Science Foundation of China under Grants 61573282 and 61833013. The third author gratefully acknowledges the support of Galatasaray University, scientific research support program under Grant # 19.401.003.

Bai Li is with the R&D Center of Automated Driving, JD Inc., Beijing, China (e-mail: libai@zju.edu.cn).

Youmin Zhang is with the Department of Mechanical, Industrial and Aerospace Engineering, Concordia Institute of Aerospace Design and Innovation, Concordia University, Montreal, Canada (phone: 514-848-2424 ext. 5225; fax: 514-848-3175; e-mail: ymzhang@encs.concordia.ca).

Tankut Acarman is with the Computer Engineering Department, Galatasaray University, Istanbul, Turkey (e-mail: tacarman@gsu.edu.tr).

Qi Kong is with the R&D Center of Automated Driving, JD Inc., Beijing, China (e-mail: Qi.Kong@jd.com).

Yue Zhang is with the Division of Systems Engineering, and Center for Information and Systems Engineering, Boston University, Boston, MA, USA (e-mail: joycez@bu.edu).

B. Related Works

Prevalent trajectory planners in this community are classified as graph-search-based, sampling-based, and optimal-control-based methods.

A graph-search planner first abstracts the continuous configuration space as nodes in a graph, and then searches for feasible links among the nodes such that the vehicle is led towards the destination. Dijkstra algorithm [5,6], A* algorithm [7–9], and dynamic programming [10] have been considered as the searcher.

In contrast to a graph-search-based planner that discretizes the configuration space, a sampling-based planner uses specific state patterns to explore the continuous space. State-lattice planner [11,12] and closed-loop Rapidly exploring Random Tree (CL-RRT) [13] respectively represent the deterministic and stochastic methods in this category.

Regardless of doing space discretization or not, the graph-search-based and sampling-based planners commonly have to avoid directly handling the entire continuous configuration space. Such an idea may be efficient in dealing with a rigid-body vehicle, but involves more challenges regarding a multi-body vehicle, because the configurations/states of the entire system are highly coupled. A clear evidence is that, quite few graph-searchers or samplers have been proposed for tractor-trailer vehicles, compared to the large numbers of publications for the rigid-body vehicles. In fact, among the existing studies mentioned above, few of them can handle the generic *n*-trailer cases in the constrained environments.

Nominally, the concerned trajectory planning task should be formulated as an optimal control problem, in which the cost function, vehicle kinematics, and collision-avoidance restrictions are clearly and objectively described. Analytical solutions are derived from the **Pontryagin's maximum** principle [14,15], whereas numerical solutions are obtained by nonlinear programming (NLP) related optimizers [1,12,16], and initialization strategies [17]. However, they do not work in the cases with complicated obstacles.

Besides the aforementioned three categories, early studies include the rule-based planners which decouple the original scheme into multiple steps such as first adjusting the heading, and then travelling in a straight line [18]. Such planners are not sufficiently intelligent, and are still inapplicable in the irregular or tiny scenarios.

To summarize, there has not been a unified local trajectory planner that can deal with generic *n*-trailer cases in the tiny environments with acceptable computational efforts.

C. Contribution

This work proposes a local trajectory planning method, which can handle generic *n*-trailer cases in the tiny/irregular environments. To this end, an optimal control problem is formulated to describe this scheme, which would be solved numerically. The core contribution lies in the proposal of a sequential warm-starting computational framework which facilitates the numerical solution process.

The rest of this paper is structured as follows. Section II gives the optimal control problem statement. Section III briefly introduces the basic numerical computation principles. An adaptively homotopic warm-starting approach is proposed in Section IV for solution facilitation. Simulation results, discussions, and conclusions are provided in Section V.

II. Tractor-Trailer Vehicle Trajectory Planning Problem Formulation

The tractor-trailer vehicle trajectory planning problem is formulated as an optimal control problem, which is about minimizing the task completion time and maximizing the obstacle clearance, subject to the kinematic constraints, collision-avoidance constraints, and boundary conditions.

