

Continuous-curvature paths for mobile robots

Emese Szádeczky-Kardoss / Bálint Kiss

Received 2009-03-26

Abstract

This paper discusses how to plan continuous-curvature paths for car-like wheeled mobile robots. The task is to generate a trajectory with upper-bounded curvature and curvature derivative. To solve this problem we use three path planning primitives, namely straight line segments, circular segments, and continuous-curvature turns (CC turns) in the path planning. We give a classification of the CC turns and we also describe the motion along different kinds of CC turns. We focus on giving computational effective formulae for real-time usage.

Keywords

car-like mobile robot · path planning · continuous-curvature

Acknowledgement

The research was partially funded by the Hungarian Science Research Fund under Grant OTKA K71762 and by the Advanced Vehicles and Vehicle Control Knowledge Center under grant RET 04/2004.

Emese Szádeczky-Kardoss

Department of Control Engineering and Information Technology, BME, Magyar Tudósok krt. 2, H-1117 Budapest, Hungary
e-mail: szadeczky@iit.bme.hu

Bálint Kiss

Department of Control Engineering and Information Technology, BME, Magyar Tudósok krt. 2, H-1117 Budapest, Hungary
e-mail: bkiss@iit.bme.hu

1 Introduction

Path planning and tracking control of car-like mobile robots become a popular research field during the last decade since a strong need emerged for fully or partly autonomous vehicles. A popular application field of the results of such research activities is represented by the assisted or fully automated parking systems [15] that car manufacturers want to offer to their clients in the near future (some pioneering systems [1] have been already introduced on major markets).

Since most of the maneuvers that are candidate for automated execution take place at low velocities, the kinematic model of the mobile robot gives a reliable approximation of the reality. The characteristics of such models is their nonholonomy and the possibility to give accurate measurement of their parameters due to their geometric nature (e.g. wheelbase distance). The models may slightly differ in the relative degree [9] of the steering input that influences the curvature of the path of the vehicle with respect to the vehicle position. This means that some models take as one of the inputs the wheel angle, whereas others take its first or sometimes second time derivative, the other input being always the longitudinal velocity of the vehicle.

There are several methods which plan the motion using such kinematic models (e.g. in a discretized configuration space with minimal number of maneuvers [2], polynomial-time algorithm for calculating the shortest path of bounded curvature (SBC) [10], using the Probabilistic RoadMap algorithm [21]). These algorithms use straight lines and circular segments usually with the minimal turning radius to build the path for the robot. These motions are optimal for robots moving in forward direction only [5] and for robots moving both in forward and backward directions [16], but the curvature is not continuous along such trajectories (i.e. the robot has to stop to reorient its front wheels between straight lines and circular segments) which is an elementary requirement for automated maneuvering especially when the velocity should not vanish along the trajectory. To meet the supplementary requirement of continuous-curvature paths, the set of curve primitives must be enriched by some additional segments (e.g. clothoid arcs [19] or segments generated by a special steering method [13]). A clothoid is a curve whose

curvature varies linearly with its arc length, hence the curvature is continuous along it.

The planning problem becomes even more involved if one also takes into account that the car-like robot can only reorient its front wheels with a finite turning velocity, hence the curvature derivative is also upper bounded. In [12] the properties of the clothoids with bounded curvature derivative are presented but the authors do not consider the minimal turning radius constraint. The existing methods (for the forward-only problem [20], for robots moving both in forward and backward directions [6]) which solve the problem of path planning with both upper bounded curvature and curvature derivative use clothoid arcs in the motion planning. The main problem with the clothoid arcs is that the displacement is calculated using Fresnel integrals which are not available in closed form. This impedes the direct real-time parametrization of the path since the value of the integrals has to be calculated off-line. The solution cited therefore suppose that the curvatures at the initial and final configurations are zero which is not always true in real situations.

In this paper we discuss how to generate continuous-curvature paths for car-like wheeled mobile robots joining arbitrary initial and final configurations with non-zero initial and final curvature. Our task is to plan a feasible trajectory which satisfies the equations of the kinematic model and the bounds on the input variables. In order to propose algorithms that can be executed in real-time we work with a set of precomputed Fresnel integral values and give an upper bound on the planning error depending on the resolution of the set on a curvature interval. The path is constructed based on a set of elementary curves, namely: straight line segments, circular segments, and continuous-curvature turns (CC turns). (The CC turn is a special clothoid arc, where the curvature varies with the allowable maximal velocity until it reaches the limit on the curvature.) The calculations are based on the continuous-curvature kinematic model given in [6].

A method that allows changing in real-time the time distribution along the path while leaving unchanged its geometry is also presented. Such a time-scaling method [8] is needed if one wants to execute the path in an open-loop fashion such that the velocity of the car is no longer an input but generated by an external source [11]. Notice that time-scaling can also be used to transform a system to gain useful properties (e.g. feedback linearizability [18]).

The methods presented in the paper for continuous-curvature path planning are all illustrated by their practical application for an automated parking system developed for a Ford Focus type vehicle.

The remaining part of the paper is organized as follows. The next section presents the kinematic model of the mobile robot with which we work throughout the paper. Section 3 describes the path planning method and the time-scaling is detailed in Section 4. Section 5 presents an example for parallel parking of a passenger car. A short summary concludes the work.

2 Kinematic model of the robot

The model used for the continuous-curvature path planning is a kinematic model introduced in [3] which differs slightly from the general kinematic model of car-like robots described for example in [17]. The general model has three state variables while we use an additional one, hence in our case the configuration of the robot is described by four variables.

The reference point of the car-like robot is the center point of the rear axle denoted by R (see Fig. 1). The configuration q of the car is described by the coordinates (x, y) of the reference point R , the orientation of the car, denoted by ψ , and the curvature κ , which is the converse of the turning radius. The curvature has a sign, which shows the turning direction (left/right). Using these notations the kinematic equation of the car-like mobile robot reads:

$$\dot{q} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \\ \dot{\kappa} \end{bmatrix} = \begin{bmatrix} \cos \psi \\ \sin \psi \\ \kappa \\ 0 \end{bmatrix} v + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \sigma \quad (1)$$

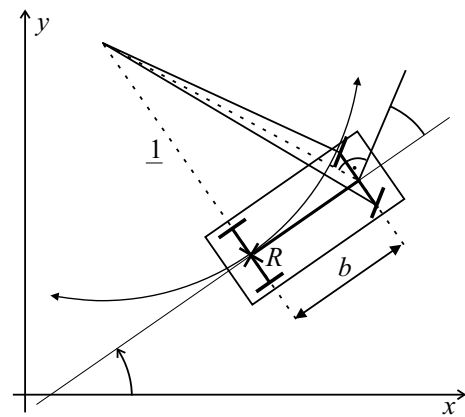


Fig. 1. The parameters of the wheeled mobile robot

The input variables of the robot are the longitudinal velocity of the reference point v and the time derivative of the curvature, denoted by σ , which is related to the steering velocity of the front wheels ($\dot{\phi}$). If b denotes the wheelbase of the vehicle, the following relationships hold (see Fig. 1):

$$\kappa = \frac{\tan \phi}{b} \quad \text{and} \quad \sigma = \dot{\kappa} = \frac{\dot{\phi}}{b \cos^2 \phi} \quad (2)$$

If a path or a path segment is defined by a set of configurations in some world coordinate system, one can shift (with (x_0, y_0) in the x - y plane) or rotate (with ψ_0 around the z axis) it by using a generalized coordinate-transformation:

$$\begin{bmatrix} x' \\ y' \\ \psi' \\ \kappa' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \psi_0 & -\sin \psi_0 & 0 & 0 & x_0 \\ \sin \psi_0 & \cos \psi_0 & 0 & 0 & y_0 \\ 0 & 0 & 1 & 0 & \psi_0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ \psi \\ \kappa \\ 1 \end{bmatrix} \quad (3)$$

3 Path planning

The goal is to plan a feasible path which satisfies some constraints. To solve this problem we use three path primitives based on the kinematic equation of the robot given in (1).

