

RESEARCH ARTICLE

# Time-optimal path following for non-redundant serial manipulators using an adaptive path-discretization

Tobias Marauli\* , Hubert Gattringer and Andreas Müller 

Institute of Robotics, Johannes Kepler University Linz, Altenberger Strasse 69, 4040 Linz, Austria

\*Corresponding author. E-mail: [tobias.marauli@jku.at](mailto:tobias.marauli@jku.at)

**Received:** 16 November 2022; **Revised:** 26 January 2023; **Accepted:** 4 February 2023; **First published online:** 13 March 2023

**Keywords:** serial manipulators, kinematics, dynamics, motion planning, singularities, optimization, second-order cone programming

## Abstract

The time-optimal path following (OPF) problem is to find a time evolution along a prescribed path in task space with shortest time duration. Numerical solution algorithms rely on an algorithm-specific (usually equidistant) sampling of the path parameter. This does not account for the dynamics in joint space, that is, the actual motion of the robot, however. Moreover, a well-known problem is that large joint velocities are obtained when approaching singularities, even for slow task space motions. This can be avoided by a sampling in joint space, where the path parameter is replaced by the arc length. Such discretization in task space leads to an adaptive refinement according to the nonlinear forward kinematics and guarantees bounded joint velocities. The adaptive refinement is also beneficial for the numerical solution of the problem. It is shown that this yields trajectories with improved continuity compared to an equidistant sampling. The OPF is reformulated as a second-order cone programming and solved numerically. The approach is demonstrated for a 6-DOF industrial robot following various paths in task space.

## 1. Introduction

Short duration times are highly requested to maximize the productivity of a robotic system. Therefore, time-optimal motion is desirable. Frequently, the task for a robot is defined by a path that should be followed. Examples are welding and painting robots, CNC machines as well as autonomous vehicles. Thus, several approaches for following paths in a (time-) optimal way are made in the literature. Most of the methods exploit that the motion along a prescribed path is described with a parameter  $\sigma$  and its time derivative  $\dot{\sigma}$ . This reduces the multi-dimensional state of a robot to a two-dimensional state-space in the optimization process. A simple, but computationally expensive approach is made in refs. [1–3], where dynamic programming is used to solve the path following problem. In refs. [4, 5], a less time expensive approach is presented, where the optimal solution is found by moving along extremals in the  $\sigma - \dot{\sigma}$  plane. Moreover, the (time-) optimal path following problem can be transformed to a convex optimal control problem through a nonlinear change of variables [6] which is computationally less time-consuming than the approaches mentioned previously. A further approach is made in ref. [7], where the convex problem is reformulated as second-order cone programming. It is reported that the calculation time compared to the convex optimal control problem is reduced even further. Since latter approach comes with low computation times, it will be used to solve the (time-) optimal path following problem in this paper.

However, the prescribed path is often deduced from process requirement (e.g., welding, polishing) as end effector (EE) path without taking singularities into account. Therefore, the inverse kinematics problem has to be solved in order to receive the corresponding joint paths. Most of the approaches for solving an optimal path following problem discussed in the literature utilize an equidistant discretization

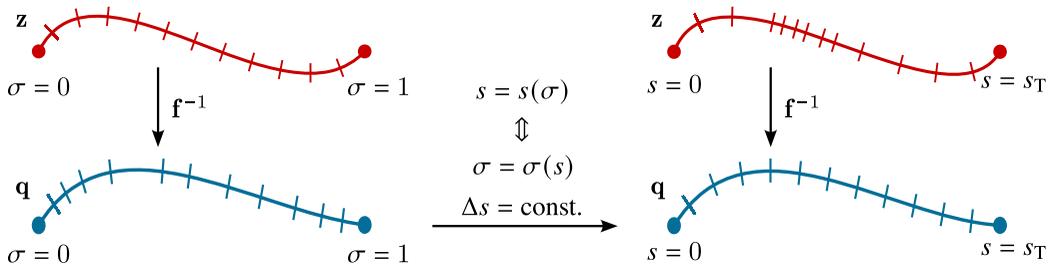


Figure 1. Principle of the adaptive sampling scheme.

of  $\sigma$  to solve the optimization problem. Thus, the question of using a non-equidistant sampling strategy in  $\sigma$  which copes with the presence of kinematic singularities arises.

In this paper, an adaptive sampling strategy which copes with kinematic singularities is addressed. The basic idea of such a sampling scheme is depicted in Fig. 1. An arbitrary EE-path  $\mathbf{z}(\sigma)$  with the path parameter  $\sigma$  and uniformly spread samplings over the whole path is assumed to be given. The inverse kinematics problem is denoted with  $\mathbf{q} = \mathbf{f}^{-1}(\mathbf{z})$ , and the solution of this problem leads to the corresponding joint path  $\mathbf{q}(\sigma)$ . Due to the nonlinear relation  $\mathbf{f}^{-1}$ , the distance of two corresponding joint values is not equidistant anymore. Especially near kinematic singularities latter values tend to vary drastically, due to the ill-posed inverse kinematics problem. This distance can be measured by the joint arc length  $s$ . With the knowledge of the nonlinear map  $s = s(\sigma)$  between  $s$  and  $\sigma$ , the joint path and thus the EE-path can be expressed as  $\mathbf{q}(s)$  and  $\mathbf{z}(s)$ . Prescribing a constant discretization of the joint arc length  $\Delta s$ , the discrete values of  $\mathbf{q}$  are spread equally over the new path parameter  $s$  in the interval  $[0, s_T]$ . Since  $s$  is a measure of distance in the joint coordinates and the distance of two consecutive joint values is kept constant ( $\Delta s = \text{const.}$ ), the reverse effect near singularities is applied to the sampling of the EE-path. Thus, the drastically varying joint values with  $\sigma$  as parameter induce an accumulation of samples in the EE-path near kinematic singularities with the parameter  $s$ , as illustrated in Fig. 1. The resulting non-equidistant sampling in  $\sigma$  is derived by  $\Delta\sigma_i = \sigma(s_{i+1}) - \sigma(s_i)$ .

The adaptive sampling scheme is derived from the joint arc length parameterized inverse kinematics, derived in ref. [8]. Therefore, the inverse kinematics problem for the (1) parameterization of an EE-path  $\mathbf{z}(\sigma)$  in terms of the path parameter  $\sigma$  [9, 10] and (2) the parameterization of the joint motion  $\mathbf{q}(s)$  in terms of the (kinetic) arc length  $s$  is discussed briefly. Each of these parameterizations gives rise to a particular inverse kinematics problem. The two parameterizations refer exclusively to the geometry of motion and are independent of the temporal execution of the motion.