A. Vehicle Kinematics

A tractor-trailer vehicle is described as a tractor towing $(n-1)$ unpowered trailers. Since the vehicle usually moves not fast in the tiny environments, the tire-sideslip effect is slim, thus the bicycle model is suitable to model the tractor kinematics [1]:

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} x_1(t) \\ y_1(t) \\ v_1(t) \\ \phi_1(t) \\ \theta_1(t) \end{bmatrix} = \begin{bmatrix} v_1(t)\cdot\cos\theta_1(t) \\ v_1(t)\cdot\sin\theta_1(t) \\ a_1(t) \\ \omega_1(t) \\ v_1(t)\cdot\tan\phi_1(t)/\mathrm{L_{W1}} \end{bmatrix},\ t\in[0,t_\mathrm{f}]. \tag{1}$$

In (1), t_f stands for the unknown terminal time, (x_1, y_1) refers to the mid-point of rear wheel axis (point P_1 in Fig. 1), θ_1 refers to the orientation angle, v_1 refers to the velocity of P_1, a_1 refers to the corresponding acceleration, ϕ_1 refers to the steering angle of the front wheels, ω_1 refers to the corresponding angular velocity, and $\mathrm{L_{W1}}$ denotes the wheelbase length. In addition, $\mathrm{L_{N1}}$ stands for the front overhang length, $\mathrm{L_{M1}}$ stands for the rear overhang length, and $\mathrm{L_{B1}}$ stands for the tractor width.

The kinematic principle of each trailer is determined according to

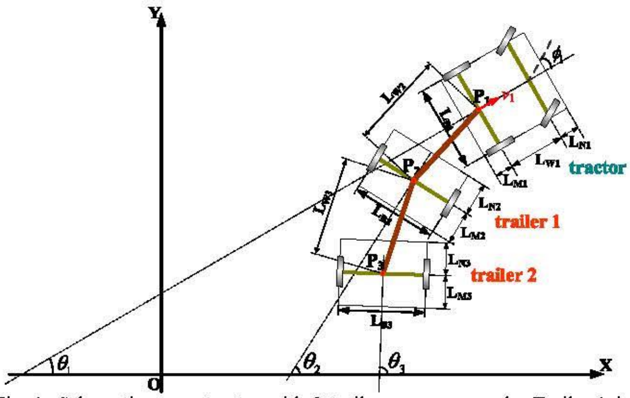

Fig. 1. Schematics on a tractor with 2 trailers as an example. Trailer 1 is hooked up at the middle point of the tractor's rear wheels, and trailer 2 is hooked up at that of trailer 1.

$$x_i(t) = x_1(t) - \sum_{j=2}^{i}\left(\mathrm{L}_{\mathrm{W}j}\cdot\cos\theta_j(t)\right),\ i\geq 2, \tag{2a}$$

$$y_i(t) = y_1(t) - \sum_{j=2}^{i}\left(\mathrm{L}_{\mathrm{W}j}\cdot\sin\theta_j(t)\right),\ i\geq 2, \tag{2b}$$

$$\frac{\mathrm{d}\theta_i(t)}{\mathrm{d}t} = \begin{cases} v_1(t)/\mathrm{L_{W2}}\cdot\sin\left(\theta_1(t)-\theta_2(t)\right),\ i=2 \\ v_1(t)/\mathrm{L}_{\mathrm{W}i}\cdot\sin\left(\theta_{i-1}(t)-\theta_i(t)\right)\cdot\left(\prod_{k=1}^{i-2}\cos\left(\theta_k(t)-\theta_{k+1}(t)\right)\right),\ i\geq 3 \end{cases}. \tag{2c}$$

Herein, (x_i, y_i) refers to the mid-point along the wheel axis of the *i*th part of the tractor-trailer vehicle (i.e. trailer $i-1$). θ_i denotes the orientation angle of trailer $i-1$. P_i is the hooking point that connects the *i*th and $(i-1)$**th** parts of the whole system. Each two adjacent hooking points P_i and P_{i-1} are connected by a rigid link with the length of $\mathrm{L}_{\mathrm{W}i}$ ($i\geq 2$). Definitions of $\mathrm{L}_{\mathrm{B}i}$, $\mathrm{L}_{\mathrm{M}i}$, and $\mathrm{L}_{\mathrm{N}i}$ are depicted in Fig. 1. Boundaries are imposed on the state profiles of the tractor:

$$|a_1(t)| \leq \mathrm{a_{max}}, \tag{3a}$$

$$|v_1(t)| \leq \mathrm{v_{max}}, \tag{3b}$$

$$|\phi_1(t)| \leq \Phi_{\max}, \tag{3c}$$

$$|\omega_1(t)| \leq \Omega_{\max}. \tag{3d}$$

Also, the orientation error between each two adjacent parts of the tractor-trailer vehicle is bounded to avoid involving the jackknife effect [19]:

$$|\theta_i(t)-\theta_{i-1}(t)| \leq \Theta\mathrm{err_{max}},\ i\geq 2. \tag{3e}$$