3.1 Constraints

In a motion planning task the feasibility of a trajectory is generally described by geometrical path constraints and kinematic or dynamic constraints [23]. The path constraints define limits on the path geometry (e.g. minimal turning radius, maximal curvature). Kinematic constraints are given for the velocity and acceleration terms (e.g. nonholonomic constraint [7]). The physical limits on torques and forces generated by the motors and the dynamic equations give the dynamic constraints. Since we use the kinematic model of the robot due to the slow longitudinal velocity we only consider the path and kinematic constraints and we disregard the dynamic constraints.

There are some constraints for the variables described in (2) which have to be considered during the motion planning. The value of the maximal curvature is limited since a minimal turning radius should be respected. Moreover, the time derivative of the curvature has also an upper bound since the steering velocity of the front wheels is also limited.

$$|\kappa| \leq \kappa_{\max} \quad \text{and} \quad |\dot{\kappa}| \leq \sigma_{\max} \quad (4)$$

In the path planning algorithm we utilize these limits to get the fastest solution.

The nonholonomic constraint of wheeled mobile robots arises from the kinematic model (1) by eliminating the inputs:

$$\dot{y} \cos \psi - \dot{x} \sin \psi = 0 \quad (5)$$

3.2 Primitives

To plan a feasible path with continuous-curvature we use three different types of primitives: straight line segments, circular arcs, and continuous-curvature turns (CC turns). The main difference between these primitives is their curvature. Moving along a straight line segment gives zero curvature. Turning with constant and finite turning radius (i.e. with non-zero constant curvature) results a circular motion. If the curvature changes linearly with the arc length during the movement a CC turn is used.

In this section these primitives are presented and all parameters are given. To avoid involved calculations first we suppose that the car has a constant velocity during the motion. This constant velocity is denoted by v . (Section 4 discusses how this constant velocity profile can be modified without changing the geometry of the path.)

3.2.1 Primitives with constant curvature

In a straight line or along a circular segment the curvature does not change. Given is a start configuration $q_0 = [x_0, y_0, \psi_0, \kappa_0]^T$, the end of the straight or circular segment

$q_1 = [x_1, y_1, \psi_1, \kappa_1]^T$ and the geometry of the whole primitive can be easily calculated if the parameter of the segment is known (see Table 1). Table 1 presents the properties of these primitives.

3.2.2 Primitives with linearly varying curvature

The most involved primitives are the CC turns which are made from clothoid arcs with upper bounded curvature and curvature derivative. First, the properties of the clothoids are presented than some simpler special CC turns are detailed. Finally, the calculation of general CC turns is given. We also discuss a solution for real-time applications.

A clothoid is a curve whose curvature varies linearly with its arc length:

$$\kappa(t) = \alpha s(t) + \kappa(0), \quad (6)$$

where α is the sharpness of the clothoid and $s(t)$ denotes the length of the arc at time t [6].

A basic CC turn is a clothoid, whose curvature varies from $\kappa(0) = 0$ to $\kappa(T_{CC}) = \kappa_{\max}$, where T_{CC} denotes the time required to finish the basic CC turn. The curvature in the basic CC turn is the following if the velocity v of the car is constant and the maximal allowable input (σ_{\max}) is used:

$$\kappa(t) = \alpha vt + \kappa(0), \quad (7)$$

where the sharpness is

$$\alpha = \frac{\sigma_{\max}}{v}. \quad (8)$$

This choice ensures that

$$\kappa(t) = \sigma_{\max} t + \kappa(0) \quad (9)$$

which satisfies the kinematic model (1) since $\dot{\kappa}(t) = \sigma_{\max}$. The time required to finish the basic CC turn can be calculated from (9) and from the fact that $\kappa(T_{CC}) = \kappa_{\max}$ for $\kappa(0) = 0$

$$T_{CC} = \frac{\kappa_{\max}}{\sigma_{\max}} \quad (10)$$

We use two different types of CC turns depending on the sign of $\frac{d|\kappa|}{dt}$: CC-in ($\frac{d|\kappa|}{dt} > 0$) and CC-out ($\frac{d|\kappa|}{dt} < 0$).

There are two different directions of rotation depending on the sign of κ : positive and negative. During a positive turn ($\kappa \geq 0$) the car rotates counterclockwise while during a negative turn ($\kappa \leq 0$) it rotates clockwise.

Using these properties four different types of CC turns can be distinguished: positive CC-in, negative CC-in, positive CC-out, and negative CC-out. (To describe the four different cases of CC turns we use the following notations: superscript $+$ means a positive turn while superscript $-$ denotes a negative one. CC-in is denoted by a subscript *in* while *out* shows a CC-out.)

If the change of the curvature is maximal (i.e. $\Delta\kappa = \pm\kappa_{\max}$) we obtain a full CC turn. This means that at the initial configuration the curvature is 0 respectively $\pm\kappa_{\max}$, while at the end configuration the curvature is $\pm\kappa_{\max}$ respectively 0.

Tab. 1. Parameters and properties of the primitives with constant curvature

	Straight line	Circular segment
Initial position (x_0, y_0)		arbitrary
Initial orientation ψ_0		arbitrary
Initial curvature κ_0	$\kappa_0 = 0$	$\kappa_0 \neq 0$
Velocity v		arbitrary constant
Steering input σ		$\sigma = 0$
Parameter of the segment	l - length	φ - turning angle
End position (x_1, y_1)	function of x_0, y_0, ψ_0, l	function of $x_0, y_0, \psi_0, \kappa_0, \varphi$
End orientation ψ_1	$\psi_1 = \psi_0$	$\psi_1 = \psi_0 + \varphi$
End curvature κ_1	$\kappa_1 = 0$	$\kappa_1 = \kappa_0$
Duration of the segment	$T_l = \frac{l}{v}$	$T_\varphi = \frac{\varphi}{v \kappa_0 }$

First, the equations of the full positive CC-in turn will be given. The exact calculations cannot be executed in real-time, since one cannot give the configurations in closed form, hence we suggest to make some precalculations for the full CC_{in}^+ . After that we will use elementary and computationally effective mathematical operations to get the equations of all other (full and general) CC turns from the beforehand calculated CC_{in}^+ in closed form.

In the sequel we will suppose that the motion starts from the $q_0 = [0, 0, 0, 0]^T$ initial configuration in case of full CC_{in} turns or from $q_0 = [0, 0, 0, \pm\kappa_{max}]^T$ in case of full CC_{out} turns. To get the CC turns starting from arbitrary initial position and orientation one should use (3).