## 2. Path kinematics

### 2.1. Forward kinematics

Denote with  $\mathbf{q} \in S^n$  the vector of joint variables and with  $\mathbf{z} \in SE(3)$  the EE-pose. In the course of the entire paper, the EE-path is supposed to be given and parameterized in terms of a path parameter  $\sigma(t) \in [0, 1]$ . Because the joint motion is derived from the EE-motion, the joint coordinates are also parameterized in terms of the path parameter  $\sigma$ . For motion planning and control, the latter is typically used. It should be mentioned that the relation of path parameter in workspace and joint space is not unique in general. With slight abuse of notation, the same symbol is used throughout the paper. The geometric forward kinematics map

$$\mathbf{z} = f(\mathbf{q}) \tag{1}$$

determines the EE-pose for prescribed joint coordinates  $\mathbf{q}$  [11]. The EE-twist, containing the translation and angular velocity, is determined by the velocity forward kinematics

$$\mathbf{V}(\mathbf{q}, \dot{\mathbf{q}}) = \begin{pmatrix} \mathbf{v} \\ \boldsymbol{\omega} \end{pmatrix} = \mathbf{J}(\mathbf{q}) \dot{\mathbf{q}} \tag{2}$$

where  $\mathbf{J}(\mathbf{q}) = \partial \mathbf{V} / \partial \dot{\mathbf{q}}$  is the geometric Jacobian and  $\dot{(\ )} = d(\ ) / dt$  the total time derivative. Since the EE-twist is linear in the joint rates and as  $\dot{\mathbf{q}} = \mathbf{q}' \dot{\sigma}$ , the velocity forward kinematics (2) can be defined in terms of  $\dot{\sigma}$ . This then yields to

$$\mathbf{V} = \frac{\partial \mathbf{V}}{\partial \dot{\sigma}} \dot{\sigma} = \mathbf{J} \mathbf{q}' \dot{\sigma}, \tag{3}$$

with  $(\ )' = \partial(\ ) / \partial \sigma$  denoting the geometric derivative. The latter results in  $\mathbf{V}(\sigma, \dot{\sigma}) = \mathbf{h}(\sigma) \dot{\sigma}$  and the relation

$$\mathbf{h}(\sigma) = \mathbf{J}(\mathbf{q}(\sigma)) \mathbf{q}'(\sigma) \tag{4}$$

where  $\mathbf{h}(\sigma) := \partial \mathbf{V} / \partial \dot{\sigma}$  is the *partial path velocity*. The relation (4) is referred to as the *velocity forward path-kinematics problem* and plays a central rule in the singularity-robust adaptive discretization. The EE-path  $\mathbf{z}(\sigma)$  and the partial path velocity  $\mathbf{h}(\sigma)$  originates from a path planning. Further, the path speed  $\dot{\sigma}$  and thus  $\mathbf{V}(\sigma, \dot{\sigma})$  are determined by solving an optimal path following problem as described in Section 4.

### 2.2. Inverse path kinematics

The classical inverse kinematics problem is to determine joint coordinates for given an EE-pose  $\mathbf{z}$ . For particular robots, this can be solved analytically, for example, wrist-partitioned serial manipulators [12]. However, it can only be solved numerically in general. Various different computational methods were discussed in refs. [12–15], including nonlinear root-finding, fixed point iteration, or integration of the joint velocity. This paper pursues the approach of integrating the joint velocity numerically. This amounts to solving the relation (2) for  $\mathbf{q}(t)$  with given EE-twist  $\mathbf{V}(t)$ . The EE-twist is determined by the partial path velocity with (3) when the path is parameterized with  $\sigma(t)$ . Then, the curve, and thus  $\mathbf{q}$ , is uniquely determined by (4), where the partial path velocity  $\mathbf{h}(\sigma)$  describes the motion along the path.

The *velocity inverse path-kinematics problem* determines  $\mathbf{q}'$  for given partial path velocity  $\mathbf{h}$ . A solution to the velocity inverse path kinematics is

$$\mathbf{q}' = \mathbf{J}(\mathbf{q})^+ \mathbf{h} \tag{5}$$

where  $\mathbf{J}(\mathbf{q})^+$  is a generalized pseudo inverse [16, 17]. The *geometric inverse path-kinematics problem* is to determine the joint motion  $\mathbf{q}(\sigma)$  in terms of the path parameter  $\sigma$ . Solving the initial value problem (IVP) (5) with  $\mathbf{q}(0) = \mathbf{q}_0$  as initial value corresponds to solving the inverse path-kinematics problem. When solving the IVP (5) numerically, numerical drifts are unavoidable. These drifts can be reduced by stabilizing a error system. Thus, the partial path velocity error is introduced as

$$\mathbf{e}' = \mathbf{h}_d - \mathbf{J}(\mathbf{q}) \mathbf{q}' \tag{6}$$

with  $\mathbf{h}_d$  denoting the prescribed partial path velocity. Therewith, the system (5) is amended as

$$\mathbf{q}' = \mathbf{J}(\mathbf{q})^+ (\mathbf{h}_d + \mathbf{K} \mathbf{e}) \tag{7a}$$

$$\mathbf{0} = \mathbf{e}' + \mathbf{K} \mathbf{e} \tag{7b}$$

where (7b) governs the error dynamics. The latter asymptotically converges to zero for  $\mathbf{K} > 0$  [18]. The geometric error  $\mathbf{e}$  in (7b) comprises the position error and orientation error and can be expressed as

$$\mathbf{e} = \begin{pmatrix} \mathbf{e}_p \\ \mathbf{e}_o \end{pmatrix} = \begin{pmatrix} \mathbf{r}_d - \mathbf{r}(\mathbf{q}) \\ \frac{1}{2}(\tilde{\mathbf{n}}(\mathbf{q})\mathbf{n}_d + \tilde{\mathbf{s}}(\mathbf{q})\mathbf{s}_d + \tilde{\mathbf{a}}(\mathbf{q})\mathbf{a}_d) \end{pmatrix} \tag{8}$$

where  $\mathbf{r}_d$  is the prescribed EE-position and  $\mathbf{R}_d = [\mathbf{n}_d \ \mathbf{s}_d \ \mathbf{a}_d]$  the prescribed EE-orientation, with columns  $\mathbf{n}, \mathbf{s}, \mathbf{a}$  [19]. The skew-symmetric (cross product) matrix of a vector  $\mathbf{x} \in \mathbb{R}^3$  is denoted with  $\tilde{\mathbf{x}} \in \mathbb{R}^{3,3}$ . The velocity inverse path-kinematics problem can be solved numerically stable using time integration schemes, with the use of latter definitions and appropriate  $\mathbf{K}$ . Shortly, a parameterization of the joint coordinates  $\mathbf{q}(\sigma)$  in terms of the parameter  $\sigma$  is attained with the solution of (5).

### 2.3. Joint arc length inverse path kinematics

The inverse path-kinematics problem is ill-posed near singularities. This leads to numerical problems and also causes extremely fast and undesirable joint motions. Following a path with constant speed  $\dot{\sigma}_0$ , for instance, leads to an EE-motion with a well-defined bounded EE-velocity. Particularly at singularities of the forward kinematics, the corresponding inverse kinematics solution  $\mathbf{q}(t)$  will not have well-defined bounds and the joint rate  $\dot{\mathbf{q}}(t)$  tends to infinity. As a consequence, when prescribing the EE-motion by an equidistant sampling of the path parameter  $\sigma$ , the ‘distance’ of joint coordinates  $\mathbf{q}$  corresponding to two consecutive values of  $\sigma$  tend to vary drastically. A measure of ‘distance’ between two consecutive joint values is given by the arc length  $s$  defined by the differential line element  $ds = \sqrt{d\mathbf{q}^T \mathbf{W} d\mathbf{q}}$  with a metric  $\mathbf{W}$  (using the generalized mass matrix  $\mathbf{M}$  of the robot would yield the kinetic line element and ensure homogeneity). The arc length gives rise to a well-defined discretization of the curve in joint space in terms of the step size. For the rest of this paper,  $\mathbf{W}$  is chosen as identity matrix since the considered robot has revolute joints only.

