

# Quality-guaranteed Dubins Path Planning for USV Based on Mixed-integer Piecewise-linear Programming for Addressing the Extended Minimum-time Intercept Problem

Xing Zhou<sup>1</sup>, Kelin Zhu<sup>2</sup>, Shuang Liu<sup>3</sup>, Zhaoqing Li<sup>1</sup>, Wenxin Zhang<sup>1</sup> and Kang Du<sup>1</sup>

Received: 08 September 2024 / Accepted: 04 September 2025  
© Harbin Engineering University and Springer-Verlag GmbH Germany, part of Springer Nature 2026

## Abstract

During the use of robotics in applications such as antiterrorism or combat, a motion-constrained pursuer vehicle, such as a Dubins unmanned surface vehicle (USV), must get close enough (within a prescribed zero or positive distance) to a moving target as quickly as possible, resulting in the extended minimum-time intercept problem (EMTIP). Existing research has primarily focused on the zero-distance intercept problem, MTIP, establishing the necessary or sufficient conditions for MTIP optimality, and utilizing analytic algorithms, such as root-finding algorithms, to calculate the optimal solutions. However, these approaches depend heavily on the properties of the analytic algorithm, making them inapplicable when problem settings change, such as in the case of a positive effective range or complicated target motions outside uniform rectilinear motion. In this study, an approach employing a high-accuracy and quality-guaranteed mixed-integer piecewise-linear program (QG-PWL) is proposed for the EMTIP. This program can accommodate different effective interception ranges and complicated target motions (variable velocity or complicated trajectories). The high accuracy and quality guarantees of QG-PWL originate from elegant strategies such as piecewise linearization and other developed operation strategies. The approximate error in the intercept path length is proved to be bounded to  $h^2/(4\sqrt{2})$ , where  $h$  is the piecewise length.

**Keywords** Minimum-time intercept problem; Dubins vehicle; Mixed-integer piecewise-linear program; Linearization; Approximate error; trigonometric function; USV

## 1 Introduction

In many scenarios requiring unmanned systems, such as antiterrorism, defense, or combat (Niu et al., 2023; Fan et al., 2023; Ntakolia et al., 2023), a pursuer vehicle needs

to reach a moving target as quickly as possible, for example, an unmanned police boat aiming to intercept an intruding terrorist boat. Many real-life pursuers, such as unmanned surface vehicles (USVs), Ackermann-style UGVs (Rathinam et al., 2019; Moon et al., 2023), fixed-wing UAVs, aircraft, USVs, and even missiles, are subject to motion constraints (LaValle, 2006; Liu et al., 2023), including the typical minimum turning radius constraint: When a pursuer travels at a constant velocity, the pursuer cannot make sharp turns. Thus, the radius of the trajectories exceeds a specific value. The pursuer is commonly referred to as a Dubins vehicle (Dubins, 1957).

In this study, we focus on a Dubins vehicle pursuer intercepting a moving target in a two-dimensional plane, as depicted in Figure 1. The moving target may be either a manned vehicle or an unmanned object. Additionally, we consider the effective interception range ( $E_r$ ), which represents the maximum distance at which the pursuer can still successfully intercept the target. For example, the pursuer can self-destruct by exploding to eliminate the target.

The exact reach problem ( $E_r = 0$ ) is known as the minimum-time intercept problem (MTIP), where the targets should be exactly reached by the pursuer (Zheng et al., 2021). The properties and original problem have been ana-

## Article Highlights

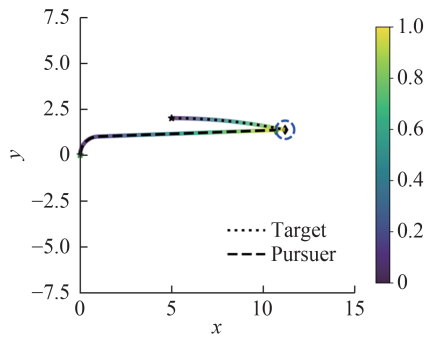
- Proposes the extended minimum-time intercept problem (EMTIP) for motion-constrained pursuers in critical scenarios like antiterrorism and combat.
- Resolves the inapplicability of existing analytic algorithms under complex constraint problem settings.
- Proposes a high-accuracy, quality-guaranteed mixed-integer piecewise-linear program QG-PWL adaptable to diverse interception ranges and complex target motions.
- Proves the bounded approximate error of the intercept path length, with the error related to the piecewise length.

✉ Xing Zhou  
zhouxing@nudt.edu.cn

<sup>1</sup> College of Intelligence Science and Technology, National University of Defense Technology, Changsha 410073, China

<sup>2</sup> Defense Innovation Institute, Beijing 100091, China

<sup>3</sup> College of Computer Science and Technology, Taiyuan University of Technology, Taiyuan 030024, China



**Figure 1** Dubins vehicle (e.g., USV) pursuer attempting to intercept a moving target as quickly as possible. “Intercept” means reaching within a prescribed effective range  $E_r, E_r \geq 0$ . The target’s motion and trajectories may be more complicated than straight lines, such as sinusoidal curves. The color indicates the time elapsed ratio

lyzed in many previous studies. In this paper, the MTIP is extended to account for the target’s motion dynamics and the pursuer’s effective range, and the resulting formulation is referred to as the extended MTIP (EMTIP). Related studies are illustrated in Figure 2, and a literature review is presented below.

In 1957, Dubins proposed the Dubins path between two known configurations  $(x_1, y_1, \theta_1)$  and  $(x_2, y_2, \theta_2)$  (Dubins 1957) for turning radius constrained vehicles, where each configuration contains the  $x, y$  coordinates and the heading angle. The quickest path for the vehicle is a concatenation of left ( $L$ ) or right ( $R$ ) minimum-radius circular arcs and straight ( $S$ ) line segments. The study proved that in an open environment, the vehicle’s optimal path between two configurations must be one of the following six types of paths: left-straight-left (LSL), left-straight-right (LSR), right-straight-left (RSL), right-straight-right (RSR), left-right-left (LRL), or right-left-right (RLR) (note that some segments of each path type may be reduced to 0 length).

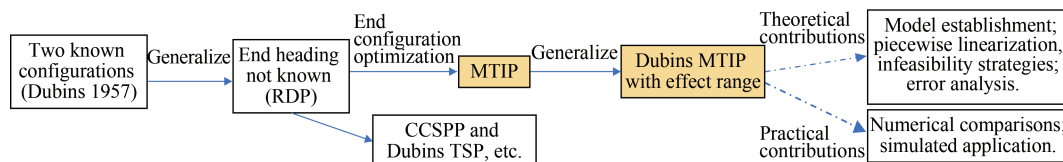
For two known configurations, Bui and Shkel (Bui et al., 1994; Shkel and Lumelsky, 2001; De Palma and Parlange, 2022) investigated the closed-form solution of the Dubins paths. However, if the heading angle in the final configuration is free while the end  $(x, y)$  is fixed, then the problem becomes the relaxed Dubins problem (RDP). Using geometric optimal control theory, Sussmann and Tang (1991) developed approaches for the RDP problem. The RDP problem has been generalized to two or more unknown headings, such as the three consecutive points problem (Sadeghi and Smith, 2016; Chen and Shima, 2019), the Curvature-Constrained Shortest-Path Problem (CCSPP), and the Dubins Traveling Salesman Problem

(Dubins TSP) (Fu et al., 2023). However, these problems are static scenarios with fixed  $(x, y)$ .

A target moving with simple motion was studied by generalizing the RDP to dynamic and adversarial scenarios (Matveev et al., 2011). Meyer et al. (2015) and Shima (2011) established some sufficient conditions for intercept. Zheng et al. (2021) studied the MTIP and transformed it to find the zeros of real-valued functions, which is theoretically and practically optimal. Related studies then considered the moving obstacle lateral intercept and multi-target (Gopalan et al., 2016; Jha et al., 2022; Buzikov et al., 2022).