B. Collision Avoidance Constraints

Suppose that each tractor/trailer is rectangular, and that each obstacle in the environment is presented as a convex polygon (a concave polygon can be decoupled as multiple convex polygons). Collision between each tractor/trailer and each polygonal obstacle should be avoided at every moment during $t\in[0,t_\mathrm{f}]$. Let us denote the k**th** vertex of the *j*th polygonal obstacle as V_{jk} ($k=1,2,\ldots,\mathrm{Npol}_j$), and denote the

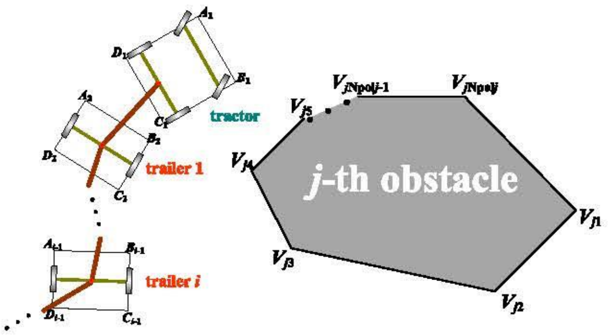

Fig. 2. Schematics on vertexes of rectangular tractor/trailer and polygonal obstacles.

four vertexes of the ith part of the vehicle as A_i, B_i, C_i, and D_i (Fig. 2).

The collision-avoidance constraint between the ith part of the vehicle and the jth polygonal obstacle is identical to the conditions that i) A_i, B_i, C_i, and D_i locate out of the jth polygon; and ii) each $\mathbf{V}_{jk}$ ($k = 1,2,\dots,\mathrm{Npol}_j$) locates out of the rectangle $A_iB_iC_iD_i$. The restriction that one point Q locates out of one convex polygon $\mathbf{W}_1\mathbf{W}_2\dots\mathbf{W}_m$ is formulated as an analytical inequality via the "triangle-area-based criterion" proposed in [20]:

$$S_{\Delta QW_mW_1} + \sum_{i=1}^{m-1} S_{\Delta QW_iW_{i+1}} > S_{\square W_1W_2\dots W_m}, \tag{4}$$

where S_Δ denotes the triangle area, and $S_\square$ denotes the polygon area. According to (4), the aforementioned conditions i) and ii) can be described as

$$S_{\Delta QV_{j\mathrm{Npol}_j}V_1} + \sum_{m=1}^{\mathrm{Npol}_j-1} S_{\Delta QV_{jm}V_{j(m+1)}} > S_{\square V_{j1}V_{j2}\dots V_{j\mathrm{Npol}_j}},$$
$$\forall Q \in \{A_i, B_i, C_i, D_i\}. \tag{5a}$$

$$S_{\Delta QA_iB_i} + S_{\Delta QB_iC_i} + S_{\Delta QC_iD_i} + S_{\Delta QD_iA_i} > S_{\square A_iB_iC_iD_i},$$
$$\forall Q \in \{V_{j1}, V_{j2}, \dots, V_{j\mathrm{Npol}_k}\}. \tag{5b}$$

Applying (5) between each tractor/trailer and each obstacle throughout $[0, t_f]$ would formulate the collision-avoidance constraints in the complete form.

C. Boundary Conditions

At the starting moment $t = 0$, the configurations of the entire vehicle (including $x_1(0)$, $y_1(0)$, $v_1(0)$, $\phi_1(0)$, $\theta_1(0)$, $a_1(0)$, $\omega_1(0)$, $\theta_2(0)$, etc.) are given. At the terminal moment t_f, the tractor-trailer vehicle is expected to stop stablely at a desired spot. Herein, the stability means

$$[v_1(t_f), \phi_1(t_f), a_1(t_f), \omega_1(t_f)] = [0,0,0,0]. \tag{6}$$

Without (6), the vehicle may not really stop when $t > t_f$.