Expressing the EE-motion  $\mathbf{z}(s)$  in terms of the arc length  $s$  and thus derive parameterizations  $\sigma = \sigma(s)$  and  $\mathbf{q} = \mathbf{q}(s)$  is referred to as the arc length path-kinematics problem. It is known from the theory of metric curves [20] that there is no such explicit relation in general. Similar to the inverse path kinematics, Section 2.2, latter problem is formulated as an ordinary differential equation (ODE). Integrating the line element  $ds$  with given joint path  $\mathbf{q}(\sigma)$ , the arc length in joint is [21]

$$s = \int_0^\sigma \|\mathbf{q}'\|_w d\xi \tag{9}$$

with  $\|\mathbf{q}'\|_w := \sqrt{\mathbf{q}'^T \mathbf{W} \mathbf{q}'}$  denoting the *joint path speed*. Introducing the jacobian (of above equation)  $\partial s / \partial \sigma = \|\mathbf{q}'\|_w$  in (4) yields

$$\frac{\partial \sigma}{\partial s} = \frac{1}{\|\mathbf{J}^+ \mathbf{h}\|_w} = \frac{1}{\|\mathbf{q}'\|_w}, \quad \frac{\partial \mathbf{q}}{\partial s} = \frac{\mathbf{J}^+ \mathbf{h}}{\|\mathbf{J}^+ \mathbf{h}\|_w} = \frac{\mathbf{q}'}{\|\mathbf{q}'\|_w} \tag{10}$$

with  $\mathbf{q}' = \mathbf{J}^+ \mathbf{h}$ . This ODE system can be solved with given initial conditions  $\sigma_0$  and  $\mathbf{q}_0$  for  $s = 0$  and a given parameter interval  $s \in [0, s_T]$ . However, this requires the knowledge of the terminal arc length  $s_T$ , which is unknown a priori. The terminal arc length has to satisfy the condition

$$\sigma(s_T) = 1. \tag{11}$$

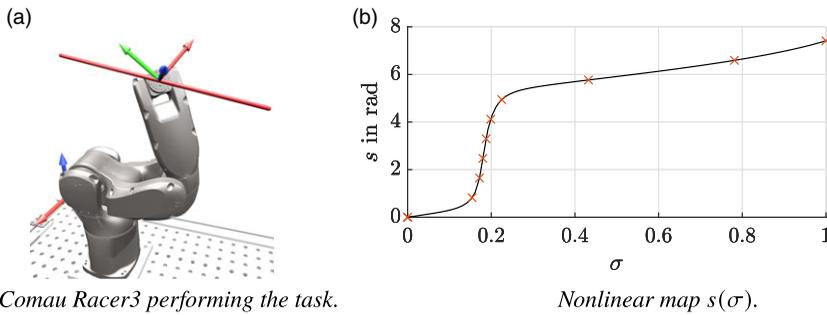
Latter condition can act as a termination criterion to terminate the integration. System (10) is then solved numerically, until  $\sigma(s) = 1$  is satisfied. This can be carried out numerically stable using integration schemes with zero-crossing event detection using advanced root-finding methods. In practice, the integration is terminated when  $\sigma > 1$ . It should be noted that the solution  $\sigma(s)$  is only used to determine the terminal arc length  $s_T$ .

The numerical drift of the arc length parameterized inverse path-kinematics subsystem (10) again has to be stabilized. As previously discussed for the standard geometric inverse path-kinematics problem, this is attained by introducing the path velocity error

$$\frac{\partial \mathbf{e}}{\partial s} = \frac{\partial \mathbf{h}_d}{\partial s} - \mathbf{J} \frac{\partial \mathbf{q}}{\partial s}, \tag{12}$$

with  $\partial \mathbf{h}_d / \partial s = \mathbf{h}'_d / \|\mathbf{q}'\|_w$ . The system (10) is amended as

$$\frac{\partial \mathbf{q}}{\partial s} = \frac{\mathbf{J}^+ \mathbf{h}}{\|\mathbf{J}^+ \mathbf{h}\|_w} + \mathbf{J}^+ \mathbf{K} \mathbf{e} \tag{13a}$$



**Figure 2.** 6-DOF Comau Racer3 following (a) a straight line in task space with constant orientation and (b) nonlinear relation between  $s$  and  $\sigma$ .

$$\mathbf{0} = \frac{\partial \mathbf{e}}{\partial s} + \mathbf{K}\mathbf{e} \tag{13b}$$

where (13b) governs the error dynamics of the error (8). With  $\mathbf{K} > 0$ , asymptotically convergence to zero is ensured.

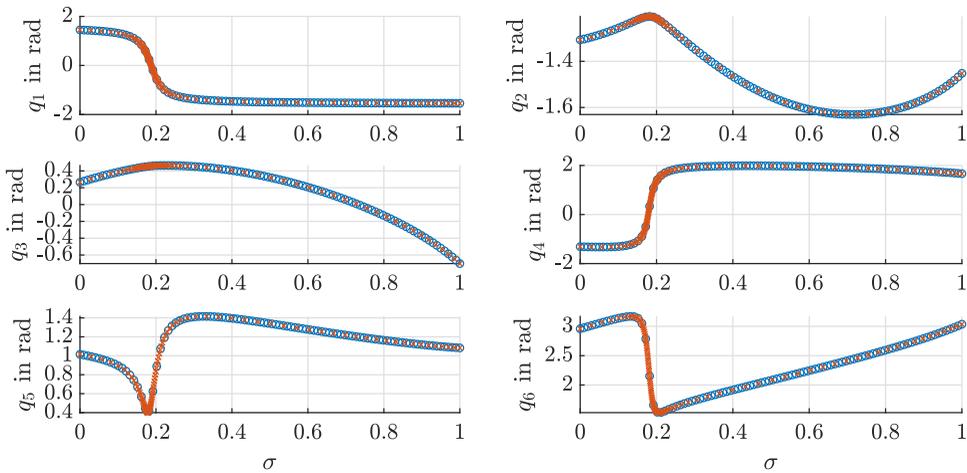
For numerical solving, cases where  $\mathbf{q}' = \mathbf{0}$  occurs have to be taken into account, due to the division by  $\|\mathbf{q}'\|_w$ . Since  $\mathbf{q}' = \mathbf{J}^+\mathbf{h}$  two cases arise. First, the case when  $\mathbf{h} = \mathbf{0}$  is hold. This is avoided by assuming a regular EE-path so that always  $\mathbf{h} \neq \mathbf{0}$ . Secondly, the partial path velocity  $\mathbf{h}$  is an element of the null space  $\mathbf{h} \in \ker \mathbf{J}^+$ . This results in EE movements which do not have any effect on the joint coordinates, which is a contradiction to the forward kinematics.

### 3. An adaptive singularity-consistent sampling scheme

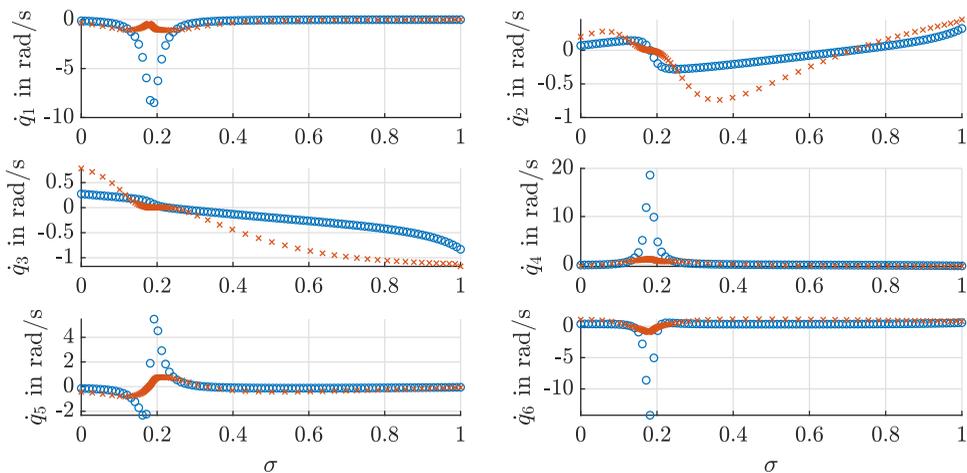
Based on the velocity inverse path-kinematics problem, the adaptive sampling approach is developed. The EE-path with parameter  $\sigma$  is assumed to be equivalent to the curve parameterized with parameter  $s$ . As mentioned before, prescribing the EE-motion by an equidistant sampling  $\Delta\sigma$  inevitably causes drastically varying joint coordinates  $\mathbf{q}$ , of two consecutive samples, when solving the inverse path-kinematics problem (5). The arc length in joint space (9) serves as a measure of the ‘distance’ between two consecutive values of  $\mathbf{q}$  and is a nonlinear relation of  $s$  and  $\sigma$ . For a 6-DOF Comau Racer3 robot following a straight line in workspace Fig. 2(a) with constant orientation and speed  $\dot{s}_0$ , the nonlinear mapping  $s(\sigma)$  with an equidistant sampling  $\Delta s$  is shown in Fig. 2(b) for illustration purposes. As the drastically increase of the joint arc length indicates, the manipulator passes near a singularity at  $\sigma \in [0.15, 0.25]$ .