However, in these studies, the evaluated targets generally had a constant velocity or rectilinear motion or exact intercept. These conditions may have been the focus of previous studies because scenarios such as those addressed herein were not encountered in those studies. The present scenarios introduce mathematical complexities that render previous analytical methods inapplicable. The trigonometric function and angular modulo constraints mitigate the effectiveness of an analytic approach or trivial mathematical programming. Therefore, new methods need to be developed. To our knowledge, algorithms that are highly accurate (optimal or with bounded error) and are capable of tackling the EMTIP (with the effective range  $E_r$ ) and complicated target motions have yet to be developed. Herein, we transform the nonlinear and mixed-integer constraints to be tackled using piecewise linearization, which has quality guarantees. The following are the contributions of this study:

- We formulate the scenario encountered in antiterrorism as an extended MTIP. This approach is more general and applicable than those in previous studies. The program of the EMTIP is elaborately built, and its objectives and constraints are first established in the literature.
- To tackle the hindrance caused by the trigonometric and modulo constraints, we employ piecewise-linear strategies and effectively realize optimization with these otherwise intractable trigonometric constraints. Moreover, the use of the big-M strategy and modular operation transformation makes the program solvable by solvers. The approximation error (tolerance) for the solution obtained with the QG-PWL approach versus the analytic optimal length is successfully analyzed in some cases, affording a guarantee smaller than  $h^2/(4\sqrt{2})$  for the piecewise length.
- The QG-PWL approach is validated by comprehensive numerical tests, demonstrating its effectiveness and generalizability compared to the genetic algorithm (GA).



**Figure 2** EMTIP problem and related problems

## 2 Formulation of EMTIP and preliminary of program

This section describes the essentials for the Dubins vehicle and the formulation of the problem. Table 1 lists and explains some notations that are used throughout the paper.

**Table 1** Notations and explanations

Symbol	Explanation
$L/R/S$	Left turn circular/right turn circular/straight path segment
$E_r$	$E_r \geq 0$ , effective range for a pursuer to successfully intercept the target
$t_m$	Shortest time for a successful intercept
$z := (x, y, \theta)$	State of the pursuer, a function of the time $t$
$x_0, y_0, \alpha$	Coordinates and heading of the pursuer at the beginning
$x_p, y_p, \beta$	Coordinates and heading of pursuer when intercepting successfully
$PT$	Set comprising the six path types, one of which is the optimal path
$t, p, q, b$	Time (also the length) of each segment in each path type, and the feasibility implication variable for each path type
$\sin[\theta](\cos[\theta])$	Piecewise-linear approximation of the sine (cosine) function

### 2.1 EMTIP

#### 2.1.1 Dubins pursuer

The state (or configuration) of the pursuer is denoted by  $z := (x, y, \theta) \in \mathbb{R}^2 \times \mathbb{S}^1$ , which consists of a point vector  $(x, y) \in \mathbb{R}^2$  and a heading angle  $\theta \in \mathbb{S}^1$ . Assume that the pursuer is based on the Dubins vehicle model, which resembles the “bicycle” model. In this model (taking the rear end of the “bicycle” as the reference point), the forward and angular speeds are aligned with the body of the “bicycle”. The linear speed to the global  $x$ -axis is  $\dot{x}$  and that to the global  $y$ -axis is  $\dot{y}$ , which correspond to  $v \cos \theta$  and  $v \sin \theta$ , respectively.  $\dot{\theta}$  is the angle speed, where  $\dot{\theta} = v/\rho$ ;  $\rho$  is the turning radius.  $\rho$  can be obtained according to trigonometric relations (Shkel and Lumelsky, 2001), i.e.,  $\tan \phi = \frac{L}{\rho}$ , where  $\phi$  is the angle of the steering-wheel,  $L$  is the body length of the vehicle, and  $\rho$  is the turning radius. Thus,  $\rho = L/\tan \phi$ . Therefore,  $\dot{\theta} = v/\rho = v/(L/\tan \phi) = \frac{v \tan \phi}{L}$ .

The kinematics of the pursuer is represented as follows:

$$\dot{z} = \begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{\theta}(t) \end{bmatrix} = \begin{bmatrix} v \cos \theta(t) \\ v \sin \theta(t) \\ \frac{v \tan \phi}{L} \end{bmatrix}, \quad |\phi| \leq \Phi_{\max} < \frac{\pi}{2} \quad (1)$$

where  $\Phi_{\max}$  is the largest possible steering-wheel angle;  $t \geq 0$  denotes the time. Note that  $\rho = \frac{L}{\tan \phi} \geq \frac{L}{\tan \Phi_{\max}} = \rho_0$ ; thus, the turning radius  $\rho$  must be larger than or equal to a threshold  $\rho_0$ . Without losing generality (Shkel and Lumelsky, 2001), the velocity of the pursuer  $v$  can be normalized to 1 m/s; the minimum turning radius  $\rho_0$  can also generally be normalized to 1 m.

After the normalization,  $\dot{x}(t) = \cos \theta(t)$ ,  $\dot{y}(t) = \sin \theta(t)$ ,  $\dot{\theta}(t) = u$ ,  $u \in \{-1, 0, 1\}$ , where  $u$  (must be in the discrete set, as proved by Dubins (1957) and Zheng et al. (2021), is the control input (also the angular speed) representing the minimum-radius right turn (with  $u = -1$  rad/s) or straight move (with  $u = 0$  rad/s), or the minimum-radius left turn (with  $u = 1$  rad/s) of the pursuer.

The left, right, and straight moves of  $\mu$  meters can be derived (Shkel and Lumelsky, 2001) using vectors or complex numbers and can be neatly represented as  $L_\mu, R_\mu, S_\mu$ , respectively:

$$\begin{aligned} L_\mu(x, y, \alpha) &= (x + \sin(\alpha + \mu) - \sin \alpha, \\ &\quad y - \cos(\alpha + \mu) + \cos \alpha, \alpha + \mu) \\ R_\mu(x, y, \alpha) &= (x - \sin(\alpha - \mu) + \sin \alpha, \\ &\quad y + \cos(\alpha - \mu) - \cos \alpha, \alpha - \mu) \\ S_\mu(x, y, \alpha) &= (x + \mu \cos \alpha, y + \mu \sin \alpha, \alpha) \end{aligned} \quad (2)$$

Note that for the normalized pursuer, the length moved is equal to the central turning angle, and the quickest path is also the shortest path because the velocity is normalized to 1 m/s.

#### 2.1.2 Moving target

The velocity of the target is denoted as  $v(\tau) = (v_{x\tau}, v_{y\tau}) \in \mathbb{R}^2$ , which comprises the motion of the target. Assume that the position of the target at the initial time  $t = 0$  is  $(x_0, y_0)$ . Thus, the position of the target at any time  $t \geq 0$  can be expressed as:

$$E(t) = (x_0, y_0) + \int_0^t v(\tau) d\tau \quad (3)$$

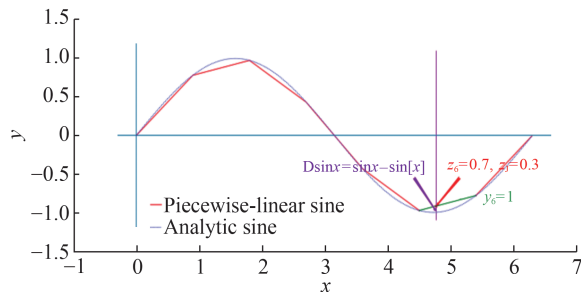
Similar to the previous study, herein, we assume that the  $E(t)$  of the target for any  $t \geq 0$  is available to the pursuer. In most previous studies, the velocity of the target (and thus the motion) is relatively simple, such as uniform rectilinear motion, whereas the developed QG-PWL approach can tackle more complicated target motion, such as uniformly accelerated motion, curved horizontal projectile motion, and even sinusoidal wave motion. The formal description of the EMTIP is as follows:

Extended minimum-time intercept problem (EMTIP) Find a minimum time  $t_m > 0$  so that the system in Eq. (1) is steered by  $u(\cdot)$  over the interval  $[0, t_m]$  from  $z_0$  at  $t = 0$  to intercept with an effective range  $E_r \geq 0$  a moving target at

$t_m$ , i.e.,  $\|(x(t_m), y(t_m)) - E(t_m)\| \leq E_r$ .