Full positive CC-in. ($\kappa \geq 0, \frac{d|\kappa|}{dt} = \dot{\kappa} > 0$) The robot follows a clothoid arc of sharpness $\alpha = \sigma_{max}/v$ with $\kappa(0) = 0$ and $\kappa(T_{CC}) = \kappa_{max}$. This case gives the equations for the basic CC turn and it is also described in [6].

If the robot moves along such a curve the time functions of the configuration changes read:

$$x_{in}^+(t) = \sqrt{\frac{\pi}{\alpha}} C_F \left(\sqrt{\frac{\kappa_{in}^+(t)^2}{\pi \alpha}} \right) \quad (11)$$

$$y_{in}^+(t) = \sqrt{\frac{\pi}{\alpha}} S_F \left(\sqrt{\frac{\kappa_{in}^+(t)^2}{\pi \alpha}} \right) \quad (12)$$

$$\psi_{in}^+(t) = \frac{\kappa_{in}^+(t)^2}{2\alpha} \quad (13)$$

$$\kappa_{in}^+(t) = \sigma_{max} t \quad (14)$$

where C_F and S_F denote the Fresnel integrals:

$$C_F(x) = \int_0^x \cos\left(\frac{\pi}{2}t^2\right) dt, \quad S_F(x) = \int_0^x \sin\left(\frac{\pi}{2}t^2\right) dt \quad (15)$$

The end configuration of a full positive CC-in turn starting from the $q_0 = [0, 0, 0, 0]^T$ configuration is:

$$q_1 = \begin{bmatrix} x_1 \\ y_1 \\ \psi_1 \\ \kappa_1 \end{bmatrix} = \begin{bmatrix} x_{in}^+(T_{CC}) \\ y_{in}^+(T_{CC}) \\ \psi_{in}^+(T_{CC}) \\ \kappa_{in}^+(T_{CC}) = \kappa_{max} \end{bmatrix} \quad (16)$$

where T_{CC} is defined by (10).

Since Fresnel integrals are required for the calculations, it is not possible to carry out them in real-time. Hence we suggest the calculation of the values of the integrals beforehand for discrete time instants (i.e. for $t = nT_S$, where $n \in \mathbb{N}$ and $0 \leq n \leq T_{CC}/T_S$). If the constant velocity of the car is the same for all path planning tasks, the same attributes are required for (15): The parameters of the constraints ($\kappa_{max}, \sigma_{max}$) are constant and the curvature varies in the same way ($\kappa(t) = \sigma_{max}t$) for all CC_{in}^+ turns. Thus the same Fresnel integrals are required for the calculations of full CC_{in}^+ turns.

Full negative CC-in. ($\kappa \leq 0, \frac{d|\kappa|}{dt} = -\dot{\kappa} > 0$) We consider the motion from $q_0 = [0, 0, 0, 0]^T$. The sharpness of the clothoid is $-\sigma_{max}/v$. During this motion the curvature and its time derivative vary as:

$$\kappa_{in}^-(t) = -\sigma_{max}t = -\kappa_{in}^+(t) \leq 0 \quad (17)$$

$$\dot{\kappa}_{in}^-(t) = -\sigma_{max} = -\dot{\kappa}_{in}^+(t) \quad (18)$$

The evolution of the x and y coordinates are the following in a negative CC-in:

$$x_{in}^-(t) = \sqrt{\pi/\alpha} C_F \left(\sqrt{\kappa_{in}^-(t)^2/(\pi\alpha)} \right) = \sqrt{\pi/\alpha} C_F \left(\sqrt{(-\kappa_{in}^+(t))^2/(\pi\alpha)} \right) = x_{in}^+(t) \quad (19)$$

$$y_{in}^-(t) = -\sqrt{\pi/\alpha} S_F \left(\sqrt{\kappa_{in}^-(t)^2/(\pi\alpha)} \right) = -\sqrt{\pi/\alpha} S_F \left(\sqrt{(-\kappa_{in}^+(t))^2/(\pi\alpha)} \right) = -y_{in}^+(t) \quad (20)$$

The equation of the orientation is:

$$\psi_{in}^-(t) = -\frac{\kappa_{in}^-(t)^2}{2\alpha} = -\frac{(-\kappa_{in}^+(t))^2}{2\alpha} = -\psi_{in}^+(t) \quad (21)$$

Notice, that using the equations (19)-(21) we really get $x_{in}^-(0) = 0, y_{in}^-(0) = 0$ and $\psi_{in}^-(0) = 0$ as initial configuration.

Full positive CC-out. ($\kappa \geq 0$, $\frac{d|\kappa|}{dt} = \dot{\kappa} < 0$) The full positive CC-out turn starts from $q_0 = [0, 0, 0, \kappa_{\max}]^T$, and the sharpness of the clothoid is $-\sigma_{\max}/v$. The equations for the positive CC-out differ also from (11)-(14). The time function and time derivative of the curvature are the following:

$$\kappa_{out}^+(t) = \kappa_{\max} - \sigma_{\max}t = \quad (22)$$

$$\sigma_{\max}(T_{CC} - t) = \kappa_{in}^+(T_{CC} - t) \geq 0$$

$$\dot{\kappa}_{out}^+(t) = -\sigma_{\max} = -\dot{\kappa}_{in}^+(t) \quad (23)$$

The equations for the x and y coordinates read:

$$\begin{aligned} x_{out}^+(t) &= -\sqrt{\pi/\alpha}C_F \left(\sqrt{\kappa_{out}^+(t)^2/(\pi\alpha)} \right) = \\ &= -\sqrt{\pi/\alpha}C_F \left(\sqrt{\kappa_{in}^+(T_{CC} - t)^2/(\pi\alpha)} \right) = -x_{in}^+(T_{CC} - t) \end{aligned} \quad (24)$$

$$\begin{aligned} y_{out}^+(t) &= \sqrt{\pi/\alpha}S_F \left(\sqrt{\kappa_{out}^+(t)^2/(\pi\alpha)} \right) = \\ &= \sqrt{\pi/\alpha}S_F \left(\sqrt{\kappa_{in}^+(T_{CC} - t)^2/(\pi\alpha)} \right) = y_{in}^+(T_{CC} - t) \end{aligned} \quad (25)$$

The equation of the orientation is:

$$\psi_{out}^+(t) = -\frac{\kappa_{out}^+(t)^2}{2\alpha} = -\frac{\kappa_{in}^+(T_{CC} - t)^2}{2\alpha} = -\psi_{in}^+(T_{CC} - t) \quad (26)$$

Constant values could be also added to the Eqs. (24)-(25) without modifying their time derivatives. We select constants which guarantee $x_{out}^+(0) = 0$ and $y_{out}^+(0) = 0$ for a 0 start position. So we get the following equations for the positive CC-out:

$$x_{out}^+(t) = -x_{in}^+(T_{CC} - t) + x_{in}^+(T_{CC}) \quad (27)$$

$$y_{out}^+(t) = y_{in}^+(T_{CC} - t) - y_{in}^+(T_{CC}) \quad (28)$$

Similarly, to get $\psi_{out}^+(0) = 0$ we also modify (26):

$$\psi_{out}^+(t) = -\psi_{in}^+(T_{CC} - t) + \psi_{in}^+(T_{CC}) \quad (29)$$

Full negative CC-out. ($\kappa \leq 0$, $\frac{d|\kappa|}{dt} = -\dot{\kappa} < 0$) The start configuration is $q_0 = [0, 0, 0, -\kappa_{\max}]^T$, the sharpness of the clothoid is σ_{\max}/v in this case. The curvature and its time derivative vary in the negative CC-out according to the following equations.