The joint coordinate  $\mathbf{q}$  evolution and the nonlinear relation between  $s$  and  $\sigma$  are both preserved by parameterizing the prescribed path using the joint arc length  $s$ , Section 2.3. Prescribing a constant arc length speed  $\dot{s}_0$  and equidistant sampling for  $s$  for following the trajectory, a varying speed  $\dot{\sigma}$  due to the nonlinear mapping (9) is induced. As a consequence of the relation, smaller speeds  $\dot{\sigma}$  are achieved at higher rates  $\partial s / \partial \sigma$  leading to more sampling points in  $\sigma$  (locally). Therefore, lower joint velocities are expected in such regions since  $\dot{\mathbf{q}} = \mathbf{q}'\dot{\sigma}$ . Furthermore, this leads to an adaptive sampling strategy for  $\sigma$  derived from the nonlinear  $s(\sigma)$ . The latter discretization scheme can then be applied for solving optimal path following problems, see Section 5.

In the following, a comparison of the two presented inverse path kinematics is made. Therefore, the functions  $\sigma(t)$  and  $s(t)$  are assumed to be given with constant time derivatives  $\dot{\sigma}(t) = \dot{\sigma}_0$  and  $\dot{s}(t) = \dot{s}_0$ , respectively. To get comparable results between the two parameterizations, the terminal time  $t_T$  is fixed. For  $t_T = 5$  s, the EE follows the prescribed path with (1) constant speed  $\dot{\sigma}_0 = 0.2 \text{ s}^{-1}$  and equidistant



**Figure 3.** Samples of the joint coordinates  $\mathbf{q}$ . Blue circles: (5) for the inverse path kinematics with constant speed  $\dot{\sigma}_0$ . Orange crosses: (10) for the inverse path kinematics with constant speed  $\dot{s}_0$ .



**Figure 4.** Resulting joint velocities  $\dot{\mathbf{q}}$ . Blue circles: (5) for the inverse path kinematics with constant speed  $\dot{\sigma}_0$ . Orange crosses: (10) for the inverse path kinematics with constant speed  $\dot{s}_0$ .

sampling in  $\sigma$  when using the velocity inverse path kinematics (5), and (2) constant arc length speed  $\dot{s}_0 = 1.482 \text{ rad/s}^{-1}$  with equidistant discretization in  $s$  for the arc length parameterized velocity inverse path kinematics (10). The resulting joint paths using both inverse path kinematics are depicted in Fig. 3. It can be seen that both strategies lead to the same evolution of the joint coordinates. The arc length parameterized solution also shows an increased number of samples at  $\sigma \in [0.15, 0.25]$ , which indicates the passing near a singularity. Viewing at the evolution of  $q_5$ , it is observed that the value is going towards zero in this region. This indicates the vicinity to the wrist singularity.

As discussed above, for fixed step size  $\Delta s$  (which means fixed time step size due to constant  $\dot{s}_0$ ), higher sampling rates of the joint coordinates in this area are induced, and hence, lower joint velocities  $\dot{\mathbf{q}} = \mathbf{q}'\dot{\sigma}$  are expected, which is illustrated in Fig. 4. This is particularly pronounced for  $\dot{q}_4$ . The joint velocity  $\dot{q}_4$  using (10) is more than ten times lower than the solution obtained with the standard inverse path kinematics (5) at  $\sigma \in [0.15, 0.25]$ . Moreover, the resulting trajectories are smoother compared to using

(5) for the inverse path kinematics. Apparently, this is a consequence of the accumulation of samples at  $\sigma \in [0.15, 0.25]$ , as indicated previously.

#### 4. Time-optimal path following problem

In this section, the optimal path following problem is described. The equations of motion (EOM) as

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{g}(\mathbf{q}, \dot{\mathbf{q}}) = \boldsymbol{\tau} \tag{14}$$

describe the dynamic behavior of the manipulator, where  $\mathbf{M}(\mathbf{q})$  is the mass matrix and  $\boldsymbol{\tau}$  are the joint torques. The Coriolis, centrifugal, and gravitational terms as well as the Coulomb and viscous friction are described by the vector  $\mathbf{g}(\mathbf{q}, \dot{\mathbf{q}})$ . If not necessary, the dependencies are omitted for simplicity and readability.

The objective of an optimal path following problem is finding a time profile  $\sigma(t)$  for a prescribed path in either the joint variables  $\mathbf{q}(\sigma)$  or the EE  $\mathbf{z}(\sigma)$ , while satisfying some constraints. Latter are, for example, the dynamic behavior (14), limiting the joint velocity  $\dot{\mathbf{q}}$ , acceleration  $\ddot{\mathbf{q}}$  and jerk  $\dddot{\mathbf{q}}$ . Therefore, a description depending only on path allocated variables, such as geometrical derivatives ( $'$ ) and the path speed  $\dot{\sigma}(\sigma)$ , is desired. Given an EE-path  $\mathbf{z}(\sigma)$ , the geometrical joint derivatives have to be computed with the inverse path kinematics, as discussed in Section 2.2. The first derivative  $\mathbf{q}'$  is given by (5). The higher derivatives are derived by higher order inverse path kinematics, that is, geometrical derivatives of (4), which yields

$$\mathbf{q}'' = \mathbf{J}(\mathbf{q})^+ (\mathbf{h}' - \mathbf{J}'\mathbf{q}') \tag{15a}$$

$$\mathbf{q}''' = \mathbf{J}(\mathbf{q})^+ (\mathbf{h}'' - 2\mathbf{J}'\mathbf{q}'' - \mathbf{J}''\mathbf{q}'). \tag{15b}$$

For the optimization, it is beneficial to introduce the squared path speed  $z := \dot{\sigma}^2$ . By assuming the path is followed only in forward direction  $\dot{\sigma} \geq 0$ , a one to one map between  $\dot{\sigma}$  and  $z$  exists. Therefore, no case separation has to be taken into account. Introducing the variable  $z$  leads to the relations as

$$\dot{\sigma} = \sqrt{z}, \quad \ddot{\sigma} = \frac{z'}{2}, \quad \dddot{\sigma} = \frac{z''}{2}\sqrt{z} \tag{16}$$

for higher time derivatives. With these relations and the higher order inverse path kinematics, the joint velocity, acceleration, and jerk can be expressed as follows

$$\dot{\mathbf{q}} = \mathbf{q}'\sqrt{z} \tag{17a}$$

$$\ddot{\mathbf{q}} = \mathbf{q}''z + \frac{1}{2}\mathbf{q}'z' \tag{17b}$$

$$\dddot{\mathbf{q}} = \mathbf{q}'''z\sqrt{z} + \frac{3}{2}\mathbf{q}''z'\sqrt{z} + \frac{1}{2}\mathbf{q}'z''\sqrt{z}. \tag{17c}$$

