2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Online Trajectory Deformation and Tracking for Self-entanglement-free Differential-Driven Robots

Jiangpin Liu, Tong Yang, *Member, IEEE*, Wangtao Lu,
Yue Wang, *Member, IEEE*, and Rong Xiong, *Member, IEEE*

***Abstract*—This paper introduces an optimisation-based trajectory deformation and tracking algorithm for tethered differential-driven mobile robots. The motivation of this work is to generate self-entanglement-free (SEF) commands for a tethered differential-driven robot to track a path. Whilst existing path planners have been capable of generating SEF paths for tethered differential-driven robots lacking an omni-directional tether retracting mechanism, no trajectory planner can handle the unavoidable movement errors that cause robot pose deviate from the pre-defined path. The trajectory deformation and tracking is challenging because the admissible heading direction of the robot is highly constrained by the SEF constraint. As a result, even with an SEF path, the robot still encounters self-entanglement issues during execution.**

This paper fills this gap by formulating the trajectory deforming and tracking (TDT) problem of a tethered robot into a multi-objective optimisation framework. Explicit consideration of the constraint of the relative angle between the tether stretching direction and the robot's heading direction to be admissible during its movement is provided in this framework. The proposed algorithm repeatedly deforms the pre-defined path for easier tracking, whilst generating a suitable velocity profile for robot execution. Compared to directly applying the commonly used untethered trajectory deformation and tracking algorithm into tethered cases, the proposed algorithm demonstrates improved performance in terms of minimising the risk of self-entanglement and maximising robot safety. These are validated in both simulated and real scenarios. An open-sourcesourcing implementation has also been provided for the benefit of the robotics community.

I. Introduction

Trajectory deforming and tracking (TDT) is a crucial module for mobile robot execution. Given the pre-calculated path to be tracked, the movement errors unavoidably occur when the robot is moving, causing the robot to deviate from the intended path. In the past decades, several kinds of algorithms have been proposed to address or alleviate this issue. One kind of algorithms adopted feedback control strategy that generates robot commands based on the current position and orientation error with respect to the selected waypoint [1]. Another category of algorithms employed sampling-based technique to find the most suitable steering direction and velocity within a small spatial-temporal window [2]. The fundamental aim of all these algorithms is motivating the robot to return to the pre-defined paths. More recently, it has been observed that when the robot deviates from the path and the environmental conditions are changing, the path pre-defined in the past might not be suitable for the current scene [3]. Therefore, concentrations have been shifted towards not only selecting the best robot commands, but also continuously deforming the trajectory to better match the robot execution [4]. A notable breakthrough in this area is the observation of the modelling of the path deforming and tracking task to a multi-objective optimisation problem [5] [6] [7].

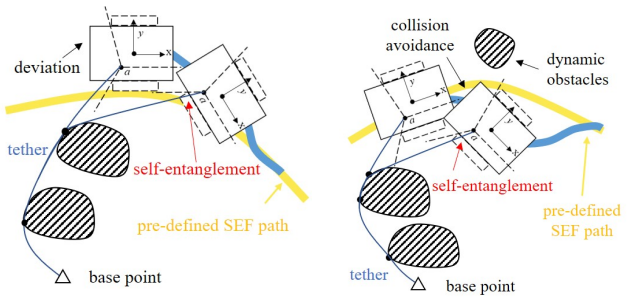


Fig. 1. Illustration of the motivation of the proposed algorithm. In the scenarios shown in this figure, the situations are observed where the robot, whilst trying to return to its pre-defined path or avoid dynamic obstacles, encounter self-entanglement issues. The tethered path deforming and tracking task has be a challenging topic because of the combinational efforts of the robot kinematics, the collision avoidance, the maximum length constraint of the robot tether, and the admissible relative angle between the robot's heading direction and the tether retracting direction. Whilst algorithms have been developed to generate self-entanglement-free (SEF) paths for these robots, the problem arises when the robot deviates from its pre-defined path due to unavoidable movement errors. At this point, there has not been an algorithm capable of guiding the robot towards the goal whilst complying with tether-related constraints.

Compared to the TDT algorithm designed for common untethered robots, the TDT task for a tethered robot is slightly different in the following aspects. The most apparent difference is in collision avoidance. Different from untethered mobile robots that can actively mobilise themselves to avoid collision with dynamic obstacles, the cable of a tethered robot is entirely passive. When a dynamic obstacle intentionally crosses the robot tether, there is no effective action that the robot can take to avoid it. Therefore, an implicit environmental requirement for a tethered TDT task is that dynamic obstacles do not come into contact the robot tether. Another difference lies in the singularity of the homotopy class within which the path can deform. Before a tethered TDT algorithm is executed, a preceding tethered path planning algorithm has explicitly

This work was supported by the National Key RD Program of China (Grant No. 2022YFB4701502). *(Corresponding author: Yue Wang and Rong Xiong.)*