D. Cost Function

We expect the vehicle to complete the travel with minimum time, and keep maximum clearance with the obstacles during the whole process. Thus the cost function J is defined as

$$J = t_f - \lambda_w \cdot \int_{\tau=0}^{t_f} \left(\sum_i \sum_k e^{-d_{ik}^2(\tau)} \right), \tag{7}$$

wherein $d_{ik}(\tau)$ stands for the Euclidean distance between (x_i, y_i) and the geometric center of the kth obstacle at time τ, and $\lambda_w > 0$ is the weighting parameter.

E. Overall Formulation

As a summary, the entire optimal control problem is formulated as

$$\begin{aligned} &\min \ (7), \\ &\text{s.t. Kinematic constraints (1)} - \text{(3)}; \\ &\quad \text{Collision-avoidance constraints (5)}; \\ &\quad \text{Initial and terminal conditions, and (6)}. \end{aligned} \tag{8}$$

The unknowns in (8) include $v_1(t)$, $a_1(t)$, $\omega_1(t)$, $\phi_1(t)$, t_f, $x_i(t)$, $y_i(t)$, and $\theta_i(t)$ ($i = 1,\dots,n$).

III. Numerical Solutions to Optimal Control Problems

Analytical solutions to (8) are generally not available because of the complicated collision-avoidance constraints (5). Instead, numerical solutions are expected. Commonly, solving (8) numerically involves two steps: i) discretizing all the state profiles to build an NLP problem; and ii) solving the converted NLP problem. In this work, we choose the first-order explicit Runge-Kutta method to do the discretization and the Interior Point Method (IPM) [21] for NLP solutions.

IV. NLP Solution Facilitation Method

A. Motivations

Simply applying the methodology in Section III does not result in optimal solutions to the generic cases, because the complicated collision-avoidance constraints (5) are difficult to handle by IPM or other gradient-based optimizers directly.

A commonly way to facilitate the computational burden is to find a near-optimal or even near-feasible initial guess, from which the NLP-solving process warmly starts [17]. Following this idea, we can define a sequence of subproblems such that the ith NLP is always slightly more difficult than the $(i-1)$th one, and the last subproblem is the *original problem*, i.e. the one stems from discretizing (8). Since the ith subproblem is slightly more difficult than the $(i-1)$th one, the optimum of the $(i-1)$th subproblem is a near-feasible initial guess for subproblem i. If we find the optimum of subproblem 1, then the sequential process will result in the optimal solution to the original problem after finite cycles. The aforementioned sequential process inherently aims to avoid handling the entire difficulties *all at once*. Instead, the entire difficulties are dispersed among the subproblems, and then tackled *progressively*. Ideally, the dispersion should guarantee that the

Accepted in ICRA 2019

V. Simulation Results and Discussions

Simulations were performed in an MATLAB+AMPL platform [22], and executed on an i5-7200U CPU with 8 GB RAM that runs at 2.50 × 2 GHz. Common parameters regarding the vehicle and Algorithm 1 are given in Table I. A video with the primary simulations results is provided at https://youtu.be/Zj8-az5mxpw. Source codes are available at https://github.com/libai1943/Tractor-trailer-trajectory-planning-case-studies-ICRA-19.

TABLE I. Parametric Settings regarding Model and Approach.

Parameter	Description	Setting
L_{N1}	Tractor front overhang length	0.25 m
L_{Ni}	Trailer front overhang length $(i \geq 2)$	1.00 m
L_{M1}	Tractor rear overhang length	0.25 m
L_{Mi}	Trailer rear overhang length $(i \geq 2)$	1.00 m
L_{Bi}	Tractor/trailer width $(i \geq 1)$	1.00 m
L_{W1}	Tractor wheelbase	1.50 m
L_{Wi}	Distance between hooking points on two adjacent bodies $(i \geq 2)$	3.00 m
a_{max}	Bound of $\lvert a_1(t) \rvert$	0.25 m/s²
v_{max}	Bound of $\lvert v_1(t) \rvert$	2.0 m/s
Φ_{max}	Bound of $\lvert \phi_1(t) \rvert$	0.7 rad
Ω_{max}	Bound of $\lvert \omega_1(t) \rvert$	0.5 rad/s
Θerr_{max}	Bound of $\lvert \theta_i(t) - \theta_{i-1}(t) \rvert$, $(i \geq 2)$	1.57 rad
step	Increment of γ	0.2
α_{reduce}	Factor to reduce *step* when a failure occurs	0.5
N_{expand}	Threshold of successive cycles with success to enlarge *step*	10
ε_{exit}	Threshold regarding *step* value to exit the sequential computation process	10^{-5}
ε_0	Setting of γ for subproblem 0	0.05
λ_w	Weighting parameter in (7)	1.0