The path parameterization of the EOM (14) is derived by introducing (17a) and (17b) as

$$\mathbf{a}(\sigma)z' + \mathbf{b}(\sigma)z + \mathbf{c}(\sigma) + \mathbf{d}(\sigma)\sqrt{z} = \boldsymbol{\tau}(\sigma). \tag{18}$$

with  $\mathbf{a} = 1/2\mathbf{M}\mathbf{q}'$  and  $\mathbf{b}$  including the projected Coriolis and centrifugal terms as well as  $\mathbf{M}\mathbf{q}''$ . A path-dependent approximation of the Coulomb friction and gravitational forces are composed in  $\mathbf{c}$ , and  $\mathbf{d}$  denotes the path projected viscous friction. For receiving time-optimal solutions, the terminal time  $t_T$  has to be minimized. The time-optimal path following (OPF) problem is formulated as

**Table I.** Numerical values of the upper bounds  $\bar{(\cdot)}$ .

Joint	$\bar{\dot{q}}$ in rad/s	$\bar{\ddot{q}}$ in rad/s <sup>2</sup>	$\bar{\overset{\text{---}}{q}}$ in rad/s <sup>3</sup>	$\bar{\tau}$ in Nm
1	6.912	100	4500	327.681
2	7.069	100	4500	320.400
3	7.854	100	4500	144.504
4	9.425	100	4500	61.139
5	9.425	100	4500	61.139
6	14.137	100	4500	39.960

$$\min_{z(\sigma)} \int_0^1 \frac{1}{\sqrt{z}} d\sigma$$

s.t.

$$z(0) = \dot{\sigma}_0^2, \quad z(1) = \dot{\sigma}_T^2$$

$$z(\sigma) \geq 0$$

$$\underline{\dot{q}} \leq \dot{q}(\sigma) \leq \bar{\dot{q}}, \quad \underline{\ddot{q}} \leq \ddot{q}(\sigma) \leq \bar{\ddot{q}}$$

$$\underline{\overset{\text{---}}{q}} \leq \overset{\text{---}}{q}(\sigma) \leq \bar{\overset{\text{---}}{q}}, \quad \underline{\tau} \leq \tau(\sigma) \leq \bar{\tau}$$

$$\text{for } \sigma \in [0, 1], \tag{19}$$

with the corresponding path description of the objective function [22], the joint constraints (17), and the EOM (18). The lower and upper bounds of the constraints are denoted by  $\underline{(\cdot)}$ ,  $\bar{(\cdot)}$ . Solving the above problem yields an optimal squared path speed profile  $z^*$ . The trajectories of the joint velocity, acceleration, jerk, and torque, then can be computed using the relations (17) and (18). The optimization problem can be solved using for example, dynamic programming [1], numerical integration [23], a sequential convex programming (SCP) approach [24], or with second-order cone programming (SOCP), and a SCP approach for non-convex constraints [7].

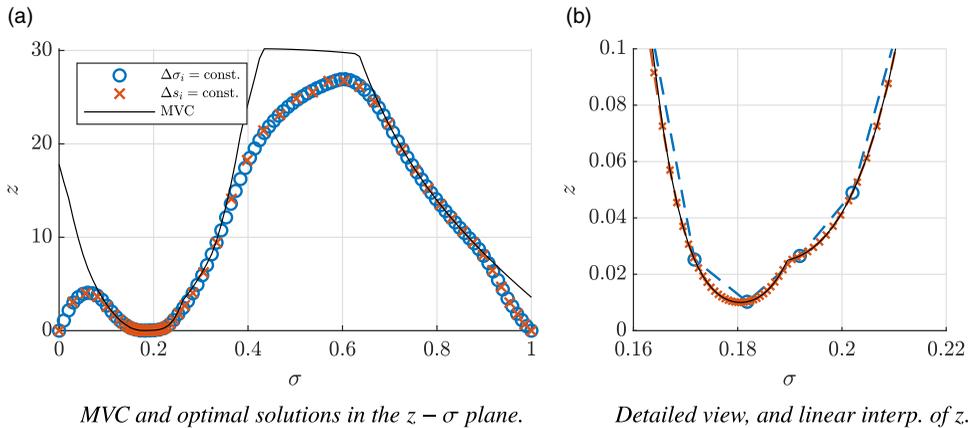
## 5. Examples

### 5.1. Optimization framework and robotic manipulator

We used an Intel(R) Core(TM) i5-9500 CPU @ 3.00 GHz CPU running on Windows10 for all computations. For a 6-DOF *Comau Racer3* robotic manipulator, the OPF problem (19) is solved in SOCP form. Neglecting the jerk and torque constraints temporary, (19) is a convex optimization problem. This convex problem is reformulated as SOCP and solved with a SCP approach for considering the joint jerk and torque constraints, as presented in ref. [7]. In order to reduce the optimization problem (19) to a finite discrete problem, the path parameter is discretized with  $N$  values  $\sigma_i, i \in \{0, \dots, N - 1\}$ . Also, a linear interpolation profile for the squared path speed  $z$  in an interval  $[\sigma_i, \sigma_{i+1}]$  is used [6].

For more details, the reader is referred to the corresponding literature, since rewriting the problem is not topic of this paper [6, 7, 25]. The SOCP form of the optimization problem was implemented in Matlab using YALMIP [26]. As solver, the conic solver MOSEK [27], a tailored solver for SOCPs, was picked. For simplicity, the upper and lower bounds of (19) are chosen to be symmetrical that is,  $\underline{(\cdot)} = -\bar{(\cdot)}$ . These values are listed in Table I.

In order to solve the optimization problem, a discretization strategy for  $\sigma$  has to be chosen. Therefore, (1) an equidistant sampling  $\Delta\sigma_i = \text{const.}$  which acts as benchmark strategy and (2) a non-equidistant sampling  $\Delta\sigma_i \neq \text{const.}$  derived from the singularity-consistent sampling scheme, Section 3, is used. Thus, the non-equidistant discretization is given by  $\Delta\sigma_i = \sigma(s_{i+1}) - \sigma(s_i)$ . For the rest of the paper, a



**Figure 5.** (a) Optimal speed  $z^*$  for following the straight line. (b) Detailed view at the section near the singularity. Black curve: MVC. Blue circles: equidistant sampling scheme in  $\Delta\sigma_i = \text{const.}$  Orange crosses: non-equidistant sampling scheme in  $\Delta\sigma_i = \sigma(s_{i+1}) - \sigma(s_i)$ , with  $\Delta s_i = \text{const.}$

uniform discretization in  $s$  ( $\Delta s_i = \text{const.}$ ) is used to compute the non-equidistant sampling  $\Delta\sigma_i \neq \text{const.}$  Figure 2(b) exemplary shows such a discretization for the straight line in workspace. Clearly, a non-equidistant sampling of the joint arc length  $s$  could be used to derive the non-equidistant sampling in  $\sigma$ . However, developing such a discretization in  $s$  would not be beneficial, since the latter sampling could be derived directly for  $\sigma$ .