Jiangpin Liu, Tong Yang, Wangtao Lu, Yue Wang, and Rong Xiong are with the State Key Laboratory of Industrial Control and Technology, Zhejiang University, P.R. China.

taken into consideration the maximum tether length constraint and selected a suitable topological route. Therefore, if the tethered trajectory is not continuously deformed but is altered to a trajectory in a different homotopy class, then there is no guarantee for the tether length admissibility. In other words, prior efforts on finding non-homotopic paths such as [8] are not applicable in the tethered TDT scenarios. Moreover, the tether length must be continuously monitored during the robot motion, which was definitely ignored when using untethered TDT algorithms to tethered scenarios. Unfortunately, most tethered robot literature just focused on the path planning phase. Even if real-world experiments were conducted, the path following was established using untethered TDT algorithms such as the naive feedback control [1] and the timed-elastic-band [5] (TEB) algorithm.

This paper considers a further challenging version of the tethered TDT problem: the tethered TDT problem with self-entanglement-free (SEF) constraint. This is motivated by the practical issue that the mobile unit adopted in a tethered robot system is usually a differential-driven one, chosen for its mobility in uneven environment, and is also conditioned by the fact that an omni-directional tether retracting mechanism is often absent on the robot. Early work [9] has introduced the SEF constraint and generated SEF tethered robot paths. However, to the best of the authors' knowledge, no existing path tracking algorithm can be applied to execute robot movement complying with the SEF constraint. Even if the path is initially designed to be SEF, it will inadvertently violate the SEF constraint after untethered TDT algorithms modify it due to robot execution error and the presence of dynamic obstacles. See Fig. 1 for illustration. The proposed algorithm bridges this gap by explicitly considering the relative angle between the robot's heading direction and the tether retracting direction, and minimising the risk for self-entanglement. The contributions of this paper are listed as follows:

1) The formulation of self-entanglement-free (SEF) trajectory deformation and tracking problem into a weighted multi-objective optimisation problem.
2) The first trajectory deformation and tracking algorithm with minimised SEF property for the safe execution of a tethered differential-driven mobile robot.
3) The open-sourcing [1] of C++ implementation within the framework of Robot Operating System (ROS).

The remainder of this paper is organised as follows: Section II reviews existing literature. Section III formulates the online trajectory deformation problem designed for tethered differential-driven robots without an omni-directional tether retracting mechanism. Section IV presents the proposed algorithm. Simulated and real-world experiments are collected in Section V, with final concluding remarks gathered in Section VI.

II. Related works

The applications of tethered mobile robots have gained increasing attentions in recent years. Classic environmental challenges such as providing stable communication and energy supply for robots become trivial issues in the tethered scenarios because of the connection between the mobile unit and the base station given by the robot tether. This makes the tethered robots suitable for operating under various extreme conditions, such as pipe inspection [10], highway maintenance [11], grass trimming [12] [13] [14], disaster recovery [15], mountain climbing tasks [16] [17], and exploration tasks [18].

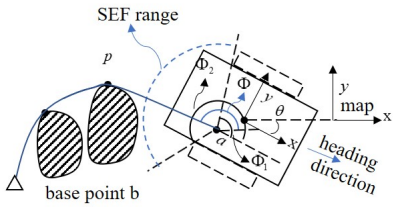


Fig. 2. Illustration of the definitions and notations of the robot model.

Planning tether-length-admissible path for a tethered robot has been the most popular topic in the past decades. This has been extensively investigated, starting with particle robots in polygonal environments [19] [20], then extending to particle robots in grid-based environments [21], and finally to circular robots in grid-based environments [22] [23]. Also, applicable scenarios have been generalised from planar environments to underwater environments [24], aerial environments [25], two linked mobile robots [26], an aerial robot linked to a mobile robot [27], tethered multi-goal visiting and travelling salesman problem [28], and cases with higher-order topological complexities [29]. Recently, [9] observed the self-entanglement issue when the mobile unit is differential-driven and lacks an omni-directional tether retracting structure, and proposed a searching-based planner for self-entanglement-free tethered path planning (SEFTPP). And [30] further proved that the validation process of the self-entanglement-free property of tethered waypoint configurations can be sparse, which significantly reduced the computational time for obtaining a SEF resultant path.

In contrast to the numerous path planning algorithms for tethered robots, little concentration has been paid to developing techniques for deforming and tracking trajectories for tethered robots that may encounter self-entanglement issues. To be precise, most of the existing research on tethered robots, including [19] [16] [31] [17] [22] [20] [23] [13] [14] [28], have only focused on simulations. [24] utilised the naive feedback control strategy for tethered underwater vehicles. Because the movement error rarely causes self-entanglement, [9] also deployed the feedback control, which is however only effective when the movement error is minimal. In particular, when dynamic obstacles make the pre-defined path unfeasible and cause the robot to deviate from the intended path, there is currently no specialised technique available to guarantee self-entanglement-free motions as the robot navigates around obstacles and continues tracking. This is done in this work.