A. On the Efficiency of Algorithm 1

Three cases have been tested, and the optimized trajectories are depicted in Figs. 4–6. The scenario in Case 1 contains a bottleneck, which is formed by two irregularly placed obstacles. In Fig. 4, none part of the whole system collides with the obstacles, which indicates that our formulated (5) is efficacious. Case 2 involves a cluttered environment. Viewing the optimized solution of Case 2 (see Fig. 7), one may find that the trajectory of the tractor is more complicated than that of each trailer, and the trajectory of trailer i is more complicated than that of trailer i+1. Rationale behind this phenomenon has been analyzed in [1]. This phenomenon is also reflected in Case 3, which represents a typical garage-parking scheme. Before the end of this subsection, let us take Case 1 as an example to investigate the sequential computation process of Algorithm 1. As shown in Fig. 8, 40 subproblems are solved before the original NLP problem gets solved finally. During the sequential process, whenever $\gamma_{achieved}$ stops increasing, a failure occurs, then *step* reduces. On the other hand, *step* still has chances to increase (see its temporary growth during cycles 20–25), which means the N_{expand} related criterion takes effect. As a summary, Algorithm 1 is efficient to handle various cases.

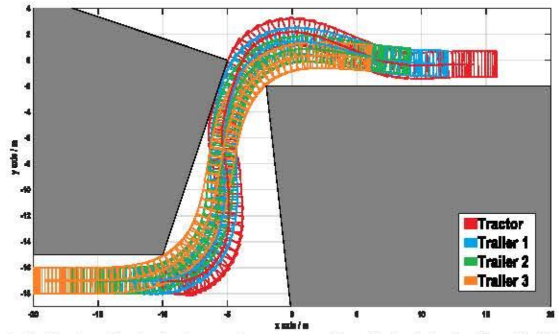

Fig. 4. Optimized trajectories and corresponding footprints in Case 1 (CPU time is 334.10 sec, and $t_f = 28.57$).

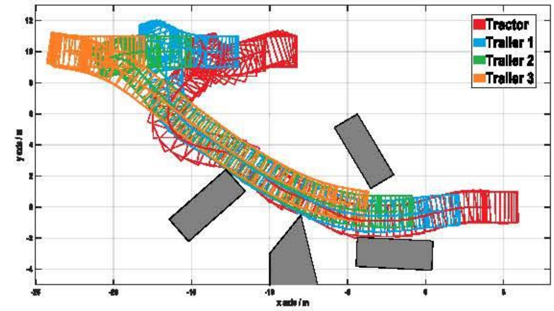

Fig. 5. Optimized trajectories and corresponding footprints in Case 2 (CPU time is 114.24 sec, and $t_f = 32.07$).

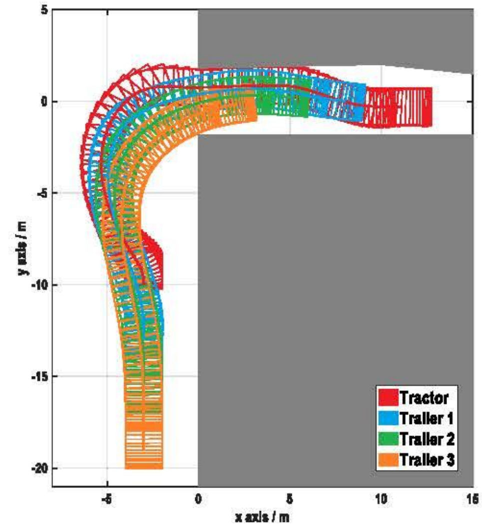

Fig. 6. Optimized trajectories and corresponding footprints in Case 3 (CPU time is 237.91 sec, and $t_f = 21.68$).