## 2.2 Preliminary analysis of piecewise-linear programs

Note that the developed method is related to the  $\sin, \cos$  trigonometric function constraints and modulo operations (in essence, a mixed integer). Thus, the derivatives cannot be determined and cannot be directly tackled using any nonlinear continuous solvers. Consequently, we need to specifically tackle these functions. Piecewise-linear (PWL) approximation aims to provide surrogate curves by using small line segments, where the tolerance error is related to the length and the breakpoints of these line segments. Figure 3 illustrates the concept of piecewise-linear approximation for the function ‘sin’ in the range  $[0, 2\pi]$ . It shows the linear approximation of the  $y = \sin x$  function in an interval with multiple line segments (here, the segment number is 7).



**Figure 3** Coarse illustration of piecewise-linear approximation of  $\sin x$  for interval  $[0, 2\pi]$  with piece length  $2\pi/7 \approx 0.9$ , enabling better visualization of the approximate error

The  $n$ -breakpoint piecewise-linear relationship  $y = PWL(x, ptx, pty)$  between variables  $x$  and  $y$ , where  $ptx, pty$  are the  $x, y$  coordinates of the breakpoints, can be constrained as follows (Gerdt et al., 2021):

$$\begin{aligned} z_1 &\leq y_1 \\ z_i &\leq y_{i-1} + y_i, \quad \forall i = 2, 3, \dots, n-1 \\ z_n &\leq y_{n-1} \\ y_1 + y_2 + \dots + y_{n-1} &= 1 \\ z_1 + z_2 + \dots + z_n &= 1 \\ x &= z_1 \cdot ptx_1 + z_2 \cdot ptx_2 + \dots + z_n \cdot ptx_n \\ y &= z_1 \cdot pty_1 + z_2 \cdot pty_2 + \dots + z_n \cdot pty_n \\ y_i &\in \{0, 1\}, \quad \forall i = 1, 2, \dots, n-1 \\ z_i &\geq 0, \quad \forall i = 1, 2, \dots, n \end{aligned}$$

where  $y_i$  can only take binary values, and the sum of the values should equal unity, meaning that only one segment  $y_i$  will be 1 (selected). Additionally, nonnegative variables  $z_i$  are used to form a convex combination of the two end-

points of the selected segments, i.e., a point between the two endpoints of the selected segment. The  $z_i, y_i$  relationships ensure that only two  $z_i$ s are nonzero, and their sum equals unity.

Solvers such as Gurobi will automatically transform the  $y = PWL(x, ptx, pty)$  constraints in the backend. Gurobi tackles these mixed-integer piecewise-linear constraints via branch-and-bound or similar techniques to find the best values for the auxiliary binary and real-number decision variables and thus the best  $x$  and  $y$ , to achieve the optimal objective. Analysis of the approximate error is essential for guaranteeing the objective value herein. Generally, this analysis is difficult, but it was successfully derived herein.

## 3 QG-PWL approach

Aided by the piecewise-linear approximation, modular operation treatment, and the big-M strategy, we constrained each of the six path types as a mixed-integer piecewise-linear program. We consider six path types because previous work for RDP has proven that six types are sufficient, and in some simple motion cases, even 4 types with  $\{LS, RS, LR, RL\}$  are sufficient. The 6-type set is a superset of the 4-type set, and as our experiments have shown, the six types are necessary and sufficient. Each  $t_m$  is solved using the built-in branch-and-bound strategy of Gurobi. Thus, the minimum intercept time among the six types can be determined.

### 3.1 Objectives and variables

Let the six pathtypes comprise a set  $PT = \{LSL, RSR, LSR, RSL, LRL, RLR\}$ . We can enumerate each path type as a PWL program, listing and transforming its constraints, and optimizing the objective of the program. Let  $t_i, p_i, q_i \geq 0$  be the length of the three segments of path type  $i \in PT$ . The program for path type  $i$  has the objective:

$$\min t_i + p_i + q_i + b_i \times M \tag{4}$$

with rest constraints. Here,  $M$  is a large value (e.g., 300) of the big-M strategy and  $b_i$  is a binary decision variable related to the feasibility ( $b_i = 0$ )/infeasibility ( $b_i = 1$ ) of the mathematical program under path type  $i$ . If a certain path type is not interceptable, a value of unity can be inferred for  $b_i$ ; thus, the objective will be quite large. For simplicity, we omitted the subscript  $i$  for  $t, p, q$  in the following discussion.

The common constraints for each of the 6 path types are presented here, whereas the individual constraints are presented within the corresponding subsections: Because the pursuer can explode with a certain effective range greater than or equal to 0, each path type (i.e., each program) has a first constraint:

$$(T_x - x_f) \times (T_x - x_f) + (T_y - y_f) \times (T_y - y_f) \leq E_r \times E_r \tag{5}$$

where the final position of the target  $(T_x, T_y)$  is related to the intercept time  $t_m$  and the velocity of the target. Thus,

$$(T_x, T_y) = E(t_m) \tag{6}$$

Where the intercept time  $t_m$  is the optimization variable.

For the rest constraints of each path type, the pursuer is constrained to sequentially turn left/turn right/move straight; thus, the changing points should be constrained via  $t, p, q$ . Previous research was hindered by constraints involving trigonometric sine and cosine functions and the modular operator, whereas in this study, we succeeded in breaking through the two hindrances by using the PWL approximation and mixed integers. The piecewise-linear constraints of the sine and cosine functions are expressed as follows:

$$\begin{aligned} \sin[\theta] &= \text{PWL}(\theta, ptx, ptys) \\ \cos[\theta] &= \text{PWL}(\theta, ptx, ptyc) \end{aligned}$$

where the PWL constraints are illustrated in Section 2.

Note that  $\sin[\cdot]$  ( $\cos[\cdot]$ ) uses piecewise-linear approximation, compared to the common notation using brackets. The  $ptx$  represents  $\theta$  for the equally spaced breakpoints ( $0 \leq ptx < 2\pi$ ), and  $ptys$  and  $ptyc$  represent the corresponding values of  $\sin(ptx)$  and  $\cos(ptx)$ . Other radians outside the interval  $0 \leq ptx < 2\pi$  can be transformed to this interval using an integer variable, as demonstrated hereinafter.

The individualized extra constraints, listed according to the easy-to-hard LSL(RSR)-LSR(RSL)-RLR(LRL) order, are as follows:

### 3.2 Rest constraints for LSL path type

The pursuer moved left, then straight, then left, respectively, corresponding to  $t, p, q$  meters to reach the position  $(x_f, y_f)$ . This means:

$$L_q(S_p(L_t(x_0, y_0, \alpha))) = (x_f, y_f, \beta) \tag{7}$$

where  $(x_0, y_0, \alpha)$  are all known (without losing generality, the configuration of the pursuer at the initial time  $t = 0$  is assumed to be  $(0, 0, \pi/2)$ ); the final position  $(x_f, y_f)$  of the pursuer will be constrained by the intercept time  $t_m$  and the  $E(t)$  of the target. Define  $d_x, d_y$  as  $d_x = x_f - x_0, d_y = y_f - y_0$ .

As introduced in the preliminaries (Eq. (2)), expanding the corresponding movement (Eq. (7)) yields the rest constraints of the LSL path type optimization problem:

$$\begin{aligned} p\cos(\alpha + t) - \sin \alpha + \sin \beta &= d_x = x_f - x_0 \\ p\sin(\alpha + t) + \cos \alpha - \cos \beta &= d_y = y_f - y_0 \\ \alpha + t + q &= \beta \pmod{2\pi} \end{aligned} \tag{8}$$

The notations indicate that the curvy sine and cosine functions are not tractable in Gurobi; thus, we have to approximate these trigonometric  $\cos(\alpha + t), \sin(\alpha + t), \cos \beta$  with  $\cos[\alpha + t], \sin[\alpha + t], \cos[\beta]$  functions in the constraints, where  $[\cdot]$  indicates that the PWL constraints ( $\cos(\alpha)$ ) can be calculated using the analytic algorithm because  $\alpha$  is known beforehand. The approximate constraints are:

$$\begin{aligned} p\cos[\alpha + t] - \sin \alpha + \sin[\beta] &= d_x = x_f - x_0 \\ p\sin[\alpha + t] + \cos \alpha - \cos[\beta] &= d_y = y_f - y_0 \end{aligned} \tag{9}$$

The last line in Eq. (8):  $\alpha + t + q = \beta \pmod{2\pi}$  cannot be handled directly to fit into solvers; however, we successfully transform it using the auxiliary integer variable  $K_{\text{LSL}}$ , such that  $\alpha + t + q - K_{\text{LSL}} \times (2\pi) = \beta$ . Hence,

$$\begin{aligned} \alpha + t + q - 2\pi < K_{\text{LSL}} \times (2\pi) &\leq \alpha + t + q \\ \beta &= \alpha + t + q - K_{\text{LSL}} \times (2\pi) \end{aligned} \tag{10}$$

Combining Eqs. (4), (5), (6), (9), (10), the optimization program with the LSL path type can be solved (according to a previous study (Shkel and Lumelsky, 2001), the  $b_{\text{LSL}}$  can be safely set to a constant value of nil.