$$\kappa_{out}^-(t) = -\kappa_{\max} + \sigma_{\max}t = \quad (30)$$

$$-\sigma_{\max}(T_{CC} - t) = -\kappa_{in}^-(T_{CC} - t) \leq 0 \dot{\kappa}_{out}^-(t) =$$

$$\sigma_{\max} = \dot{\kappa}_{in}^-(t) \quad (31)$$

We give again the evolution of the configuration variables.

$$\begin{aligned} x_{out}^-(t) &= -\sqrt{\pi/\alpha}C_F \left(\sqrt{\kappa_{out}^-(t)^2/(\pi\alpha)} \right) = \\ &= -\sqrt{\pi/\alpha}C_F \left(\sqrt{(-\kappa_{in}^-(T_{CC} - t))^2/(\pi\alpha)} \right) = \\ &= -x_{in}^-(T_{CC} - t) \end{aligned} \quad (32)$$

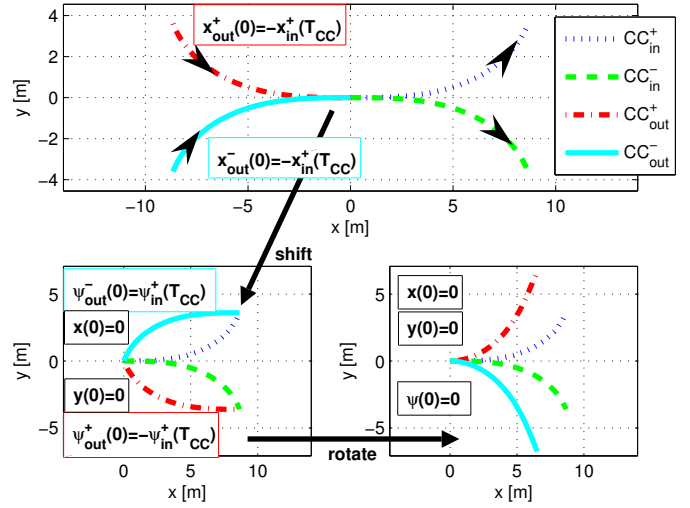


Fig. 2. Four different full CC turns in the x - y plane

$$\begin{aligned} y_{out}^-(t) &= -\sqrt{\pi/\alpha}S_F \left(\sqrt{\kappa_{out}^-(t)^2/(\pi\alpha)} \right) = \\ &= -\sqrt{\pi/\alpha}S_F \left(\sqrt{(-\kappa_{in}^-(T_{CC} - t))^2/(\pi\alpha)} \right) = \\ &= -y_{in}^-(T_{CC} - t) \end{aligned} \quad (33)$$

$$\psi_{out}^-(t) = \frac{\kappa_{out}^-(t)^2}{2\alpha} = \frac{(-\kappa_{in}^-(T_{CC} - t))^2}{2\alpha} = \psi_{in}^-(T_{CC} - t) \quad (34)$$

We add again constants to equations (32)-(34) to satisfy $x_{out}^-(0) = 0$, $y_{out}^-(0) = 0$ and $\psi_{out}^-(0) = 0$

$$x_{out}^-(t) = -x_{in}^-(T_{CC} - t) + x_{in}^-(T_{CC}) \quad (35)$$

$$y_{out}^-(t) = y_{in}^-(T_{CC} - t) - y_{in}^-(T_{CC}) \quad (36)$$

$$\psi_{out}^-(t) = \psi_{in}^-(T_{CC} - t) - \psi_{in}^-(T_{CC}) \quad (37)$$

It can be seen from the equations that the configurations of all full CC turns can be determined in real-time provided that the evolution of the configuration variables are calculated for the full CC_{in}^+ turn. Moreover, the four different CC turns are congruent. Using mirroring, shifting and rotation we can get all the full CC turns from one (see Fig. 2).

General CC turn. In all full CC turns the curvature changes from 0 to the maximal curvature $\pm\kappa_{\max}$ or conversely. In the general case the curvature varies continuously from an arbitrary $-\kappa_{\max} \leq \kappa_0 \leq \kappa_{\max}$ value to an arbitrary $-\kappa_{\max} \leq \kappa_1 \leq \kappa_{\max}$ value. We only suppose that $\kappa_0 \cdot \kappa_1 \geq 0$, which means that κ_0 and κ_1 do not have different signs. (If $\kappa_0 \cdot \kappa_1 < 0$ then the path can be divided into two CC turns: a CC turn with curvature between κ_0 and 0 and a second CC turn with curvature varying from 0 to κ_1 .)

So the general CC turn moves the robot from the arbitrary $q_0 = [x_0, y_0, \psi_0, \kappa_0]^T$ initial configuration to a $q_1 = [x_1, y_1, \psi_1, \kappa_1]^T$ end configuration, where $-\kappa_{\max} \leq \kappa_1 \leq \kappa_{\max}$ is an arbitrary final curvature, with $\kappa_0 \cdot \kappa_1 \geq 0$.

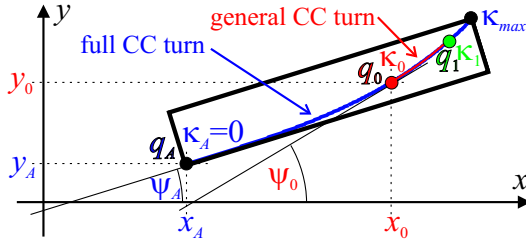


Fig. 3. Calculation of the general CC turn

To calculate the whole geometry or the q_1 end configuration of the general CC turn one needs the following parameters:

- x_0, y_0, ψ_0 – start position and orientation,
- κ_0 – curvature at the initial position,
- v – velocity of the car, which is supposed to be constant,
- a parameter describing the length or the duration of the segment (e.g. $\Delta\kappa = \kappa_1 - \kappa_0$ – change in the curvature, $T_{\Delta\kappa} = \Delta\kappa/\sigma_{\max}$ – duration of the segment).

In order to calculate the general case in real-time we derive it from the full CC turn. The general turn is a section of the full CC turn defined by κ_0 and κ_1 as depicted in Fig. 3.

Without loss of generality we only consider now the positive CC-in turn since all the other cases give similar results. This implies that $0 \leq \kappa_0 \leq \kappa_1 \leq \kappa_{\max}$ and the sharpness of the clothoid is $\alpha = \sigma_{\max}/v$.