[1] https://github.com/jiangpin-legend/T-TDT

III. Problem Definition

Robot Kinematics. See Fig. 2 for illustration. The model of the tethered differential-driven robot follows existing conventions, which are briefly re-stated here. A configuration of the robot consists of two parts, the spatial pose of the differential-driven mobile unit, denoted as $c = [x, y, \theta]$, and the shape of the tether s.

The tether's thickness is neglected, and it is assumed to be taut at all times. As such, the shape of the tether s can be estimated as a sequence of points, beginning with the fixed anchoring point b in the environment, followed by the points where the tether contacts obstacles, denoted as P, and ending at the tether-robot contact point a. The last tether-obstacle contact point is denoted as p.

Regarding the collision model of the robot, the collision between the tether and dynamic obstacles is neglected, because the tether is a passive structure of the robot system which cannot actively avoid obstacles. The collision module of the mobile unit is assumed as a polygon.

To prevent the robot from self-entanglement, at every moment, the heading orientation of the mobile unit is conditioned by the tether retracting direction which is estimated as $\vec{ap}$. See Fig. 2 for illustration. The admissible relative angle between these orientations is designated as $[\Phi_1, \Phi_2]$, a parameter determined by the mechanical structure of the robot.

Environmental Assumption. Tether kinematics introduce additional requirements for the dynamic obstacles in the environment. Firstly, it is necessary to assume that any obstacle in contact with the robot tether remains static, as otherwise, there is a possibility of a situation where the shape of the tether changes even if the robot itself does not move. Secondly, the presence of dynamic obstacles should not affect the existence of a tether-length-admissible resultant path in the same homotopy class as the pre-defined path. To be precise, it has been proven in [28] that a tether-length-admissible resultant path exists only if the induced goal configuration is tether-length-admissible. However, although the tether length admissibility of the robot's next configuration is checked at every moment in the trajectory deformation and tracking algorithm, verifying the admissibility of the goal configuration is not local, i.e., is a task that goes beyond the scope of a local path tracking algorithm, which only focuses on the vicinity of the robot.

Tethered Trajectory Deforming and Tracking. Given a 2-dimensional environment which have been pre-modelled into a map M, and a pre-defined robot path connecting the starting location and the goal location, the tethered trajectory deforming and tracking (T-TDT) task is to solve for a sequence of robot's steering and velocity commands, such that the robot can traverse from the starting location to the goal location, satisfying that:

1) (Common constraints for navigation) The commands are kinematically feasible. The robot does not hit any obstacle.
2) (Tether length admissible) The length of the robot tether does not exceed its maximal length.

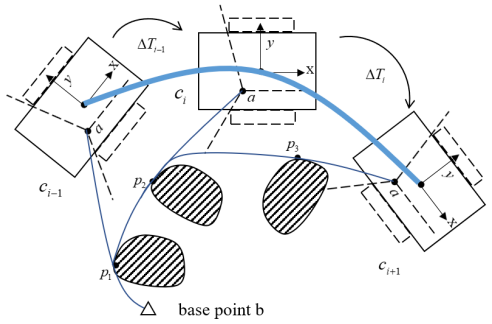


Fig. 3. Illustration of parameters used for describing a tethered robot motion.

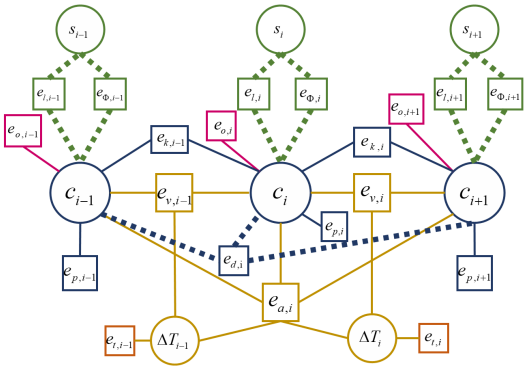


Fig. 4. Illustration of the factor graph structure. Vertices are variables of interests. Edges are error functions to be optimised. Green edges indicate the error functions related to the robot tether, blue edges indicate the errors related only to the pose of the mobile unit, pink edges represent the penalty for obstacle avoidance, and yellow edges are time-related penalties, such as velocity and acceleration. Dashed edges are the modules freshly introduced in this paper.

3) (Self-entanglement-free) The relative angle between the robot's heading orientation and the tether retracting orientation is admissible.

IV. Proposed Algorithm

In this section, the proposed algorithm is presented which utilises hyper-graph optimisation [32]. A brief overview of the objectives in existing hyper-graph optimisation algorithms is provided for self-containment. Then, we delve into the details of the additional self-entanglement-free constraint.

A. Hyper-graph-based Optimisation

Existing framework transforms the untethered trajectory deformation and tracking task to a hyper-graph structure, where vertices are the variables to be optimised, and edges are error functions relative to the variables [5] . Then, graph-based optimisation techniques can be applied for the trajectory deformation and tracking.