### 3.3 Rest constraints for RSR path type

In the QG-PWL approach, the minimum  $t + p + q$  is determined for each of the six path types. To optimize the RSR type, we expand

$$R_q(S_p(R_t(x_0, y_0, \alpha))) = (x_f, y_f, \beta) \tag{11}$$

The rest of the constraints are similar to those of the LSL. Eqs. (4), (5), (6), (11) represent the optimization program for the RSR path type.

### 3.4 Rest constraints for LSR path type

After expanding  $R_q(S_p(L_t(x_0, y_0, \alpha))) = (x_f, y_f, \beta)$ , we obtain the corresponding PWL constraints:

$$\begin{aligned} p\cos[\alpha + t] + 2\sin[\alpha + t] - \sin \alpha - \sin[\beta] &= d_x \\ p\sin[\alpha + t] - 2\cos[\alpha + t] + \cos \alpha + \cos[\beta] &= d_y \\ \alpha + t - q &= \beta \pmod{2\pi} \end{aligned} \tag{12}$$

To estimate  $b_{\text{LSR}}$ , define a discriminative  $\text{term}_{\text{LSR}} = -2 + d_x \times d_x + d_y \times d_y + 2 * \cos[\alpha - \beta] + 2 \times d_x \times (\sin \alpha + \sin[\beta]) + 2 \times d_y \times (-\cos \alpha - \cos[\beta])$ . When  $\text{term}_{\text{LSR}} < 0$ , there is no intercept solution for LSR. Thus, the infeasibility to  $b_{\text{LSR}} = 1$  and feasibility to  $b_{\text{LSR}} = 0$  can be mapped as indicated below by applying the big-M strategy:

$$\begin{aligned} \text{term}_{\text{LSR}} - (1 - b_{\text{LSR}}) \times M &< 0 \\ -\text{term}_{\text{LSR}} - b_{\text{LSR}} \times M &\leq 0 \end{aligned} \tag{13}$$

Considering this infeasibility possibility, the first two lines in Eq. 12 are altered to:

$$\begin{aligned}
 & p\cos[\alpha + t] + 2\sin[\alpha + t] - \sin\alpha - \sin[\beta] - \\
 & \quad b_{\text{LSR}} \times M \leq d_x, \\
 & d_x - b_{\text{LSR}} \times M \leq p\cos[\alpha + t] + 2\sin[\alpha + t] - \\
 & \quad \sin\alpha - \sin[\beta], \\
 & p\sin[\alpha + t] - 2\cos[\alpha + t] + \cos\alpha + \cos[\beta] - \\
 & \quad b_{\text{LSR}} \times M \leq d_y, \\
 & d_y - b_{\text{LSR}} \times M \leq p\sin[\alpha + t] - 2\cos[\alpha + t] + \\
 & \quad \cos\alpha + \cos[\beta]
 \end{aligned} \tag{14}$$

When  $b_{\text{LSR}} = 0$ , the four inequalities become the two original equality constraints. However, if  $b_{\text{LSR}} = 1$ , the four equations are naturally satisfied for the given  $t, p, q$ . Because  $b_{\text{LSR}} \times M$  is quite large, the optimization objective will also be quite large; thus, this path type will not be the optimal intercept choice.

### 3.5 Rest constraints for RSL path type

The rest constraints are determined in a manner similar to the method used for the LSR path type.

### 3.6 Rest constraints for RLR path type

Expanding  $R_q(L_p(R_t(x_0, y_0, \alpha))) = (x_p, y_p, \beta)$ , we obtain the corresponding PWL equations:

$$\begin{aligned}
 & 2\sin[\alpha - t + p] - 2\sin[\alpha - t] = d_x - \sin\alpha + \sin\beta, \\
 & -2\cos[\alpha - t + p] + 2\cos[\alpha - t] = d_y + \cos\alpha - \cos\beta, \tag{15} \\
 & \alpha - t + p - q = \beta \pmod{\pi}
 \end{aligned}$$

We can derive a discriminative term from the first two lines by applying the sum-to-product identities in trigonometry, as follows:

$$\begin{aligned}
 \text{term}_{\text{RLR}} = & \frac{1}{8} \left( 6 - d_x * d_x - d_y * d_y + 2\cos[\alpha - \beta] + \right. \\
 & \left. 2d_x(\sin\alpha - \sin[\beta]) + 2*d_y*(-\cos\alpha + \cos[\beta]) \right) \tag{16}
 \end{aligned}$$

When  $\text{term}_{\text{RLR}} \times \text{term}_{\text{RLR}} \leq 1$  then  $b_{\text{RLR}} = 0$ , otherwise,  $b_{\text{RLR}} = 1$ . The mapping can be constructed as follows, according to the big-M strategy:

$$\begin{aligned}
 & (\text{term}_{\text{RLR}} \times \text{term}_{\text{RLR}} - 1) - b_{\text{RLR}} \times M \leq 0, \\
 & -(\text{term}_{\text{RLR}} \times \text{term}_{\text{RLR}} - 1) - (1 - b_{\text{RLR}}) \times M < 0 \tag{17}
 \end{aligned}$$

Considering this infeasibility possibility, the first two lines in Eq. (15) are altered to:

$$d_y - b_{\text{RLR}} \times M \leq p\sin[\alpha + t] - 2\cos[\alpha + t] + \cos\alpha + \cos\beta$$

$$\begin{aligned}
 & p\cos[\alpha + t] + 2\sin[\alpha + t] - \sin\alpha - \sin\beta - b_{\text{RLR}} \times M \leq d_x, \\
 & d_x - b_{\text{RLR}} \times M \leq p\cos[\alpha + t] + 2\sin[\alpha + t] - \sin\alpha - \sin\beta, \\
 & p\sin[\alpha + t] - 2\cos[\alpha + t] + \cos\alpha + \cos\beta - b_{\text{RLR}} \times M \leq d_y, \\
 & d_y - b_{\text{RLR}} \times M \leq p\sin[\alpha + t] - 2\cos[\alpha + t] + \\
 & \quad \cos\alpha + \cos\alpha + \cos\beta
 \end{aligned} \tag{18}$$

### 3.7 Rest constraints for LRL path type

The rest constraints are determined in a manner similar to the method used for the RLR path type.

## 4 Experimental validations

Zheng et al. (2021) reported four classic examples demonstrating their bisectional approach for the original MTIP. Herein, to clearly validate the performance and demonstrate the accuracy and efficiency, the QG-PWL approach is applied to the same four examples at the first stage. Thereafter, we apply the approach to extended scenarios, including interception under different effective ranges, uniformly accelerated target motion, and complicated target motion such as wave motion. The classic examples and extended scenarios were all normalized so that the speed of the pursuer is 1 m/s and the turning radius is 1 m. The unit of the variables can be omitted. Besides comparing the QG-PWL data to the results obtained with the analytic exact algorithm, comparative experiments were also conducted using a GA, the Geatpy package, when there were no analytic exact algorithms. The performance of the GA was compared with that of QG-PWL to assess both the convergence speed and the optimality of the obtained solutions.

The computational experiments were conducted on a PC with an Intel(R) Core(TM) i7-8750H CPU@2.20 GHz processor, 16 GB RAM, with GUROBI 9.0 using Windows 10. The breakpoints were equally spaced with a piecewise length of 0.01. The big-M value was 300. The time limit for the solver was set to 1 800 s, which was never reached because all of the solutions were fully computed within 3 s.