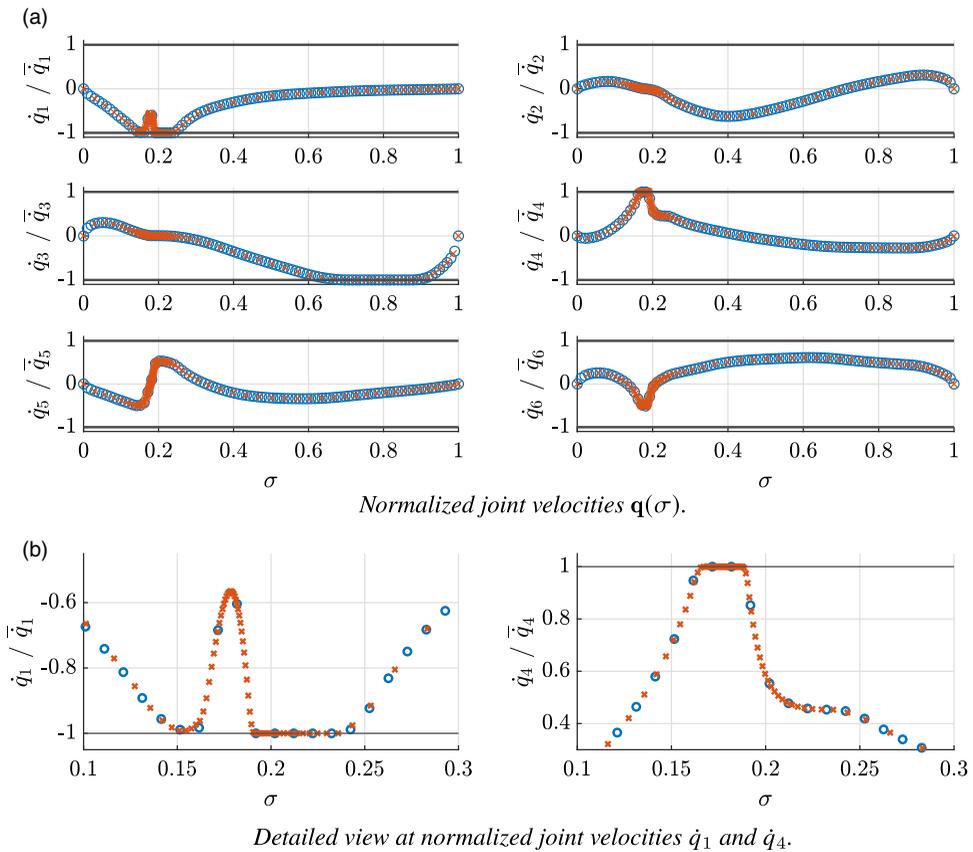
To evaluate the quality of the optimal solution  $z^*$ , the maximum velocity curve (MVC) [4] is used. This curve describes the highest possible speed  $z$  along the path. The MVC is derived by computing the intersection of joint velocity, acceleration, and torque constraints in the  $z' - z$  plane and choosing the intersection point with the smallest value of  $z$ .

**5.2. Results**

*5.2.1. Straight line*

As the first example, the straight line near a singularity in the task space is selected, Fig. 2(a). The corresponding joint coordinates computed with an inverse path kinematics, Section 2, are shown in Fig. 3 as seen before. Solving the OPF problem (19) for the sampling schemes discussed earlier with  $N = 100$  samples yields the optimal solutions  $z^*$  in Fig. 5(a). Both strategies lead to nearly the same evolution  $z^*(\sigma)$ . Also, the passing near a singularity at  $\sigma \in [0.15, 0.25]$  is reflected, since the MVC and thus the optimal solutions become nearly zero in this area. The non-equidistant sampling, marked with the small orange crosses, again shows accumulated sampling points in this region as discussed earlier. As a consequence, the resulting trajectories are expected to be smoother compared to the equidistant discretization, as discussed previously in Section 3. Moreover, the constraint violation in an interval of two consecutive sampling points, when using the linear interpolation profile for  $z$  is much lower. This is pointed out in Fig. 5(b) especially at  $\sigma \in [0.17, 0.18]$ .

The resulting optimal joint velocities in Fig. 6(a) show that  $\dot{q}_4$  is limited by the velocity constraint, when close to the singularity. Due to the vicinity to the wrist singularity ( $q_5 \rightarrow 0$ ), see Fig. 3 at  $\sigma \approx 0.2$ , either  $\dot{q}_4$  or  $\dot{q}_6$  was expected to fulfill the velocity constraint. In the remaining section of passing near the singularity,  $\dot{q}_1$  is limited by the constraint. Thus, the evolution of  $\dot{q}_1$  and  $\dot{q}_4$  at this region is viewed in more detail, Fig. 6(b). As mentioned before, the joint velocities are much smoother in this section, with the non-equidistant sampling compared to the equidistant sampling of  $\sigma$ . This, besides lower constraint violation when using the linear interpolation profile in  $z$ , inevitable leads to smoother higher derivatives



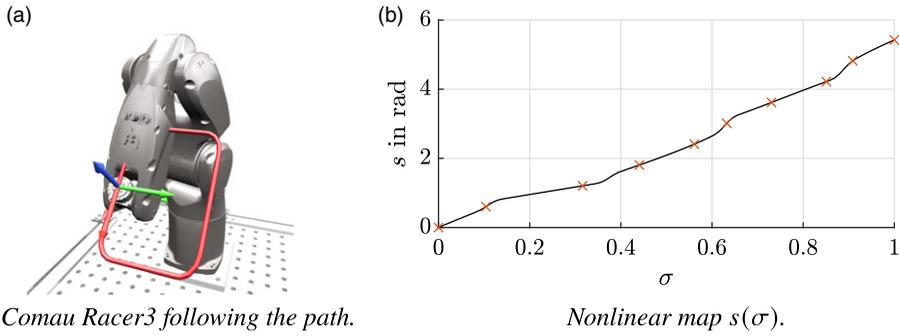
**Figure 6.** (a) Resulting optimal joint velocity trajectory  $\dot{\mathbf{q}}(\sigma)$  for the straight line. (b) Detailed view at the velocities of joint 1 and 4 near the singularity. Blue circles: equidistant sampling scheme  $\Delta\sigma_i = \text{const}$ . Orange crosses: non-equidistant sampling  $\Delta\sigma_i = \sigma(s_{i+1}) - \sigma(s_i)$ , with  $\Delta s_i = \text{const}$ .

that is, joint acceleration and jerk with the adaptive sampling strategy. The impact of the non-equidistant sampling of  $\sigma$  on terminal and computation time is covered in Section 5.2.3.

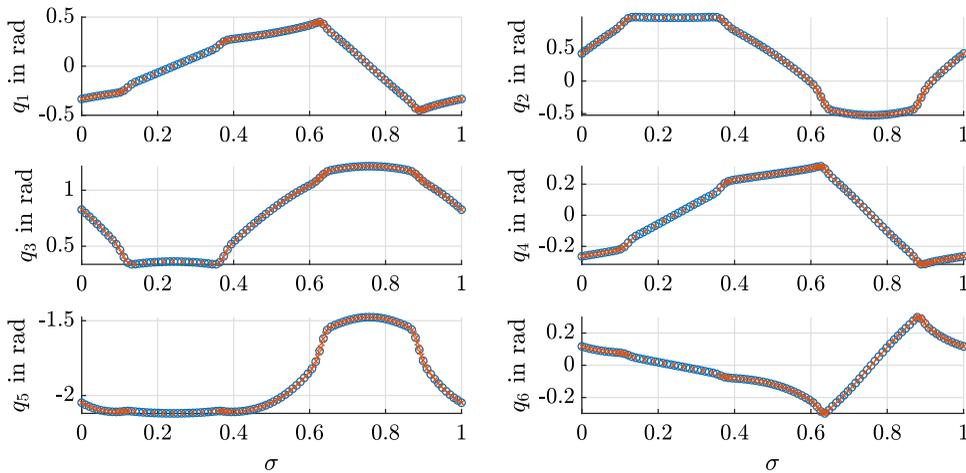
5.2.2. Rectangular path with rounded corners

The following EE-path is inspired by the international norm ISO-9283 for industrial robot calibration and performance testing. The norm involves different geometrical paths in the workspace, such as circles, straight lines, and a rectangle with rounded corners. For performance tests, the various paths are passed in a sequence. In this paper, the rectangle with rounded corners was picked. This shape was preferred of two reasons: First, because the curve has a higher geometrical complexity compared to the circle and the straight line (which is already covered in Section 5.2.1); second, because the length of this path is longer compared to the other shapes. The rectangle is placed in front of the robot with an angle of 45° to the vertical axes, Fig. 7(a). The path is passed in counterclockwise direction, starting at the point shown in the figure. Again the nonlinear map  $s(\sigma)$  with an equidistant sampling in  $s$  is presented in Fig. 7(b), to illustrate the non-equidistant sampling scheme  $\Delta\sigma_i = \sigma(s_{i+1}) - \sigma(s_i)$ . Compared to the relation from the straight line, the map is approximately piece-wise linear. Thus, the samples of the non-equidistant scheme are pronounced in regions with high slopes and vice versa.