To be precise, given the initial robot location and the global path, a short sequence of the path waypoints that the mobile unit of the tethered robot is about to visit is selected, denoted as $c_0, c_1, \cdots, c_n$. Then, based on the maximum velocity of the robot, the travelling time between each pair of consecutive waypoints are initialised, denoted as $\Delta T_0, \Delta T_1, \cdots, \Delta T_{n-1}$, See

Fig. 3 for illustration. Then the set of parameters subject to optimisation is defined by:

$$\beta = \{c_0, \Delta T_0, c_1, \Delta T_1, ..., c_{n-1}, \Delta T_{n-1}, c_n\} \tag{1}$$

Under this parameterisation, the trajectory deforming and tracking problem has been formulated as finding the path that minimises the following weighed cost:

$$Cost(\beta) = \sum_k \omega_k * f_k(\beta) \tag{2}$$

where $f_k(\cdot)$ are metrics, and ω_k are the corresponding weights set by the user. Finally, Levenberg-Marquardt iteration is applied to optimise the hyper-graph and minimise the combinational costs. The main metrics that have been incorporated into the hyper-graph framework [5] are listed as follows:

1) The penalty $e_{v,i}$ and $e_{a,i}$ for the robot commands that violate the maximum velocity constraint and the maximum acceleration constraint.
2) The penalty $e_{k,i}$ for the robot commands that violate the robot kinematics, where in the differential-driven case it is usually assigning non-zero velocity to the robot's y-direction.
3) The penalty $e_{o,i}$ for the robot poses that stay close to obstacles.
4) The penalty $e_{t,i}$ and $e_{p,i}$ for long robot execution time and long travelling distance.

See Fig. 4 (solid edges) for the demonstration of the hyper-graph structure.

B. Tether-related Objectives

In addition to the previously discussed factors, this subsection introduces several new terms aiming at optimising trajectories specifically for tethered robots.

Self-Entanglement-Free Property. The shape of the robot tether, serving as a fundamental structure for detecting self-entanglement and verifying tether length admissibility, must be explicitly calculated. This calculation is doable because the tether has been assumed to be taut, allowing the estimation using an existing module in [9]. In each iteration, once the tether shape is calculated, the relative angle Φ_i is known. A penalty function for the violation of the self-entanglement-free constraint is defined as follows

$$e_{\Phi,i} = \begin{cases} \Phi_1 - \Phi_i, & \text{if } \Phi_i < \Phi_1 \\ \Phi_i - \Phi_2, & \text{if } \Phi_i > \Phi_2 \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

Tether-Length-Admissibility. Now that the shape of the tether has been known, its length l_i becomes known as well. Consequently, a penalty function for exceeding the maximum tether length is naturally defined as

$$e_{l,i} = \begin{cases} l_i - l_{\max}, & l_i > l_{\max} \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

where $l_{\max}$ is the maximum allowable length of the tether.

Switching between Forward and Backward Movements. The empirical findings of the authors suggest that frequent back-and-forth movements are effective in providing lateral movement without introducing large angular changes. However, an excessive number of these back-and-forth movements inevitably impact the overall execution quality, affecting factors such as the time to completion and the trajectory smoothness. Therefore, for the T-TDT task, the explicit introduction of an additional penalty for the switches between forward and backward movements might become a vital component for the improved quality of the resultant trajectory. This can be simply defined as a binary cost

$$e_{d,i} = \begin{cases} 1, & \text{if the movement direction changes} \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

The positions where the new modules are incorporated have been visualised also in Fig. 4 (dashed edges).

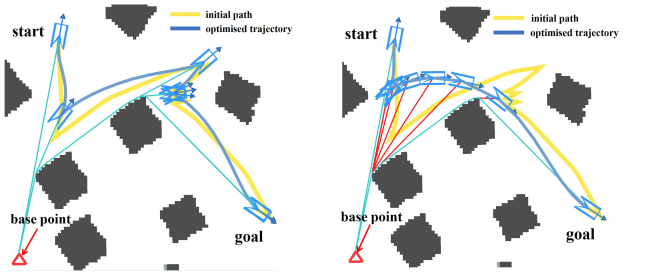
Finally, it is important to note that estimating the shape of the tether, which essentially involves the identification of all tether-obstacle contact points, is not a differentiable task. Specifically, when the tether contacts an obstacle, the location of the last tether-obstacle contact point changes abruptly, without a continuous transition. As a result, the tether shape estimation can only be implemented outside the optimisation framework, before the optimisation process begins.