The GA was utilized with the aim of solving an optimization problem where a Dubins vehicle, starting from a specific position, must intercept a moving target. The GA works by iteratively evolving a population of potential solutions, where each solution is represented as a chromosome encoded in Gray code. The GA provided in the Geatpy package was implemented using the exact default settings for the GA as the official guide documentation, which included the population size, crossover probability, mutation rate, and other parameters for the specific problem at hand. The detailed description of the GA is as follows:

- **Objective Function of GA:** The objective function of the GA calculates the intercept time and the distance

between the vehicle and the target after a given time, considering the initial orientation and position of the vehicle. Specifically, the algorithm seeks to minimize the time taken (“*t*”, “*p*”, “*q*”) plus the distance (“dis”) between the vehicle and the target. A penalty factor is applied to the “dis” when the distance is larger than the intercept range. This first reflects a feasible intercept, after which the intercept time is optimized, as required by the developed EMTIP.

- **GA Variable and Encoding Setup.** Three variables are optimized: time “*t*”, “*p*”, and “*q*”, with specific bounds on each. The variables are continuous, and the GA uses a Gray encoding scheme with a precision of 6 decimal places to represent the variables. The encoding and other parameters are set up to allow the GA to efficiently search the solution space. Note that the GA was not required to optimize the discrete path type (the best path type was provided to the GA), and it only needed to optimize the time. Despite removing these requirements for the GA, the developed QG-PWL outperformed the GA.

- **Genetic Algorithm Operations**

**Population Initialization:** A population of 20 individuals is initialized, each representing a possible solution.

**Selection:** The GA uses stochastic universal sampling (SUS) to select individuals for reproduction based on their fitness, which is directly related to the value of the objective function.

**Crossover and Mutation:** The selected individuals undergo two-point crossover with a high probability (90%) and mutation with a probability that depends on the chromosome length (in total 1 bit mutation). These operations introduce new genetic variations and cover the solution space.

**Fitness Evaluation:** After crossover and mutation, the offspring are evaluated, and the best solution is carried over to the next generation.

- **Evolution and Convergence:** The GA runs for 100 generations. In each generation, the algorithm tracks the average and best values of the objective function across the population. The algorithm converges when the best solution stabilizes across generations. The GA computation cutoff time is set as the QG-PWL computation cutoff time.

### 4.1 Four basic classic examples

In the four examples, the start configurations of the pursuer are all at  $(0, 0, \pi/2)$ . The detailed information and comparison of the performance of the developed system with that of previous algorithms in relation to the four examples are summarized in Table 2.

The first two rows are the known information for the target. As shown in the row indicating the intercept time, the PWL-based intercept time is nearly the same as in the previous study, indicating the effectiveness of QG-PWL. All intercept path types are LSL types. In the next row, *t, p, q* indicates the first, second, and third segments of the pursuer. The data show that the value of *q* is almost 0, as also noted in previous articles (Ding et al., 2019; Chen et al., 2023; Zheng et al., 2021).  $x_f, y_f$  is the terminal position of the pursuer, which is related to *t, p, q*. The terminal position of the target is not shown. The position can be calculated using the intercept time and target motion; however, we found that the distances between the terminal positions of the pursuer and the target are almost 0. Although the time for case D was not explicitly specified in previous studies, the resulting terminal positions of the pursuer calculated using the intercept time herein are very close to the reported positions, showing the correctness of QG-PWL.

Upon completion, the GA outputs the optimal solution, including the decision variables (“*t*”, “*p*”, and “*q*”), and reports the total intercept time. The data show that the GA is also highly accurate, where the calculated intercept times are quite close to the analytic solutions. In the following scenario, we compare the QG-PWL with GA, where no analytic algorithm is present.

The results obtained with QG-PWL were quite close to the numerical solutions (Zheng et al., 2021) when the effective range was zero. Hereinafter, we demonstrate that QG-PWL can also be applied to more complicated scenarios.

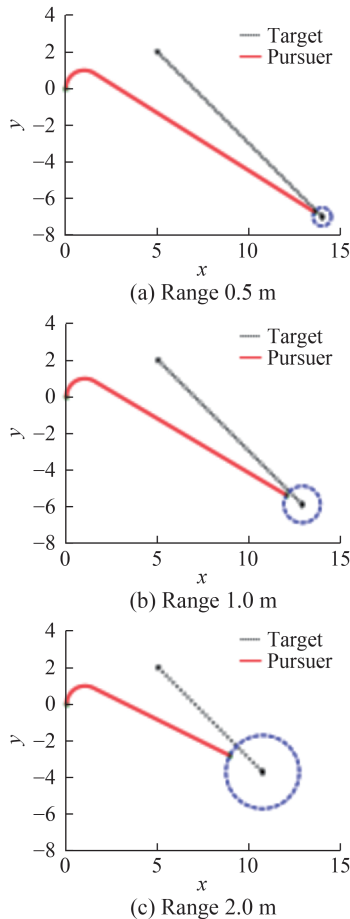
### 4.2 Interception under different effective ranges

As indicated in the last subsection, the effective range  $E_r$  can be viewed as set to 0. Suppose the Dubins pursuer

**Table 2** Motion information for each classic case and performance

Information	Case A	Case B	Case C	Case D
$x_0, y_0$	5, 2	1.2, 0	-3, 0.8	$-\frac{1 + \sqrt{3}}{2}, \frac{\sqrt{3}}{2}$
$v_x, v_y$	0.55, -0.55	-0.1, -0.1	0.15, 0	$\frac{\sqrt{3}}{4\pi}, 0$
Intercept time (time reported by Zheng et al. (2021), for GA)	18.45 (18.45, 18.46)	5.43 (5.43, 5.45)	3.15 (3.15, 3.15)	6.28 (not given, 6.29)
<i>t, p, q</i>	2.15, 16.29, 0.00	0.21, 5.22, 0.00	1.71, 1.40, 0.00	1.05, 5.23, 0.00
$x_f, y_f$	15.14, 8.15	0.66, 0.54	-2.53, 0.80	-0.50, 0.86
Distance	0.0008	0.002	0.0097	0.004

can intercept the target with an effective range greater than 0, then QG-PWL can also be applied without modification. The performance of QG-PWL was tested using case A with three different effective ranges: 0.5, 1.0, and 2.0 m. Figure 4 presents the interception results, with the detailed information summarized in Table 3.



**Figure 4** Interception performance under different pursuer effective ranges (0.5, 1.0 and 2.0 m)

Because the GA was sufficiently accurate in the previous scenario, we can compare the intercept time to that obtained with the GA in this scenario and to that of case A; the data are presented in Table 2. The figure shows that with a larger effective range, the pursuer can more quickly intercept the target. From the table, it can be seen that the distance between the pursuer and the target, i.e., the distance between  $(x_p, y_p)$  and  $(T_x, T_y)$ , is nearly identical to

the prescribed value for the effective range.

In this experiment, QG-PWL consistently outperformed the GA, particularly in terms of accuracy. This difference is particularly noticeable in scenarios involving effective ranges, where QG-PWL’s ability to effectively handle non-linearities and constraints was highlighted.

### 4.3 Target with uniformly accelerating motion

This subsection tests the performance of QG-PWL in scenarios where the target moves with constant acceleration. When the acceleration is in the same line as the target’s initial velocity, the target will speed up (positive acceleration) or slow down. In contrast, when the acceleration is not co-linear to the target’s initial velocity, the target will move in projectile motion.

Figure 5 shows the interception of a target moving with different motions at constant acceleration. The basic initial setting corresponds to case A, where the effective range is 0.5 m and the target accelerates in three different modes: (acceleration in the x-axis  $a_x = 0.001$ , acceleration in the y-axis  $a_y = -0.001$ ), where the target will speed up; and  $(a_x = -0.001, a_y = 0.001)$ , where it slows down;  $(a_x = 0, a_y = -0.001)$ , corresponding to the projectile. The acceleration is controlled to a moderate value because rapid acceleration may result in a large target velocity, where the pursuer with a velocity of 1 m/s cannot catch up. The color map in the subfigure shows the procedure status, where a lighter color indicates a later procedure: the color of the target’s path changes non-uniformly, indicating acceleration. The speed-down subfigure has a smaller intercept time than the speed-up subfigure. QG-PWL has the general ability to tackle different and complicated motions other than uniformly rectilinear motions.