First we determine the starting point of the corresponding full CC turn. This configuration is denoted by $q_A = [x_A, y_A, \psi_A, 0]^T$. Notice that q_0 is the configuration of this full CC turn where the curvature is κ_0 and this configuration can be reached in $t_A = \kappa_0/\sigma_{\max}$ time from q_A . Using (3), the configuration at the point A reads:

$$\psi_A = \psi_0 - \frac{\kappa_0^2}{2\alpha} \quad (38)$$

$$x_A = x_0 - \sqrt{\frac{\pi}{\alpha}} C_F \left(\sqrt{\frac{\kappa_0^2}{\pi\alpha}} \right) \cos \psi_A + \quad (39)$$

$$\sqrt{\frac{\pi}{\alpha}} S_F \left(\sqrt{\frac{\kappa_0^2}{\pi\alpha}} \right) \sin \psi_A$$

$$y_A = y_0 - \sqrt{\frac{\pi}{\alpha}} C_F \left(\sqrt{\frac{\kappa_0^2}{\pi\alpha}} \right) \sin \psi_A - \quad (40)$$

$$\sqrt{\frac{\pi}{\alpha}} S_F \left(\sqrt{\frac{\kappa_0^2}{\pi\alpha}} \right) \cos \psi_A$$

The points of the general CC turn are the points of a section of the full CC turn from the configuration q_0 . If one needs t time to reach a configuration from q_0 in the general CC turn, it takes $t_A + t = \kappa_0/\sigma_{\max} + t$ time to reach the same configuration from

q_A in the full CC turn. (Notice, that $t \leq T_{\Delta\kappa}$ and $t_A + t \leq T_{CC}$.)

$$x_g(t) = x_A + \sqrt{\frac{\pi}{\alpha}} C_F \left(\sqrt{\frac{\kappa_g(t)^2}{\pi\alpha}} \right) \cos \psi_A - \quad (41)$$

$$\sqrt{\frac{\pi}{\alpha}} S_F \left(\sqrt{\frac{\kappa_g(t)^2}{\pi\alpha}} \right) \sin \psi_A = x_A + x_{in}^+(\kappa_0/\sigma_{\max} + t)$$

$$\cos \psi_A - y_{in}^+(\kappa_0/\sigma_{\max} + t) \sin \psi_A$$

$$y_g(t) = y_A + \sqrt{\frac{\pi}{\alpha}} C_F \left(\sqrt{\frac{\kappa_g(t)^2}{\pi\alpha}} \right) \sin \psi_A + \quad (42)$$

$$\sqrt{\frac{\pi}{\alpha}} S_F \left(\sqrt{\frac{\kappa_g(t)^2}{\pi\alpha}} \right) \cos \psi_A = y_A +$$

$$x_{in}^+(\kappa_0/\sigma_{\max} + t) \sin \psi_A + y_{in}^+(\kappa_0/\sigma_{\max} + t) \cos \psi_A$$

$$\psi_g(t) = \psi_A + \frac{\kappa_g(t)^2}{2\alpha} = \psi_A + \psi_{in}^+(\kappa_0/\sigma_{\max} + t) \quad (43)$$

$$\kappa_g(t) = \sigma_{\max} \left(\frac{\kappa_0}{\sigma_{\max}} + t \right) = \kappa_0 + \sigma_{\max} t = \quad (44)$$

$$\kappa_{in}^+(\kappa_0/\sigma_{\max} + t) \leq \kappa_1$$

Observe that the calculations of the general CC turn use same Fresnel integrals as in the full positive CC-in turn since $0 \leq \kappa_g(t) \leq \kappa_{\max}$, and $\kappa_{\max}, \sigma_{\max}, v$ have the same values. Hence if the points of the full positive CC-in turn are calculated beforehand, than the general positive in turns can also be determined in real-time. Notice also that the general CC turns sections can be calculated for other (negative and/or out) CC turns similarly according to the relationships between the different CC turns.

3.3 Planning the whole path

There are several methods to patch together a trajectory using the three primitives enumerated in Section 3.2. Probabilistic or deterministic path planning algorithms can also be used to solve the task. For example the Probabilistic RoadMap method in [21] or the Rapidly Exploring Random Tree algorithm in [14] can be applied with these primitives. To carry out special tasks, for example in the automatic parking case, deterministic methods can be used. The following patch rules have to be satisfied to get a continuous-curvature path with upper bounded curvature and curvature derivative:

R1 If the curvature κ equals zero only straight line segment or CC turn starting from $\kappa_0 = 0$ can be used (both full or general turns).

R2 If the curvature is a non-zero, say κ_A , the motion can be continued along a circular segment with turning radius $|1/\kappa_A|$ or using a general CC turn starting from $\kappa_0 = \kappa_A$. The direction of the rotation is determined by the sign of κ_A .

R3 The length of the straight line and circular segments is arbitrary while the length of a CC turn is restricted by the upper-

bound of the curvature (see (4)) and by the $\kappa_0 \cdot \kappa_1 \geq 0$ condition (see Section 3.2.2).

We suggest not to utilize the entire range for κ and σ given in (4) for the path planning. These bounds should be reduced during the motion planning to leave margin for the tracking controller: $\kappa'_{\max} = S_\kappa \kappa_{\max}$ and $\sigma'_{\max} = S_\sigma \sigma_{\max}$, where $0 < S_\kappa \leq 1$ and $0 < S_\sigma \leq 1$ are design parameters.

3.4 Accuracy

Since real-time calculation is required, the Fresnel integrals in (15) are calculated beforehand at discrete time instants (i.e. for $t = nT_S$ where $n \in \mathbb{N}$ and $0 \leq n \leq T_{CC}/T_S$). The error of path planning depends on T_S and on the velocity v .

The upper bound on the path error in a CC turn can be calculated from (11)-(14). This error is the maximum change in the configuration between time moments $T_1 \leq T_{CC}$ and $T_1 + T_S$:

$$\Delta \kappa_{\max} = \max(\kappa(T_1 + T_S) - \kappa(T_1)) = \sigma_{\max} T_S \quad (45)$$

$$\Delta x_{\max} = \max(x(T_1 + T_S) - x(T_1)) \leq v T_S \quad (46)$$

$$\Delta y_{\max} = \max(y(T_1 + T_S) - y(T_1)) \leq v T_S \quad (47)$$

$$\Delta \psi_{\max} = \max(\psi(T_1 + T_S) - \psi(T_1)) \leq \frac{(\Delta \kappa_{\max})^2}{2\alpha} + \kappa_{\max} v T_S \quad (48)$$

These results meet our expectations: for larger T_S , v , σ_{\max} and κ_{\max} values one obtains larger path errors. Using the data of the Ford Focus, for which we design the automated parking system we get the following results:

$$\begin{aligned} \kappa_{\max} &= 0.24 \text{m}^{-1}, & \sigma_{\max} &= 0.16(\text{ms})^{-1}, \\ v &= 3 \text{ms}^{-1}, & T_S &= 0.01 \text{s} \end{aligned}$$

$$\begin{aligned} \Delta \kappa_{\max} &= 0.0016 \text{m}^{-1}, & \Delta x_{\max} &= \Delta y_{\max} = 0.03 \text{m}, \\ \Delta \psi_{\max} &= 0.0072 \text{rad} \end{aligned}$$

For a car-like robot and considering the accuracy of other modules of the parking system (position estimation, distance measurements) these values are small enough.

4 Time-scaling of the reference

We supposed that the velocity v of the car is constant during the movement. If this velocity varies the geometry of the path can be modified. Tracking a straight line segment or a circular path with different velocities does not change the geometry of the path. If the velocity of the car differs in the different CC turns the geometry of the path varies. To avoid this, we suggest to use time-scaling of the reference trajectory.