V. Experimental Results

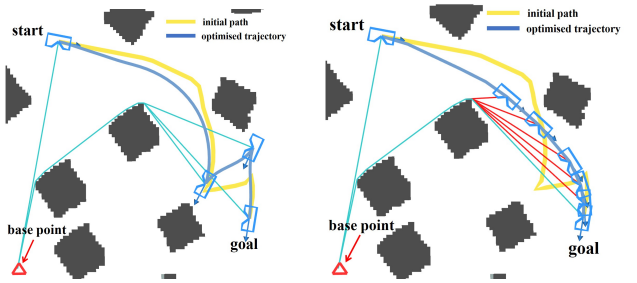
The proposed algorithm performs trajectory deforming and tracking for a tethered differential-driven robot, by considering the presence of environmental obstacles and optimising multiple task-specific metrics. These metrics include maximising the distance to obstacles, minimising the risk of tether self-entanglement and exceeding the tether length limit, as well as other common objectives such as maximising velocity for trajectory tracking and minimising the number of switches between forward and backward movements. To the best of the authors' knowledge, there did not exist an algorithm that explicitly considered the SEF constraint for tethered robots. In Section V-A, comparative experiments are carried out to show the necessity of the proposed algorithm. Then in Section V-B, enormous simulated experiments are provided to demonstrate the computational performance of the proposed algorithm. Finally, the proposed algorithm is deployed into real-world robot to validate its applicability.

A. Comparative Studies

Comparative experiments are provided to show the functionality of the proposed algorithm against existing algorithms. Given the same base point, initial location, goal location, admissible range of relative angle, and the initial path for tracing, the resultant paths generated by the proposed algorithm and the Timed-Elastic-Band (TEB) algorithm [5] are provided in Fig. 5. In both Case back-1 and Case back-2, the tether of the robot extends to the back. And in Case right-1 and Case right-2, the tether extends to the right. In contrast to the existing algorithm that might change a SEF path to a non-SEF one, the proposed algorithm can keep the tether-related constraints whilst the robot execution. See Fig. 6 for the illustration of the relative angle variation using both algorithms. Therefore, when the self-entanglement issue of a tethered robot cannot be



(a) Case back-1 (Ours) (b) Case back-2 (TEB)



(c) Case right-1 (Ours) (d) Case right-2 (TEB)

Fig. 5. Comparison of the results provided by the proposed method and the Timed-Elastic-Band (TEB) algorithm given the same initial path and other settings. In (a) and (b), the same initial SEF path is given, and another same initial SEF path is given in (c) and (d). The red lines representing the tethers in (b) and (d) indicate the violation of the SEF constraint.

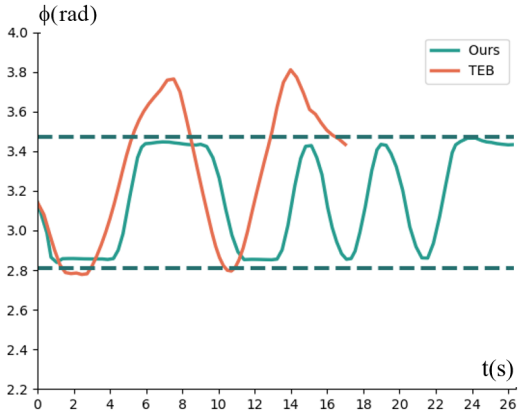


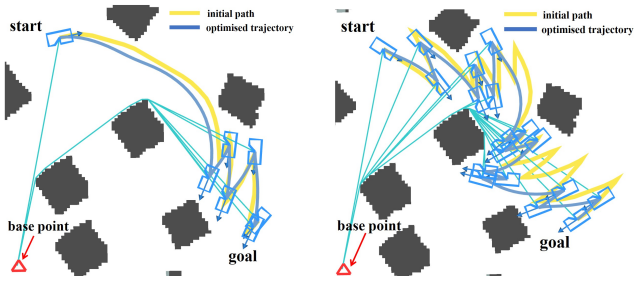
Fig. 6. The variation of the relative angle between the robot's heading direction and the tether retracting direction. Comparison are conducted between TEB and the proposed algorithm.

ignored, the proposed algorithm has the specialised advantage to be applied.

B. Extensive Experiments

In this subsection, more simulated experiments are conducted to further demonstrate the performance of the proposed algorithm. In these experiments, base locations, starting locations, goal locations, and relative angles are all changing to different values. Relative statistics are summarised in Table. I.

For the most challenging scenes please see the testing results for a tethered robot whose tether extends to its right in Fig. 7, (a) and (b), where the proposed algorithm has to leverage



(a) Challenging case-1 (b) Challenging case-2

Fig. 7. Illustration of challenging T-PDT solutions.

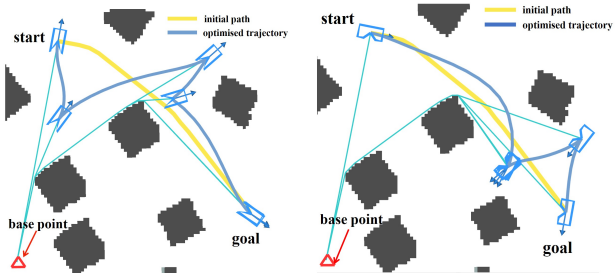


Fig. 8. Illustration of the feasible robot motions (blue) optimised from non-SEF initial paths (yellow).

all the narrow corridors between obstacles to perform best-possible solutions, obtaining trade-offs among maximising distance to obstacles, minimising the number of movement direction switches, and most importantly, complying with the robot kinematics. The proposed algorithm successfully finished all these tasks.