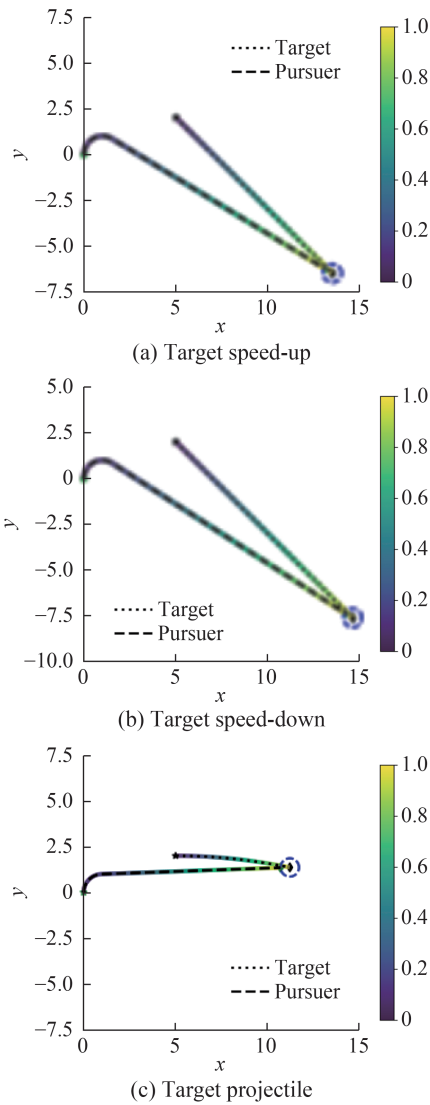
To analyze the control  $u$  and the velocity of the pursuer and target over time, the optimal  $u$  and the corresponding velocities are depicted in Figure 6. As shown in the figure, the optimal path of the pursuer is RS ( $u = -1$  for the right turn); the pursuer’s velocity is always 1 m/s; the target’s velocity, according to Pythagorean theorem, is  $\sqrt{vx^2 + (vy + a_y \cdot t)^2}$ , which resembles a straight line in the figure owing to the small  $a_y$ .

### 4.4 Target with complicated motion

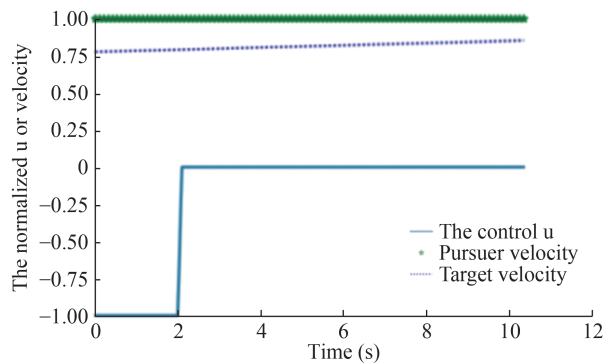
In this subsection, the process of intercepting a target

**Table 3** Detailed information on interception under different effective ranges

Information	0.5 m range	1.0 m range	2.0 m range
Intercept time using the GA result	16.36 (16.58)	14.31 (14.90)	10.36 (11.02)
$(t, p, q)$	2.13, 14.23, 0.00	2.11, 12.20, 0.01	2.03, 8.33, 0.00
Pursuer (target) terminal position	13.58, -6.73 (14.00, -7.00)	12.01, -5.36(12.87, -5.87)	8.91, -2.80 (10.70, -3.70)
terminal distance	0.500	1.002	2.002



**Figure 5** (Best viewed in color) Interception of target moving with different motions at constant acceleration



**Figure 6** Control  $u$  (rad/s) and velocity (m/s) for the pursuer and the target of the projectile motion in Figure 5

with time-varying acceleration, and thus time-varying velocity, is evaluated. The target moves along a sinusoidal curve, where the motion equation for the target with respect to

time ( $t$ ) is  $T_x = x_0 + 0.1 \times t, T_y = y_0 + \sin(x_0 + 0.1 \times t)$

Let  $(x_0, y_0) = (0, 0)$ , where the effective range is set to 0 m. We enumerated all six path types on this target motion. The resulting intercept plot is presented in Figure 7.

The intercept time for the six path types is 8.03 (LSL), 4.91 (LSR), 8.65 (RSL), 5.49 (RSR), 8.65 (RLR), and 4.91 (LRL).

A solution was found for each of the six path types, where the intercept time (4.91) was shortest for (b) LSR and (f) LRL. Further evaluation of the length of the pursuer’s three segments indicated that the S-segment in (b) LSR is zero, and the last L-segment in (f) LRL is zero; thus, the two different path types vanish to the same intercept path. These findings prove the correctness and general applicability of QG-PWL.

### 4.5 Quality analysis

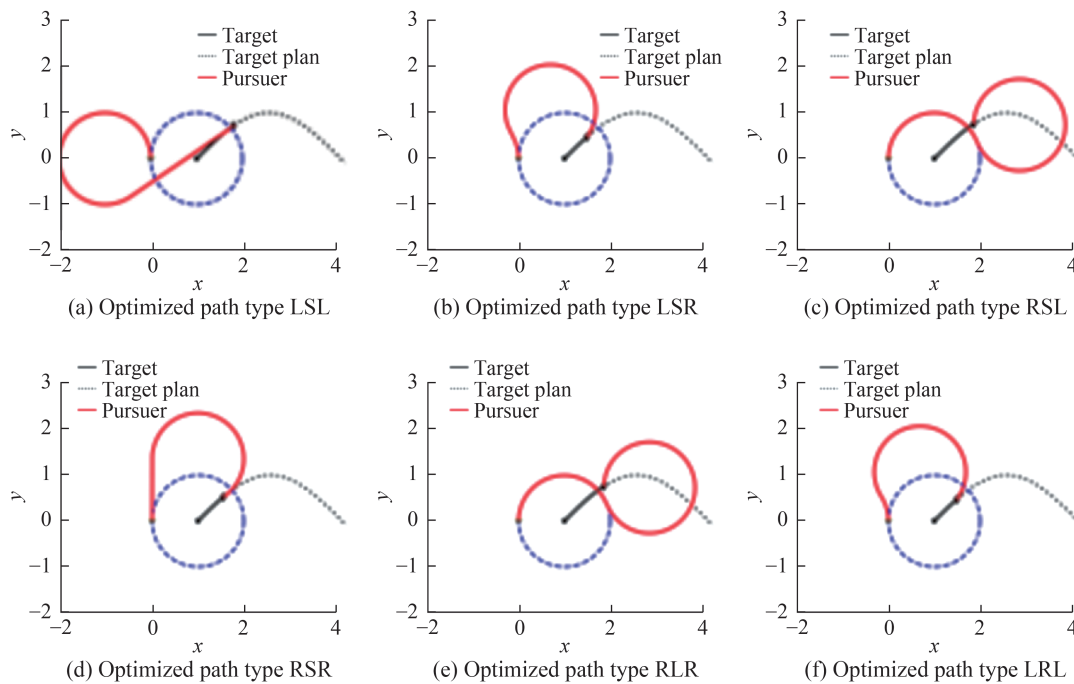
In this subsection, we theoretically analyze the approximation guarantee of the PWL path length in the LS path type with  $E_r = 0$ , where the length of the third segment  $q$  is usually zero (Zheng et al., 2021) in the MTIP. The error in the simple case can enhance our understanding of the developed approach (and a full analysis of EMTIP can be left for future studies). In this section, we prove that the optimal length and the piecewise-linear computed length have an absolute error no larger than  $h^2 / (4\sqrt{2})$ , where  $h$  is the piece length. If we set  $h = 0.01$ , then the length error may be  $1 \times 10^{-4}$ . Suppose the optimal length is 20, then the calculated approximate length may be 20.0001.