The reason of the geometrical changes in CC turns is that the time T_{CC} , which is required to make a full CC turn, is constant, it does not depend on the velocity v of the car, since according to (10) T_{CC} is only determined by the car parameters (κ_{\max} and σ_{\max}). The length of a CC turn (s_{CC}) is then determined by this constant T_{CC} and the v velocity of the car which is also supposed to be constant:

$$s_{CC} = v T_{CC} = \frac{\kappa_{\max}}{\sigma_{\max}} v \quad (49)$$

For different velocities the length of the CC turn changes, thus the geometry of the whole path also varies.

To overcome this problem time-scaling [4, 8] can be used to modify the velocity profile of a previously designed path without changing its geometry. In our work we use this method to change the constant velocity of the planned trajectory to an arbitrary non-vanishing velocity profile.

Several methods in the literature use the same design procedure: first the geometry is calculated than the velocity profile is determined (e.g. [23]). The goal of the time-scaling may be to get optimal input signals [8], to improve the closed-loop behavior [22] or to control an underactuated robot [11].

Suppose that the planned reference trajectory evolves according to time τ and its velocity profile can be modified using time-scaling such that the resulted reference evolves according to time $t(\tau)$. Both references satisfy the kinematic equation (1) of the robot, i.e. for the reference evolving according to time τ

$$x'_\tau(\tau) = v_\tau(\tau) \cos \psi_\tau(\tau) \quad (50)$$

$$y'_\tau(\tau) = v_\tau(\tau) \sin \psi_\tau(\tau) \quad (51)$$

$$\psi'_\tau(\tau) = v_\tau(\tau) \kappa_\tau(\tau) \quad (52)$$

$$\kappa'_\tau(\tau) = \sigma_\tau(\tau) \quad (53)$$

where $x' = \frac{dx(\tau)}{d\tau}$. From this reference we would like to get a new reference evolving with time t such that

$$\dot{x}_{ref}(t) = v_{ref}(t) \cos \psi_{ref}(t) \quad (54)$$

$$\dot{y}_{ref}(t) = v_{ref}(t) \sin \psi_{ref}(t) \quad (55)$$

$$\dot{\psi}_{ref}(t) = v_{ref}(t) \kappa_{ref}(t) \quad (56)$$

$$\dot{\kappa}_{ref}(t) = \sigma_{ref}(t) \quad (57)$$

where $v_{ref}(t)$ describes the arbitrary non-vanishing velocity profile of the new reference.

To ensure this the time-scaling is used as

$$x_{ref}(t) = x_\tau(\tau) \quad (58)$$

$$y_{ref}(t) = y_\tau(\tau) \quad (59)$$

$$\psi_{ref}(t) = \psi_\tau(\tau) \quad (60)$$

$$\kappa_{ref}(t) = \kappa_\tau(\tau) \quad (61)$$

and the derivative of the time t according to τ should be chosen such that

$$\frac{dt}{d\tau} = \frac{1}{\dot{t}} = \frac{v_\tau(\tau)}{v_{ref}(t)} \quad (62)$$

where $t(0) = 0$, since the motion should start from the same initial position. Notice, that $v_{ref}(t) \neq 0$ and $v_\tau(\tau) \neq 0$ are both supposed.

There is one more condition for $v_{ref}(t)$. To follow a scaled reference trajectory modified input signals are required. The restriction is that the modified input should also satisfy the input bound limit in (4):

$$\sigma_{ref}(t) = \dot{\kappa}_{ref}(t) = \frac{d\kappa_\tau(\tau)}{dt} = \kappa'_\tau(\tau) \dot{t} = \sigma_\tau(\tau) \frac{v_{ref}(t)}{v_\tau(\tau)} \leq \sigma_{\max} \quad (63)$$

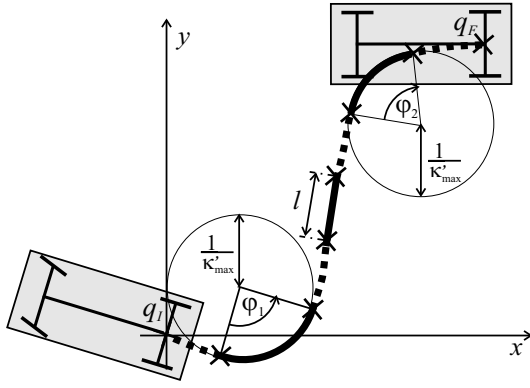


Fig. 4. Path for parallel parking with $q_I = [x_I = 0, y_I = 0, \psi_I \neq 0, \kappa_I \neq 0]^T$ and $q_F = [x_F \neq 0, y_F \neq 0, \psi_F = 0, \kappa_F = 0]^T$

We mention that the time-scaling can be performed on-line (see [4]), hence the whole path with arbitrary non-vanishing velocity profile can be calculated in real-time.

5 Path planning for parking maneuvers

The goal is to plan a continuous-curvature path for a Ford Focus type vehicle allowing the realization of a parallel parking maneuver (the solution for the other types of parking maneuvers being similar). The initial configuration of the car is $q_I = [x_I, y_I, \psi_I, \kappa_I]^T$ and it has to reach the goal position $q_F = [x_F, y_F, \psi_F, 0]^T$ (see Figure 4). If the curvature in the initial configuration (κ_I) is 0, the path starts with a full positive CC-in turn, otherwise a general CC turn gives the first segment of the trajectory. Then a circular motion is required with turning angle ϕ_1 . This is then followed by a positive CC-out turn. The middle segment of the path is a straight line with length l . A negative turn finishes the motion, this part contains a negative CC-in, a circular segment with turning angle ϕ_2 and finally a negative CC-out turn.

The path has three parameters, namely the turning angles in the circular segments (ϕ_1, ϕ_2) and the length of the straight line (l). We have three equations for the end configuration: the desired end position (x_F, y_F) and the desired final orientation (ψ_F) determine q_F . (The finishing part of the trajectory is a full CC-out turn, which ensures that the curvature at the final position is 0.) The three equations are given by the geometry of the path planning primitives using the three path parameters:

$$x_F = f(\kappa'_{\max}, \sigma'_{\max}, v, x_I, \psi_I, \kappa_I, \phi_1, \phi_2, l) \quad (64)$$

$$y_F = g(\kappa'_{\max}, \sigma'_{\max}, v, y_I, \psi_I, \kappa_I, \phi_1, \phi_2, l) \quad (65)$$

$$\psi_F = h(\kappa'_{\max}, \sigma'_{\max}, v, \psi_I, \kappa_I, \phi_1, \phi_2) \quad (66)$$

From the three equations the parameters ϕ_1, ϕ_2 and l can be calculated, since they are the only unknown variables in the equations. If the equations for the full positive CC-in turn (see (11)-(14)) or the Fresnel integrals which are required to solve (11)-(12) are calculated beforehand, the parameters ϕ_1, ϕ_2 and l and the geometry of the path can be calculated in real-time.

The parking is not possible if the robot collides with an obstacle while it follows the reference path. To check this one has to

take into account the dimensions of the car, the specified safety distances and the points of the planned reference path.

For the simulations Matlab with Simulink was used. The real-time calculations were done by the AutoBox of dSpace. In the example the car has to start the motion from $q_I = [0, 0, -\pi/10, 0.1]^T$ and the desired goal configuration is $q_F = [15, 9, 0, 0]^T$ where distances are in meters, angles are in radians and curvatures are in m^{-1} . It has to move backwards with $v = 2ms^{-1}$ velocity.