As for the computational efficiency of the proposed algorithm, although the resultant paths are non-intuitive, the searching space is actually highly restricted given by the tether-related constraints. Results show that in most cases, the optimisation time is less than 100 milliseconds for a path of 10 metres, which satisfies the efficiency requirement as an online algorithm.

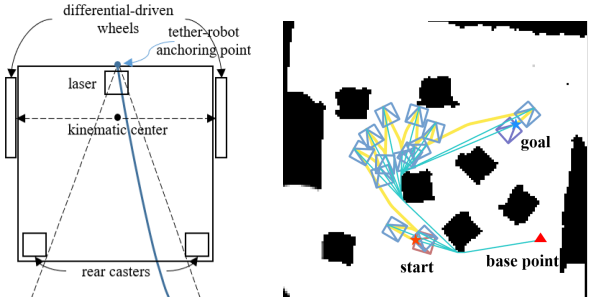
The readers are referred to the supplementary video for the animation of the full robot execution in simulation, where the proposed algorithm iteratively acquires unoptimised path in the vicinity of the robot and establishes online optimisation in a real-time manner.

It is also noteworthy that the proposed algorithm cannot replace existing SEF path planners. To reveal this, the initial paths for optimisation were changed to non-SEF paths, which were generated by A* [33] as well as a naive path smoothing process. See Fig. 8 for illustration and Table. I for the computational time. Results show that although the proposed algorithm is still capable of generating a valid solution for the robot execution, the computational time is much longer than those dealing with a SEF initial path. Hence, the practitioners are recommended to use both SEF path planners and the proposed T-TDT algorithm to obtain the best performance.

TABLE I
PERFORMANCE OF CASE STUDIES

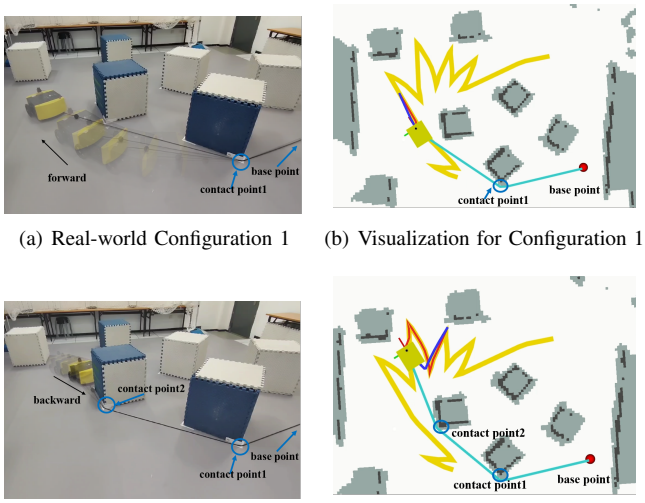
Case	Initial Path	$[\Phi_1,\Phi_2]$	Path Length	Path Time	Dist	Cost	Build Time	Opt Time	Total Time	Fault Num	Pose Num
back-1	SEF Path	[2.73,3.56]	11.16 m	25.80 s	0.35 m	10.34	41.11 ms	19.44 ms	61.27 ms	**0**	95
back-2		[2.81,3.47]	9.88 m	19.04 s	0.29 m	18.21	35.65 ms	8.02 ms	44.13 ms	**0**	36
back-3		[2.75,3.54]	13.57 m	32.12 s	0.43 m	75.60	35.79 ms	32.27 ms	69.14 ms	**0**	111
back-4		[2.75,3.54]	9.13 m	20.86 s	0.37 m	14.65	22.09 ms	11.18 ms	33.79 ms	**0**	73
back-5		[2.75,3.54]	13.81 m	29.76 s	0.35 m	9.94	80.54 ms	16.51 ms	97.61 ms	**0**	119
back-6		[2.77,3.51]	8.75 m	13.45 s	0.37 m	8.50	21.35 ms	7.36 ms	29.07 ms	**0**	48
right-1		[3.98,4.66]	7.75 m	14.37 s	0.37 m	10.97	45.36 ms	4.29 ms	49.88 ms	**0**	27
right-2		[3.98,5.45]	6.41 m	14.68 s	0.37 m	14.14	54.83 ms	7.85 ms	63.05 ms	**0**	60
right-3		[4.76,5.45]	19.31 m	45.32 s	0.34 m	15.33	92.33 ms	18.39 ms	111.49 ms	**0**	182
right-4		[3.98,5.45]	7.99 m	17.54 s	0.51 m	10.06	52.40 ms	10.94 ms	63.85 ms	**0**	64
right-5		[3.98,4.66]	18.25 m	51.49 s	0.38 m	20.04	84.43 ms	24.30 ms	109.65 ms	**0**	192
right-6		[4.76,5.45]	11.62 m	37.35 s	0.36 m	20.50	58.32 ms	19.80 ms	78.91 ms	**0**	134
back-1-teb [5]		[2.73,3.56]	8.33 m	15.93 s	0.28 m	10.61	57.43 ms	5.94 ms	63.66 ms	15	38
back-2-teb [5]		[2.81,3.47]	7.60 m	12.29 s	0.34 m	8.29	37.91 ms	4.30 ms	42.44 ms	9	29
right-1-teb [5]		[3.98,4.66]	6.63 m	11.04 s	0.41 m	7.76	44.95 ms	4.33 ms	49.60 ms	11	23
right-2-teb [5]		[3.98,5.45]	6.10 m	14.41 s	0.39 m	8.40	35.10 ms	15.54 ms	51.05 ms	57	59
right-1-a-star	Non-SEF Path	[3.98,4.66]	9.17 m	22.83 s	0.41 m	31.61	571.38 ms	131.03 ms	708.85 ms	**0**	77
right-2-a-star		[3.98,5.45]	7.28 m	15.55 s	0.38 m	6.85	86.51 ms	14.91 ms	102.14 ms	**0**	51
back-2-a-star		[2.81,3.47]	10.17 m	16.79 s	0.31 m	7.89	47.44 ms	5.09 ms	571.84 ms	**0**	59