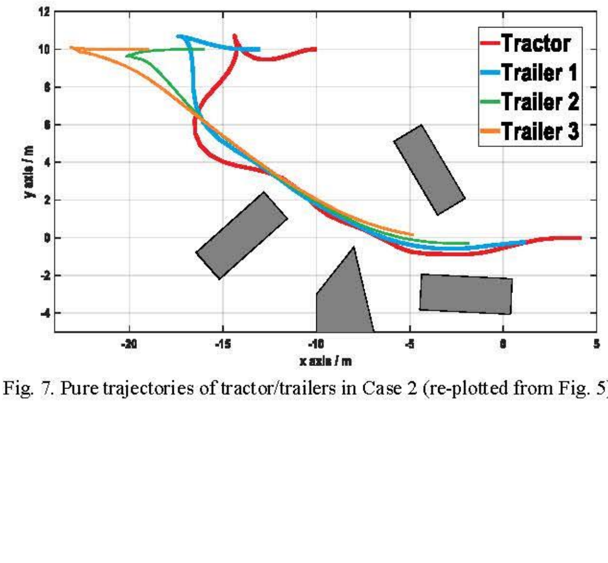

Fig. 7. Pure trajectories of tractor/trailers in Case 2 (re-plotted from Fig. 5).

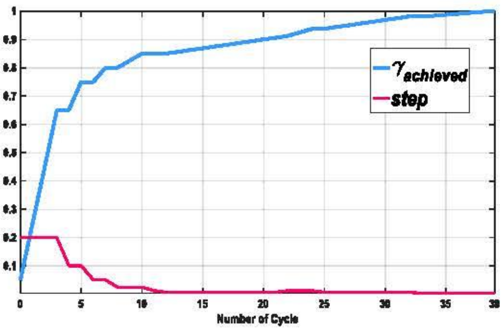

Fig. 8. Evolutions of *step* and $\gamma_{achieved}$ in applying Algorithm 1 (Case 1).

B. Comparisons with Other Planners

Comparisons have been made among Algorithm 1 and the methodologies in [1], [13], [17] and [23]. Ref. [1] represents the basic numerical optimization without initialization. Through simply adopting a linear initialization, the solver in [1] converged to infeasibility quite fast for each of three cases. This means initialization is useful. Ref. [23] is an easier version of Algorithm 1, which fixes *step* to a specified value. If *step* is set large, gaps between adjacent subproblems would be overly big, thus failures are inevitable; if *step* is set small, CPU time is wasted compared to our *step* enlargement strategy in Algorithm 1. To conclude, the capability of [23] never outperforms Algorithm 1, theoretically. Ref. [17] builds a similar sequential and adaptive computation framework according to the evolution of cost function value. That strategy does not work here because the cost function value may not be a good criterion for difficulty division. Ref. [13] represents the sampling-based methods. In a generic *n*-trailer system, the degree of freedom is far less than the dimension of state profiles, which makes it difficult to link the sampled nodes with kinematical feasibility. We believe that an incremental search- or sampling-based planner would gradually become inefficient for a tractor with more trailers.

C. How to Derive Online Solutions?

Online trajectories can be obtained through a nonlinear model predictive control (NMPC) method, wherein the rolling optimization is done with the help of warm starting. Taking Case 3 as an example, we find that the average CPU time for solving the infinite-horizon open-loop optimal control problem is 1.29 s in each receding horizon. Simplifications on (8) would further accelerate the computation. One may prepare offline typical trajectories *a priori* and utilize them as reference solutions in the online NMPC framework.

REFERENCES

[1] B. Li, and Z. Shao, "Time-optimal trajectory planning for tractor-trailer vehicles via simultaneous dynamic optimization," In *Proc. 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3844–3849, 2015.

[2] T. Nayl, G. Nikolakopoulos, T. Gustafsson, D. Kominiak, and R. Nyberg, "Design and experimental evaluation of a novel sliding mode controller for an articulated vehicle," *Robotics and Autonomous Systems*, vol. 103, pp. 213–221, 2018.