We assume a path planning case from  $(x, y, \alpha)$  to  $(x', y', \beta)$ . Suppose that the setting is interceptable for LS. We also do not consider floating-point numerical issues because they can be assumed to be sufficiently precise. For the analytical/mathematical computation, the optimization problem is as follows:

$$\begin{cases} \min t + p \\ \text{s.t.} \\ x' = x + v_x * t_m \\ y' = y + v_y * t_m \\ p \cos(\alpha + t) - \sin \alpha + \sin \beta = x' - x \triangleq d_x \\ p \sin(\alpha + t) + \cos \alpha - \cos \beta = y' - y \triangleq d_y \\ \beta = \alpha + t \pmod{2\pi} \end{cases} \quad (19)$$

Assume that this program has an optimal analytical intercept time  $t_m^*$ . Also, assume that the PWL program-based intercept time precisely corresponds to the ground truth  $t_m^*$ : we aim to analyze the path-length error between the PWL and analytic methods with respect to the piecewise length  $h$ . The analysis is performed as follows:

Substituting  $\beta = \alpha + t \pmod{2\pi}$  and using the amplitude-phase equation, then:



**Figure 7** Interception of target with complicated motion in sinusoidal trajectories.

$$\begin{aligned}
 p \cos(\alpha + t) - \sin \alpha + \sin(\alpha + t) &= d_x, \\
 \Rightarrow \sqrt{p^2 + 1} \cos(\alpha + t - \phi) &= d_x + \sin \alpha, \\
 p \sin(\alpha + t) + \cos \alpha - \cos(\alpha + t) &= d_y, \\
 \Rightarrow \sqrt{p^2 + 1} \sin(\alpha + t - \phi) &= d_y - \cos \alpha
 \end{aligned} \tag{20}$$

where  $\cos \phi = p / \sqrt{p^2 + 1}$ . By squaring and summing the above two equations, we get  $p^2 = (d_x + \sin \alpha)^2 + (d_y - \cos \alpha)^2 - 1$ ,  $t = \text{atan2}(d_y - \cos \alpha, d_x + \sin \alpha) + \phi - \alpha$ . The approximation error of  $\sin, \cos$  yields an error of  $p$ .

To derive the absolute error of the piecewise-linear-based  $t + p$ , we need the following lemma.

**Lemma 1.** Let  $f(x) \in C^{n+1}[a, b]$  (which indicates that  $f(x)$  is continuously differentiable up to order  $n + 1$  on  $[a, b]$ ), and let the interpolation points be  $a \leq x_0 < x_1 < \dots < x_n \leq b$ . Thus, for the interpolation polynomial  $L_n(x)$ , for any  $x \in [x_i, x_{i+1}]$ , we have

$$\begin{aligned}
 |R_n(x)| &= |f(x) - L_n(x)| = \\
 & \left| \frac{f^{(n+1)}(\xi)}{(n+1)!} (x - x_i)(x - x_{i+1}) \right|
 \end{aligned}$$

In this study,  $n = 1$  indicates linear interpolation in the Lagrangian interpolation polynomial  $L_n$ . With this lemma, we can now derive the absolute error. Define the difference between curvy and piecewise-linear sine as  $\text{dsin} x = |\sin(x) - \sin[x]|$ .

Thus, according to the lemma, it holds that

$$\begin{aligned}
 \text{dsin} x &\approx |\sin x - L_1(x)| \leq \frac{|f''(\xi_1)|}{2!} |(x - x_i)(x - x_{i+1})| \leq \\
 & \frac{1}{2} \cdot |f''(\xi_1)| \frac{h^2}{4} = \frac{1}{2} \cdot \frac{h^2}{4} \\
 \text{dcos} x &\leq \frac{1}{2} |g''(\xi_1)| \frac{h^2}{4} = \frac{1}{2} \cdot \frac{h^2}{4}
 \end{aligned}$$

where  $h$  is the piece length  $x_{i+1} - x_i$ , i.e., the distance  $x$  between two consecutive breakpoints; the rightmost  $\leq$  has applied the arithmetic-mean–geometric-mean inequality.

$S$  denotes the square of  $p$ , and  $K_1 = d_x + \sin \alpha, K_2 = d_y - \cos \alpha$ ; thus, according to the Eq. (21), we have:

$$\begin{aligned}
 \text{d}S &= 2(d_x + \cos \alpha) \text{dsin} \alpha + 2(d_y - \cos \alpha) \text{dcos} \alpha \leq \\
 2k_1 \frac{h^2}{8} + 2k_2 \frac{h^2}{8} &= (k_1 + k_2) h^2 / 4
 \end{aligned}$$

$$\begin{aligned}
 \text{Therefore, } \text{d}p &= \frac{\text{d}S}{2p} \leq \frac{((k_1 + k_2)h^2/4)}{2\sqrt{k_1^2 + k_2^2 - 1}} \approx \frac{((k_1 + k_2)h^2/4)}{2\sqrt{k_1^2 + k_2^2}} \leq \\
 2\sqrt{\frac{k_1^2 + k_2^2}{2}} \frac{h^2}{4} &= \frac{h^2}{4\sqrt{2}}.
 \end{aligned}$$

where the term after “ $\Rightarrow$ ” comes from the relation  $S = p^2$ , which is  $\text{d}S = 2p \text{d}p$ . The last  $\leq$  applied arithmetic-mean–quadratic-mean inequality.

Finally,

$$d(t + p) = dt + dp \leq dt + \frac{h^2}{4\sqrt{2}} =$$

$$d(\text{atan}(K_2, K_1)) + \frac{h^2}{4\sqrt{2}} = \frac{h^2}{4\sqrt{2}}$$

where  $K_1$  and  $K_2$  are unrelated to the PWL; thus, the difference error is 0. This simple theoretical analysis demonstrates that QG-PWL can achieve high accuracy with nominal errors.

## 5 Discussions

Robustness, generalizability, and potential applications The QG-PWL approach exhibits effectiveness in managing intricate path-planning situations involving curvature-constrained vehicles. The capacity to manage both discrete and continuous variables enables effective approximation of complex trigonometric and geometric constraints. This approach is applicable to multiple domains, including UAV rendezvous, UAV reconnaissance, autonomous ground vehicle navigation, and USV patrolling, where efficient mission planning with low energy consumption is essential. The approach has demonstrated feasibility for applications, effectively addressing problems within a reasonable timeframe. The characteristics of QG-PWL render it appropriate for various path-planning tasks, especially in contexts where curvature constraints, targets with effective ranges, and energy efficiency are critical. Several steps are necessary for its deployment in real-world applications. The scenario must be modeled, and the vehicle's speed should be normalized to unity. The QG-PWL determines the optimal path segments by inputting the scenario parameters into the solver GUROBI. Subsequently, a PID or MPC controller (Xu and Guedes Soares, 2023) can effectively follow the path. If the velocity remains constant, the designated path is both the shortest and the fastest.

**Idealized assumptions.** This study marks an advancement; however, it remains a modest progression, with several idealized assumptions, such as the exclusion of wind, current effects (Ghassemzadeh et al., 2024), and the hydrodynamic effects on the surface of the vehicle, with side-slip phenomena. In the Dubins path framework, side-slip is not considered because the model assumes that the vehicle maintains a fixed orientation relative to its direction of travel. This means that the vehicle is treated as a mass point and does not slide sideways; it strictly follows the path defined by the turning radius and the steering angle. The absence of environmental effects simplifies the calculations and ensures that the path remains within the constraints of the model. However, constant noise can be incorporated into the kinematic equations of the model (Eq. (1)). Thereafter, the constraints with unknown variables  $t, p, q$

can be listed and solved using the developed approach. However, for variable or unknown noises, future considerations may be needed to include more realistic scenarios, such as environmental disturbance, as well as predict-then-intercept path planning.

**Error bound.** In our current analysis, the error bound was evaluated for scenarios with a single target, with zero effective range. The behavior of the error for different settings was discussed. As the intercept time increases, the number of binary variables also increases, leading to larger demands for computational time. However, the error itself remains stable as long as the piece length is kept the same, which is a desirable attribute of QG-PWL. We have not yet considered scenarios with multiple targets and have not conducted an in-depth analysis, but this is an important direction for future work.

**More advanced meta-heuristics.** Although GAs are classic meta-heuristic methods, they offer stable and robust performance and thus remain highly competitive in solving diverse optimization problems. In this study, the GA produced results close to the analytic solutions in comparable scenarios, which validated the effectiveness of the GA and also validated the effectiveness of QG-PWL. Generally, no single classic heuristic optimization method universally outperforms others across all problems. However, we concede that exploring other algorithms, such as Particle Swarm Optimization (PSO), Simulated Annealing (SA), and Ant Colony Optimization (ACO), especially incorporating problem-specific knowledge into the algorithm designs, could enrich the study, particularly for future research on more complex problems, such as multi-target interception.