Fig. 5a shows the designed path for the midpoint of the rear axle. The path parameters are the following in this case: $\phi_1 = 0.7136rad$, $\phi_2 = 0.3347rad$, and $l = 4.227m$. The time functions and the first time derivatives of the configuration along this path are depicted in Figs. 6a-7b. The input signals which are required to follow the path depicted in Fig. 5a are shown in Figs. 5b, 7b.

Recall, that once the Fresnel integrals were calculated beforehand, all the calculations were done in real-time.

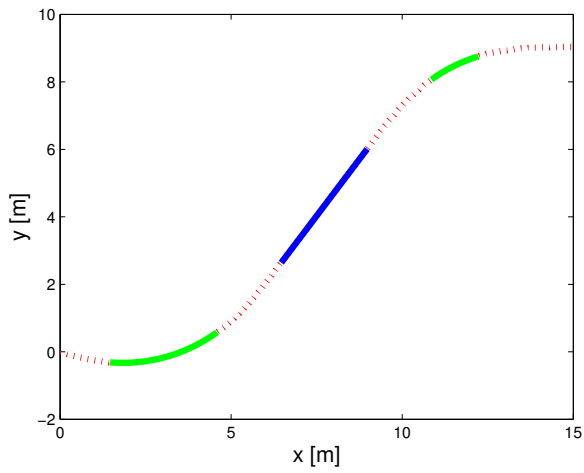
The previous path was designed for a constant velocity (see Fig. 5b). Now we use the time-scaling to modify the velocity profile of the path without changing its geometry as it is also detailed in Section 4. Figs. 8a-9b show the results of time-scaling. The time functions of x_{ref}, y_{ref} and ψ_{ref} are depicted in Figs. 8a-9a for the velocity profiles shown in Fig. 9b.

6 Conclusion

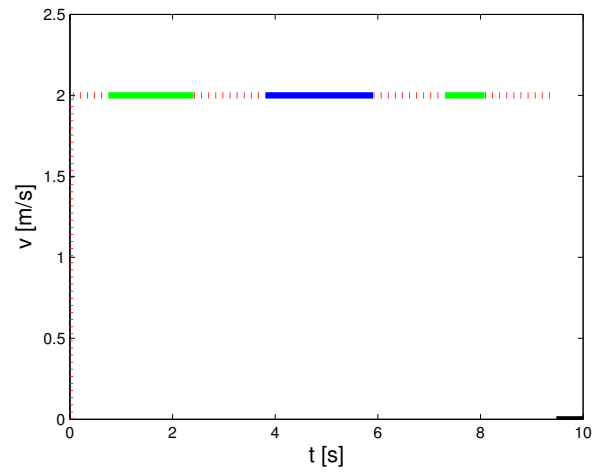
This paper deals with path planning for car-like mobile robots. The primitives which can be used to plan continuous-curvature trajectories are described; the most involved one, the CC turn is detailed. The paper shows that after some preliminary calculations the path can be determined using computationally effective mathematical operations and the time distribution along it can be modified in real-time without changing its geometry. An example is presented for the parallel parking of a passenger car, the results of real-time simulations are also given.

References

- 1 **AISIN SEIKI Co. Ltd.**, *Intelligent Parking Assist*, available at <http://www.aisin.com/product/automotive/info/ot.html>.
- 2 **Barraquand J, Latombe J C**, *On non-holonomic mobile robots and optimal maneuvering*, *Revue d'Intelligence Artificielle* **3** (1989), no. 2, 77–103.
- 3 **Boissonnat J-D., Cérézo A, Leblond J**, *Shortest paths of bounded curvature in the plane*, *J. of Intelligent and Robotics Systems* **11** (1994), 5–20, DOI 10.1007/BF01258291.
- 4 **Dahl O, Nielsen L**, *Torque-limited path following by on-line trajectory time scaling*, *IEEE Trans. on Robotics and Automation* **6** (1990), no. 5, 554–561, DOI 10.1109/ROBOT.1989.100131.
- 5 **Dubins L E**, *On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents*, *American J. of Mathematics* **79** (1957), 497–517, DOI 10.2307/2372560.
- 6 **Fraichard Th, Scheuer A**, *From Reeds and Shepp's to continuous-curvature paths*, *IEEE Trans. on Robotics and Automation* **20** (2004), no. 6, 1025–1035, DOI 10.1109/TRO.2004.833789.
- 7 **Harmati I, Lantos B, Payandeh S**, *Nonholonomic stratified motion planning using decomposed trajectories*, *Proc. of 8th IEEE Int. Conf. on Methods and Models in Automation and Robotics (MMAR)*, 2002, pp. 953–958.

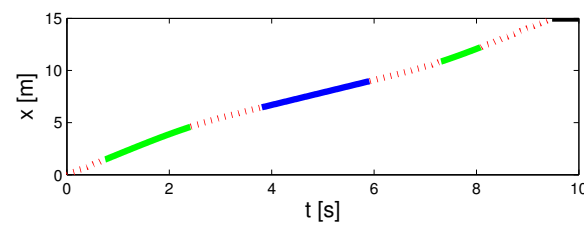


(a) Path in the $x - y$ plane

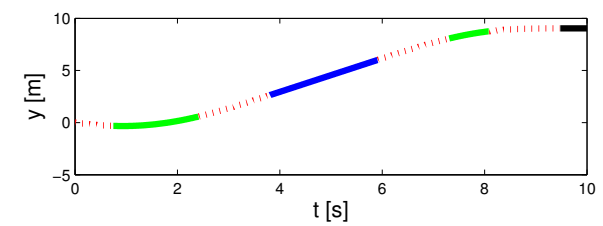


(b) $v_{ref}(t)$

Fig. 5. The reference path and its velocity profile

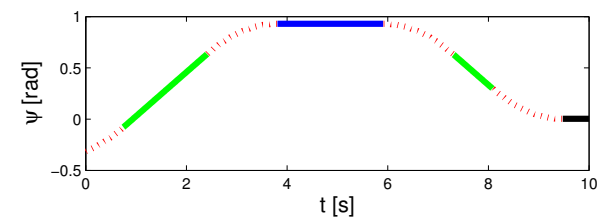


(a) $x_{ref}(t)$ and $\dot{x}_{ref}(t)$

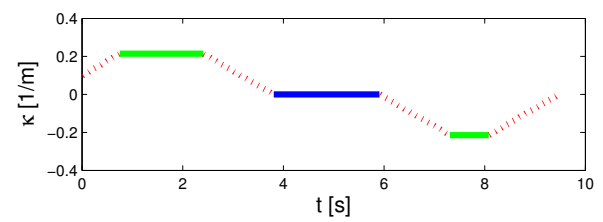


(b) $y_{ref}(t)$ and $\dot{y}_{ref}(t)$

Fig. 6. The reference coordinates



(a) $\psi_{ref}(t)$ and $\dot{\psi}_{ref}(t)$



(b) $\kappa_{ref}(t)$ and $\sigma_{ref}(t) = \dot{\kappa}_{ref}(t)$

Fig. 7. The reference values for orientation and curvature

8 **Hollerbach J M**, *Dynamic scaling of manipulator trajectories*, Trans. of the ASME, J. of Dynamic Systems, Measurement, and Control **106** (1984), no. 1, 102–106, DOI 10.1115/1.3149652.