Figure 8 shows the corresponding joint paths space using both inverse path kinematics discussed in Section 2. Viewing the evolution of the joint coordinates and the nonlinear map  $s(\sigma)$ , it is evident



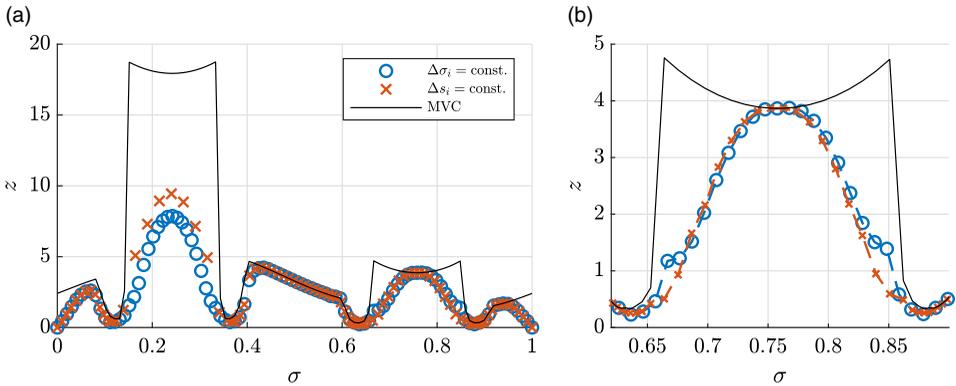
**Figure 7.** 6-DOF Comau Racer3 following (a) a rectangle with rounded corners in task space with constant orientation, and (b) nonlinear relation between  $s$  and  $\sigma$ .



**Figure 8.** Corresponding joint coordinates  $\mathbf{q}$  of the rectangular EE-path. Blue circles: solution (5) for the inverse path kinematic with constant speed  $\dot{\sigma}_0$ . Orange crosses: solution (10) for arc length parameterized inverse path kinematics with constant arc length speed  $\dot{s}_0$ .

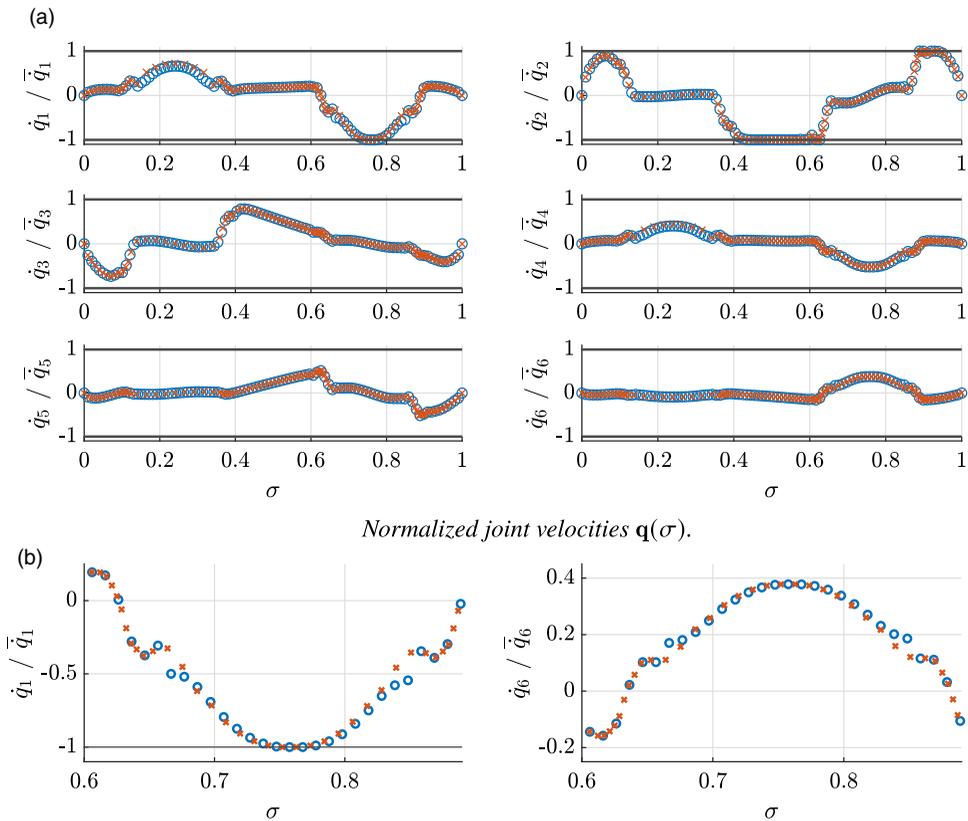
that the robot is not passing near a singularity. Furthermore, the lower straight line of the rectangle at  $\sigma \in [0.15, 0.35]$  leads to mainly linear joint paths. This is especially pronounced for  $q_1, q_4$  and  $q_6$ . As a consequence of the linear joint evolution, the number of samples, using  $\Delta\sigma_i = \sigma(s_{i+1}) - \sigma(s_i)$ , is reduced in this region. Since following simple geometries, for example, linear paths, does not require much sampling points, the latter behavior is even desirable. At  $\sigma \in [0.65, 0.85]$ , the robot is passing the upper line of the rectangle, again resulting in mainly linear joint coordinates  $q_1, q_4$  and  $q_6$ , but with a higher gradient as for the lower line. Thus, more samples are present for this linear path.

Solving (19) for the rectangle with both sampling schemes yields to the solution illustrated in Fig. 9(a). At the rounded corners of the rectangle, only lower path speeds can be achieved. This is reflected in the MVC at  $\sigma \in \{0.16, 0.38, 0.65, 0.9\}$ . Both solutions again share nearly the same evolution over the curve parameter  $\sigma$ . The highest deviations are present in the section  $\sigma \in [0.15, 0.35]$ , which corresponds to the lower straight line of the rectangle mentioned before. Due to the fact that the sampling points are not the same  $\sigma_i \neq \sigma(s_i)$ , the constraints in the optimization problem are evaluated at different values of  $\sigma$ . Therefore, constraints become active or inactive at a different progression of the path, effecting the further evolution. In this case, even higher path speeds  $z^*$ , for the lower straight line, are achieved with the adaptive sampling strategy  $\Delta\sigma_i = \sigma(s_{i+1}) - \sigma(s_i)$ . Figure 9(b) shows the  $z - \sigma$  plane of the upper straight line at  $\sigma \in [0.65, 0.85]$  in more detail. The number of samples in this section is quite



MVC and the two optimal solutions in the  $z - \sigma$  plane. Detailed view, and linear interp. of  $z$ .