## 6 Conclusions

We investigated the EMTIP for a Dubins vehicle (e.g., USV) pursuing a target moving in straight or curvy trajectories with the intent to intercept. The achievements of this approach are that the velocity of the target does not have to be constant, and the pursuer's effective range can be non-zero, which has not been evaluated in previous studies. The comparisons with analytic solutions and GA prove the advantages of QG-PWL; the approximate error was derived. We used a mixed-integer piecewise-linear program for the otherwise intractable trigonometric functions and the periodic headings. Moreover, various linearization techniques were employed, such as the infeasibility transformation using the big-M strategy. The analysis results of the pursuer's length show that nearly all situations have a small and bounded tolerance to the analytical/theoretical solutions.

In the future, we intend to consider distributed algorithms for enabling multiple pursuers to jointly intercept a target or targets with an unknown motion that needs to be predicted,

or consider interception in environments with obstacles. In addition, the EMTIP can be extended to three-dimensional scenarios. This method can potentially be extended to various similar complicated situations or applications.

Supplementary Information. GIFs are provided for Figure 4, along with the trajectory of the Dubins vehicle for intercepting different targets.

**Funding** This work is partly supported by the National Natural Science Foundation of China (Grant No. 62306325)

**Competing interests** The authors have no competing interests to declare that are relevant to the content of this article.

## References

- Bui XN, Boissonnat JD, Soueres P, Laumond JP (1994) Shortest Path Synthesis for Dubins Non-Holonomic Robot. Proceedings of the 1994 Ieee International Conference on Robotics and Automation San Diego CA USA: 2-7. <https://doi.org/10.1109/ROBOT.1994.351019>
- Buzikov ME, Galyaev AA (2022) Minimum-Time Lateral Interception of a Moving Target by a Dubins Car. *Automatica* 135: 109968. <https://doi.org/10.1016/j.automatica.2021.109968>
- Chen Z, Shi MT (2019) Shortest Dubins Paths Through Three Points. *Automatica* 105: 368-375. <https://doi.org/10.1016/j.automatica.2019.04.007>
- Chen Z, Wang K, Shi H (2023) Elongation of Curvature-Bounded Path. *Automatica* 151: 110936. <https://doi.org/10.1016/j.automatica.2023.110936>
- De Palma D, Parlangei G (2022) Shortest Path Type Classification for Real-Time Three-Points Dubins Problems. 2022 30th Mediterranean Conference on Control and Automation (Med), Athens, Greece, 520-525. <https://doi.org/10.1109/MED54222.2022.9837205>
- Ding Y, Xin B, Chen J (2019) Curvature-Constrained Path Elongation with Expected Length for Dubins Vehicle. *Automatica* 108: 108495. <https://doi.org/10.1016/j.automatica.2019.108495>
- Dubins LE (1957) On Curves of Minimal Length with a Constraint on Average Curvature, and with Prescribed Initial and Terminal Positions and Tangents. *American Journal of Mathematics* 79(3): 497-516. <https://doi.org/10.2307/2372560>
- Fan J, Zhang X, Zou Y (2023) Hierarchical Path Planner for Unknown Space Exploration Using Reinforcement Learning-Based Intelligent Frontier Selection. *Expert Systems with Applications* 230: 120630. <https://doi.org/10.1016/j.eswa.2023.120630>
- Fu J, Sun G, Liu J, Yao W, Wu L (2023) On Hierarchical Multi-Uav Dubins Traveling Salesman Problem Paths in a Complex Obstacle Environment. *IEEE Transactions on Cybernetics* 54: 1-13. <https://doi.org/10.1109/TCYB.2023.3265926>
- Gerds M, Rogovs S, Valenti G (2021) A Piecewise Linearization Algorithm for Solving Minlp in Intersection Management. *VEHITS, Virtual, Online*, 438-445. <https://doi.org/10.5220/0010437104380445>
- Ghassemzadeh A, Xu H, Guedes Soares C (2024) Optimal Path Following Controller Design Based on Linear Quadratic Regulator for Underactuated Ships in Varying Wave and Wind Conditions. *Journal of Marine Science and Application* 23 (2): 927-946. <https://doi.org/10.1007/s11804-024-00540-0>
- Gopalan A, Ratnoo A, Ghose D (2016) Time-Optimal Guidance for Lateral Interception of Moving Targets. *Journal of Guidance, Control, and Dynamics* 39(3): 510-525. <https://doi.org/10.2514/1.G001527>
- Jha B, Chen Z, Shima T (2022) Time-Optimal Dubins Trajectory for Moving Obstacle Avoidance. *Automatica* 146: 110637. <https://doi.org/10.1016/j.automatica.2022.110637>
- LaValle SM (2006) *Planning Algorithms*. New York, NY, USA: Cambridge university press. <https://doi.org/10.1017/CBO9780511546877>
- Liu L, Wang X, Yang X, Liu H, Li J, Wang P (2023) Path Planning Techniques for Mobile Robots: Review and Prospect. *Expert Systems with Applications: An International Journal* 227 (C). <https://doi.org/10.1016/j.eswa.2023.120254>
- Matveev AS, Teimoori H, Savkin AV (2011) A Method for Guidance and Control of an Autonomous Vehicle in Problems of Border Patrolling and Obstacle Avoidance. *Automatica* 47(3): 515-524. <https://doi.org/10.1016/j.automatica.2011.01.024>
- Meyer Y, Isaiiah P, Shima T (2015) On Dubins Paths to Intercept a Moving Target. *Automatica* 53: 256-263. <https://doi.org/10.1016/j.automatica.2014.12.039>
- Moon B, Sachdev S, Yuan J, Scherer S (2023) Time-Optimal Path Planning in a Constant Wind for Uncrewed Aerial Vehicles Using Dubins Set Classification. *IEEE Robotics and Automation Letters* 9(3): 1-8. <https://doi.org/10.1109/LRA.2023.3333167>
- Niu Y, Yan X, Wang Y, Niu Y (2023) Three-Dimensional Uav Path Planning Using a Novel Modified Artificial Ecosystem Optimizer. *Expert Systems with Applications* 217: 119499. <https://doi.org/10.1016/j.eswa.2022.119499>
- Ntakolia C, Moustakidis S, Siouras A (2023) Autonomous Path Planning with Obstacle Avoidance for Smart Assistive Systems. *Expert Systems with Applications* 213: 119049. <https://doi.org/10.1016/j.eswa.2022.119049>
- Rathinam S, Manyam SG, Zhang Y (2019) Near-Optimal Path Planning for a Car-Like Robot Visiting a Set of Waypoints with Field of View Constraints. *IEEE Robotics and Automation Letters* 4(2): 391-398. <https://doi.org/10.1109/LRA.2018.2890433>
- Sadeghi A, Smith SL (2016) On Efficient Computation of Shortest Dubins Paths Through Three Consecutive Points. 2016 Ieee 55th Conference on Decision and Control (Cdc), Las Vegas, NV, USA, 6010-5. <https://doi.org/10.1109/CDC.2016.7799192>
- Shima T (2011) Intercept-Angle Guidance. *Journal of Guidance, Control, and Dynamics* 34(2): 484-492. <https://doi.org/10.2514/1.51026>
- Shkel AM, Lumelsky V (2001) Classification of the Dubins Set. *Robotics and Autonomous Systems* 34(4): 179-202. [https://doi.org/10.1016/S0921-8890\(00\)00127-5](https://doi.org/10.1016/S0921-8890(00)00127-5)
- Sussmann HJ, Tang G (1991) Shortest Paths for the Reeds-Shepp Car: A Worked Out Example of the Use of Geometric Techniques in Nonlinear Optimal Control. *Rutgers Center for Systems and Control Technical Report* 10: 1-71
- Xu H, Guedes Soares C (2023) Review of Path-Following Control Systems for Maritime Autonomous Surface Ships. *Journal of Marine Science and Application* 22(2): 153-171. <https://doi.org/10.1007/s11804-023-00338-6>
- Zheng Y, Chen Z, Shao X, Zhao W (2021) Time-Optimal Guidance for Intercepting Moving Targets by Dubins Vehicles. *Automatica* 128: 109557. <https://doi.org/10.1016/j.automatica.2021.109557>