Experiments are conducted on a laptop with Intel Core i5 3.5GHz with 16 GB of RAM.
Dist: the average distance to nearest obstacle.
Total Time: total time for algorithm calculation.
Opt Time: time for non-linear optimisation.
Build Time: time for building hyper-graph for optimisation.
Fault Num: the number of robot configurations that violate the SEF constraint.



(a) robot structure (b) the environment

Fig. 9. Illustration of (a) The real-world robot kinematics and (b) the map of the real-world environment.



(a) Real-world Configuration 1 (b) Visualization for Configuration 1

(c) Real-world Configuration 2 (d) Visualization for Configuration 2

Fig. 10. Illustration of the real-world experiment.

C. Real-world Illustrations

The proposed algorithm's effectiveness is further validated through real-world scenarios, as depicted in Fig. 9(a) for the robot model and Fig. 9(b) for the real-world scene. In these experiments, the tether is connected to the front bottom of the robot chassis, below the laser, and extends to the back, positioned between two rear casters. The map is pre-builted as a grid map using Catographer [2], [9] is used to generate the initial path. Given the SEF constraint, the robot is unable to perform a large right turning, and must rely on frequent forward and backward movements to achieve slow lateral motion. The proposed algorithm does not optimise the path by the intuitive straightening and smoothing, as is common in untethered trajectory planners. Instead, it preserves the SEF motion patterns and successfully guides the robot towards the goal, as shown in Fig. 10. The readers are referred to the supplementary video for the real-world robot executions.

VI. CONCLUSION

This paper presented an optimisation-based trajectory deformation and tracking algorithm for tethered differential-driven robots. Although the most recent path planning algorithms for tethered differential-driven robots without an omni-directional tether retracting mechanism have been capable of generating self-entanglement-free (SEF) robot paths, when facing with unavoidable movement errors, the robot still encounters the self-entanglement issue, as no existing trajectory planner can iteratively generate robot commands complying with SEF constraint. This paper fills this gap by explicitly introducing the tether-robot relative angle estimation and the SEF constraint into existing optimisation-based framework. Simulated and real-world experiments have revealed its validity to guide the robot towards the goal with minimised self-entanglement risk. The open-source implementation has been released as a C++ ROS package for the benefits of the community.

[2]https://github.com/cartographer-project/cartographer.git

REFERENCES

[1] O. Amidi and C. E. Thorpe, "Integrated mobile robot control," W. H. Chun and W. J. Wolfe, Eds., pp. 504–523.

[2] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robotics & Automation Magazine*, vol. 4, no. 1, pp. 23–33.

[3] S. Quinlan and O. Khatib, "Elastic bands: Connecting path planning and control," in *[1993] Proceedings IEEE International Conference on Robotics and Automation*, pp. 802–807 vol.2.

[4] H. Kurniawati and T. Fraichard, "From path to trajectory deformation," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 159–164.

[5] C. Rösmann, W. Feiten, T. Wosch, F. Hoffmann, and T. Bertram, "Efficient trajectory optimization using a sparse model," in *Proceedings of the 2013 European Conference on Mobile Robots*. IEEE, pp. 138–143.

[6] C. Rösmann, F. Hoffmann, and T. Bertram, "Kinodynamic trajectory optimization and control for car-like robots," in *Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5681–5686.

[7] C. Rösmann, F. Hoffmann, and T. Bertram, "Integrated online trajectory planning and optimization in distinctive topologies," *Robotics and Autonomous Systems*, vol. 88, pp. 142–153.

[8] C. Rösmann, F. Hoffmann, and T. Bertram, "Planning of multiple robot trajectories in distinctive topologies," in *Proceedings of the 2015 European Conference on Mobile Robots (ECMR)*. IEEE, 2015, pp. 1–6.