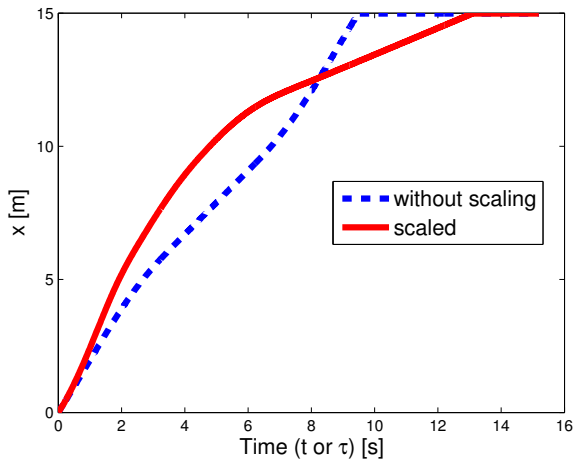
9 **Isidori A**, *Nonlinear control systems*, Springer, Berlin, 1995.

10 **Jacobs P E, Canny J.**, *Planning smooth paths for mobile robots*,

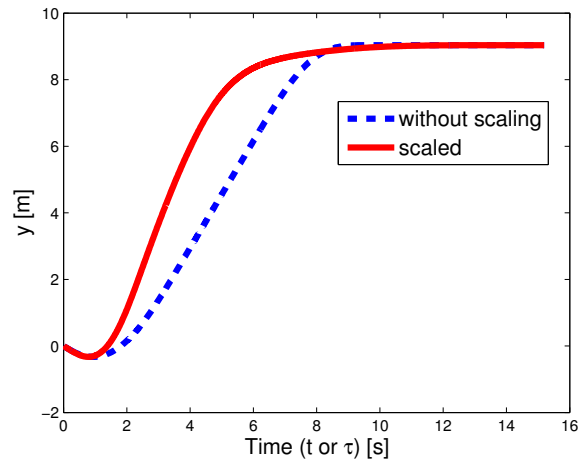
Proc. of IEEE Conf. on Robotics and Automation, 1989, pp. 2–7, DOI 10.1109/ROBOT.1989.99959, (to appear in print).

11 **Kiss B, Szádeczky-Kardoss E.**, *Tracking control of the orbitally flat kinematic car with a new time-scaling input*, Proc. of IEEE Conf. on Decision and Control, 2007, pp. 1969–1974.

12 **Kostov V. P., Degtiarova-Kostova E. V.**, *The planar motion with bounded*

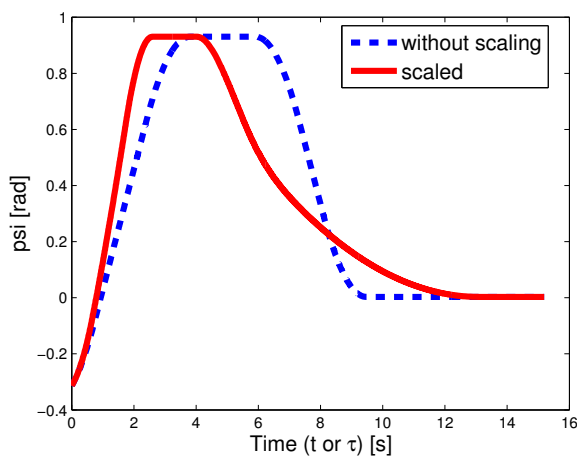


(a) x_{ref} without and with time-scaling

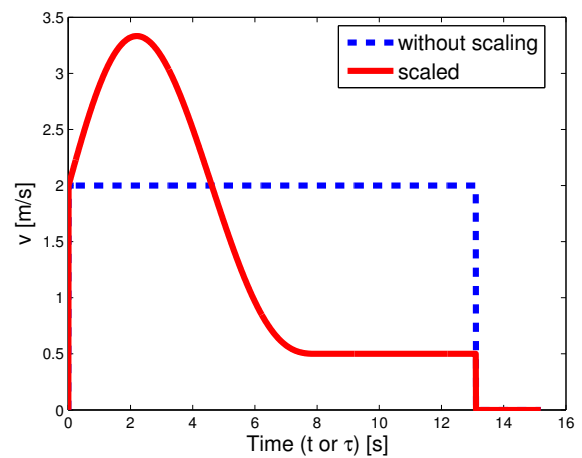


(b) y_{ref} without and with time-scaling

Fig. 8. Reference coordinates without and with time-scaling



(a) ψ_{ref} without and with time-scaling



(b) v_{ref} without and with time-scaling

Fig. 9. Reference orientation and velocity without and with time-scaling

derivative of the curvature and its suboptimal paths, Acta Mathematica Universitatis Comenianae (1995), 185–226.

- 13 Lamiroux F., Laumond J.-P., *Smooth motion planning for car-like vehicles*, IEEE Trans. on Robotics and Automation **17** (2001), no. 4, DOI 10.1109/70.954762.
- 14 LaValle S. M., Kuffner J. J., *Rapidly-exploring random trees: Progress and prospects*, Proc. of 2000 int. workshop on the algorithmic foundations of robotics, 2000.
- 15 Paromtchik I. E., Laugier C., *Autonomous parallel parking of nonholonomic vehicle*, Proc. of IEEE intelligent vehicles symposium, 1996, pp. 3117–3122, DOI 10.1109/IVS.1996.566343, (to appear in print).
- 16 Reeds J. A., Shepp L. A., *Optimal paths for a car that goes both forwards and backwards*, Pacific J. of Mathematics **145** (1990), no. 2.
- 17 Rouchon P., Fliess M., Lévine J., Ph. Martin, *Flatness and motion planning: The car with n-trailers*, ECC'93, proc. of european control conference, 1993, pp. 1518–1522.
- 18 Sampei Mitsuji, Furuta Katsuhisa, *On time scaling for nonlinear systems: Application to linearization*, IEEE Trans. on Automatic Control **AC-31** (1986), 459–462.
- 19 Scheuer A., Fraichard Th., *Planning continuous-curvature paths for car-like robots*, IEEE/RSJ int. conf. on intelligent robots and systems, 1996, pp. 1304–1311, DOI 10.1109/IROS.1996.568985.
- 20 Scheuer A., Laugier Ch., *Planning sub-optimal and continuous-curvature paths for car-like robots*, IEEE/RSJ int. conf. on intelligent robots and systems, 1998, pp. 25–31, DOI 10.1109/IROS.1998.724591.
- 21 Svestka P., Overmars M. H., *Motion planning for carlike robots using a probabilistic learning approach*, Int. J. of Robotics Research **16** (1997), no. 2, 119–143, DOI 10.1177/027836499701600201.
- 22 Szádeczky-Kardoss E., Kiss B., *Tracking error based on-line trajectory time scaling*, Ines 2006, proc. of 10th int. conf. on intelligent engineering systems, 2006, pp. 80–85, DOI 10.1109/INES.2006.1689346, (to appear in print).
- 23 Wu Weiguo, Chen Huitang, Woo Peng-Yung, *Time optimal path planning for a wheeled mobile robot*, J. of Robotic Systems **17** (2000), no. 11, DOI 10.1002/1097-4563(200011)17:11<585::AID-ROB1>3.3.CO;2-Z.