[3] A. González-Cantos, and A. Ollero, "Backing-up maneuvers of autonomous tractor-trailer vehicles using the qualitative theory of nonlinear dynamical systems," *International Journal of Robotics Research*, vol. 28, no. 1, pp. 49–65, 2009.

[4] M. Yue, X. Hou, R. Gao, and J. Chen, "Trajectory tracking control for tractor-trailer vehicles: A coordinated control approach," *Nonlinear Dynamics*, vol. 91, no. 2, pp. 1061–1074, 2018.

[5] M. Viale, T. Tsubochi, and S. I. Yuta, "A practical path and motion planner for a tractor-trailer robot," In *Proc. 1997 IEEE/RSJ International Conference on Intelligent Robot and Systems (IROS)*, pp. 989–996, 1997.

[6] 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, 2016.

[7] S. Beyersdorfer, and S. Wagner, S, "Novel model based path planning for multi-axle steered heavy load vehicles," In *Proc. 16th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 424–429, 2013.

[8] Y. Xu, and R. Liu, "Path planning for mobile articulated robots based on the improved A* algorithm," *International Journal of Advanced Robotic Systems*, vol. 14, no. 4, pp. 1–10, 2017.

[9] M. Cirillo, "From videogames to autonomous trucks: A new algorithm for lattice-based motion planning," In *Proc. 2017 IEEE Intelligent Vehicles Symposium (IV)*, pp. 148–153, 2017.

[10] P. Ferbach, "A method of progressive constraints for nonholonomic motion planning," *IEEE Transactions on Robotics and Automation*, vol. 14, no.1, pp. 172–179, 1998.

[11] O. Ljungqvist, N. Evestedt, M. Cirillo, D. Axehill, and O. Holmer, "Lattice-based motion planning for a general 2-trailer system," In *Proc. 2017 IEEE Intelligent Vehicles Symposium (IV)*, pp. 819–824, 2017.

[12] J. W. Choi, and K. Huhtala, "Constrained global path optimization for articulated steering vehicles," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 4, pp. 1868–1879, 2016.

[13] N. Evestedt, O. Ljungqvist, and D. Axehill, "Motion planning for a reversing general 2-trailer configuration using closed-loop RRT," In *Proc. 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3690–3697, 2016.

[14] A. Mohamed, J. Ren, H. Lang, and M. El-Gindy, "Optimal collision free path planning for an autonomous articulated vehicle with two trailers," In *Proc. 2017 IEEE International Conference on Industrial Technology (ICIT)*, pp. 860–865, 2017.

[15] M. Chyba, and S. Sekhavat, "Time optimal paths for a mobile robot with one trailer," In *Proc. 1999 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1669–1674, 1999.

[16] J. Liu, W. Han, H. Peng, and X. Wang. "Trajectory planning and tracking control for towed carrier aircraft system," *Aerospace Science and Technology*, vol. 84, pp. 830–838, 2019.

[17] B. Li, and Z. Shao, "An incremental strategy for tractor-trailer vehicle global trajectory optimization in the presence of obstacles," In *Proc. 2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 1447–1452, 2015.

[18] F. Cuesta, G.-B. Fernando, and O. Aníbal. "Parking maneuvers of industrial-like electrical vehicles with and without trailer," *IEEE Transactions on Industrial Electronics*, vol. 51, no. 2, pp. 257–269, 2004.

[19] D. Zobel, E. Balcerak, and T. Weidenfeller, "Minimum parking maneuvers for articulated vehicles with one-axle trailers," In *Proc. 9th International Conference on Control, Automation, Robotics, and Vision (ICARCV)*, pp. 1–6, 2006.

[20] B. Li, and Z. Shao, "A unified motion planning method for parking an autonomous vehicle in the presence of irregularly placed obstacles," *Knowledge-Based Systems*, vol. 86, pp. 11–20, 2015.

[21] A. Wächter, and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, 2006.

[22] R. Fourer, D. M. Gay, and B. W. Kernighan. *AMPL: A Modeling Language for Mathematical Programming*. San Francisco, CA, USA: Scientific, 2003.

[23] K. Bergman, and D. Axehill, "Combining homotopy methods and numerical optimal control to solve motion planning problems," In *Proc. 2018 IEEE Intelligent Vehicles Symposium (IV)*, pp. 347–354, 2018.