[9] T. Yang, J. Liu, Y. Wang, and R. Xiong, "Self-Entanglement-Free Tethered Path Planning for Non-Particle Differential-Driven Robot," in *Proceedings of the 2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7816–7822.

[10] A. A. Nassiraei, Y. Kawamura, A. Ahrary, Y. Mikuriya, and K. Ishii, "Concept and design of a fully autonomous sewer pipe inspection mobile robot "kantaro"," in *Proceedings of the 2007 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2007, pp. 136–143.

[11] D. Hong, S. A. Velinsky, and K. Yamazaki, "Tethered mobile robot for automating highway maintenance operations," *Robotics and Computer-Integrated Manufacturing*, vol. 13, no. 4, pp. 297–307, 1997.

[12] I. Shnaps and E. Rimon, "Online coverage by a tethered autonomous mobile robot in planar unknown environments," *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 966–974, 2014.

[13] L. Mechsy, M. Dias, W. Pragithmukar, and A. Kulasekera, "A novel offline coverage path planning algorithm for a tethered robot," in *Proceedings of the 2017 International Conference on Control, Automation and Systems (ICCAS)*. IEEE, 2017, pp. 218–223.

[14] G. Sharma, P. Poudel, A. Dutta, V. Zeinali, T. T. Khoei, and J.-H. Kim, "A 2-approximation algorithm for the online tethered coverage problem." in *Robotics: Science and Systems (RSS)*, 2019.

[15] K. S. Pratt, R. R. Murphy, J. L. Burke, J. Craighead, C. Griffin, and S. Stover, "Use of tethered small unmanned aerial system at berkman plaza ii collapse," in *Proceedings of the 2008 IEEE International Workshop on Safety, Security and Rescue Robotics (SSRR)*. IEEE, 2008, pp. 134–139.

[16] P. Abad-Manterola, I. A. Nesnas, and J. W. Burdick, "Motion planning on steep terrain for the tethered axel rover," in *Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2011, pp. 4188–4195.

[17] M. M. Tanner, J. W. Burdick, and I. A. Nesnas, "Online motion planning for tethered robots in extreme terrain," in *Proceedings of the 2013 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2013, pp. 5557–5564.

[18] D. Shapovalov and G. A. Pereira, "Exploration of unknown environments with a tethered mobile robot," in *Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 6826–6831.

[19] P. Xavier, "Shortest path planning for a tethered robot or an anchored cable," in *Proceedings of the 1999 IEEE International Conference on Robotics and Automation*, vol. 2. IEEE, pp. 1011–1017.

[20] P. Brass, I. Vigan, and N. Xu, "Shortest path planning for a tethered robot," *Computational Geometry*, vol. 48, no. 9, pp. 732–742, 2015.

[21] Soonkyum Kim and M. Likhachev, "Path planning for a tethered robot using Multi-Heuristic A* with topology-based heuristics," in *Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 4656–4663.

[22] S. Kim, S. Bhattacharya, and V. Kumar, "Path planning for a tethered mobile robot," in *Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 1132–1139.

[23] S. Kim and M. Likhachev, "Path planning for a tethered robot using multi-heuristic a* with topology-based heuristics," in *Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 4656–4663.

[24] S. McCammon and G. A. Hollinger, "Planning and executing optimal non-entangling paths for tethered underwater vehicles," in *Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 3040–3046.

[25] M. Cao, K. Cao, S. Yuan, T.-M. Nguyen, and L. Xie, "Neptune: Nonentangling trajectory planning for multiple tethered unmanned vehicles," *IEEE Transactions on Robotics*, 2023.

[26] R. H. Teshnizi and D. A. Shell, "Motion planning for a pair of tethered robots," *Autonomous Robots*, vol. 45, no. 5, pp. 693–707, 2021.

[27] S. Martínez-Rozas, D. Alejo, F. Caballero, and L. Merino, "Optimization-based Trajectory Planning for Tethered Aerial Robots," in *Proceedings of the Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 362–368.

[28] T. Yang, R. Xiong, and Y. Wang, "Efficient distance-optimal tethered path planning in planar environments: The workspace convexity," *arXiv preprint arXiv:2208.03969*, 2022.

[29] S. Bhattacharya and R. Ghrist, "Path homotopy invariants and their application to optimal trajectory planning," *Annals of Mathematics and Artificial Intelligence*, vol. 84, no. 3, pp. 139–160, 2018.

[30] T. Yang, J. Liu, Y. Wang, and R. Xiong, "Sparse waypoint validity checking for self-entanglement-free tethered path planning," *arXiv preprint arXiv:2308.15931*, 2023.

[31] S. Bhattacharya, M. Likhachev, and V. Kumar, "Topological constraints in search-based robot path planning," *Autonomous Robots*, vol. 33, no. 3, pp. 273–290, 2012.

[32] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "G2o: A general framework for graph optimization," in *Proceedings of the 2011 IEEE International Conference on Robotics and Automation*, pp. 3607–3613.

[33] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.