**Figure 9.** (a) Optimal speed  $z^*$  for following the rectangle. (b) Detailed view at the MVC of the upper straight line. Black curve: MVC. Blue circles: equidistant sampling scheme in  $\Delta\sigma_i = \text{const.}$  Orange crosses: non-equidistant sampling scheme in  $\Delta\sigma_i = \sigma(s_{i+1}) - \sigma(s_i)$ , with  $\Delta s_i = \text{const.}$



Normalized joint velocities  $\mathbf{q}(\sigma)$ . Detailed view at normalized joint velocities  $\dot{q}_1$  and  $\dot{q}_6$ .

**Figure 10.** (a) Resulting optimal joint velocity trajectory  $\dot{\mathbf{q}}(\sigma)$  of the rectangular path. (b) Detailed view at the velocities of joint one and six at  $\sigma \approx 0.65 \dots 0.85$ . Blue circles: equidistant sampling scheme  $\Delta\sigma_i = \text{const.}$  Orange crosses: non-equidistant sampling  $\Delta\sigma_i = \sigma(s_{i+1}) - \sigma(s_i)$ , with  $\Delta s_i = \text{const.}$

**Table II.** Terminal and computation times as well as their relative differences compared to the equidistant sampling. Denoting with (1) the equidistant sampling  $\Delta\sigma_i = \text{const.}$ , (2) the non-equidistant discretization  $\Delta\sigma_i = \sigma(s_{i+1}) - \sigma(s_i)$  with  $\Delta s_i = \text{const.}$ , and with (A) the straight line, (B) the rectangular path, (C) the meander-shaped EE-path and (D) the arbitrary spatial path.

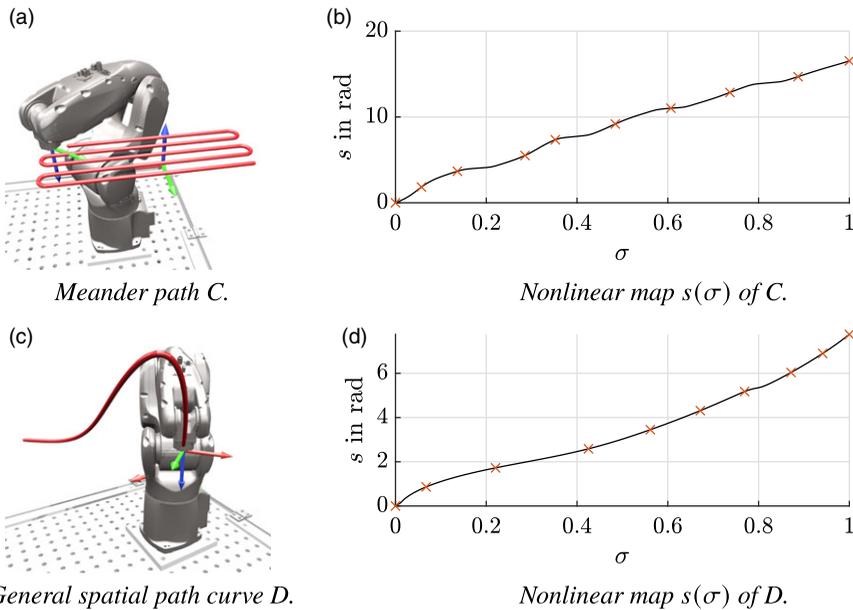
			N = 100		N = 200		N = 500	
			(1)	(2)	(1)	(2)	(1)	(2)
(A)	$t_T$	in s	0.739	0.761	0.754	0.761	0.759	0.760
	$t_{\text{comp.}}$	in s	12.335	8.959	19.211	17.21	53.731	49.262
	$p_{t_T}$	in %	–	–3.000	–	–0.885	–	–0.170
	$p_{t_{\text{comp.}}}$	in %	–	27.375	–	10.417	–	8.317
(B)	$t_T$	in s	0.858	0.844	0.865	0.865	0.865	0.864
	$t_{\text{comp.}}$	in s	7.041	7.011	14.850	14.850	40.293	40.211
	$p_{t_T}$	in %	–	1.683	–	0.000	–	0.100
	$p_{t_{\text{comp.}}}$	in %	–	0.432	–	0.000	–	0.204
(C)	$t_T$	in s	1.936	1.986	1.956	1.976	1.961	1.967
	$t_{\text{comp.}}$	in s	9.000	6.742	13.753	11.924	24.348	23.264
	$p_{t_T}$	in %	–	–2.592	–	–0.920	–	–0.353
	$p_{t_{\text{comp.}}}$	in %	–	25.102	–	13.230	–	4.453
(D)	$t_T$	in s	0.824	0.820	0.825	0.823	0.825	0.824
	$t_{\text{comp.}}$	in s	6.836	6.363	12.810	12.042	40.775	38.476
	$p_{t_T}$	in %	–	0.538	–	0.222	–	0.063
	$p_{t_{\text{comp.}}}$	in %	–	6.903	–	5.977	–	5.639

similar for both strategies. Viewing the transition between the rounded corners and the straight line at the beginning and end of this section, it is observed that the non-equidistant sampling scheme leads to a smoother optimal path speed  $z^*$ . Thus, the joint velocities, accelerations, and jerks are expected to be smoother at this transition areas.

As mentioned above, the adaptive sampling results in higher values of  $z$  at  $\sigma \in [0.65, 0.85]$ . Thus, higher joint velocities are achieved as can be seen in Fig. 10(a), particularly for  $\dot{q}_1$  and  $\dot{q}_4$ . This results in a faster traversal of the lower straight line with  $\Delta\sigma_i = \sigma(s_{i+1}) - \sigma(s_i)$ . In Fig. 10(b), a detailed view of passing the upper straight line is depicted for the joint velocities  $\dot{q}_1$  and  $\dot{q}_6$ . As previously indicated, the transition between the rounded corners and the straight line at  $\sigma \approx 0.65$  and  $\sigma \approx 0.85$  is smoother for the non-equidistant discretization.

5.2.3. Computation time vs. terminal time

In this section, the terminal time  $t_T$  and computation time  $t_{\text{comp.}}$  are compared. The corresponding time values are depicted in Table II, for different number of samples  $N$  and four EE-paths. The viewed paths include the straight line and rectangle as discussed above as well as a meander-shaped EE-path and an arbitrary spatial curve. The latter two paths are illustrated in Fig. 11, with their corresponding nonlinear map  $s(\sigma)$ . In addition, the relative difference  $p_{t_T}$  and  $p_{t_{\text{comp.}}}$  compared to the equidistant discretization strategy in  $\sigma$  is also listed in the table. Thus, negative numbers reflect a loss in time (higher values of  $t_T$ ) and vice versa. First, the results A) of the straight line are viewed. The adaptive sampling has an overall positive impact on the computation time for this path. This is especially pronounced for  $N = 100$  with a reduction of 27%, compared to the equidistant sampling. Viewing the terminal time a loss in time is observed. This can be explained due to the fact that the integral of (19) is approximated numerically in



**Figure 11.** Illustration of (a) the meander path  $C$  and (c) arbitrary spatial path in the workspace  $D$ , as well as their nonlinear relations between  $s$  and  $\sigma$  in (b), (d), respectively.

the optimization process. Therefore, the size of a sampling interval  $\Delta\sigma_i$  noticeably influences the quality of the approximation. Since the number of samples  $N$  is fixed for the whole path, a raise of samples in one section reduces the number of samples for the remaining curve in return. For the straight line, most of the samples are accumulated in the region near the singularity, see for example, Fig. 5. Therefore, the sampling intervals  $\Delta\sigma_i$  of the remaining path are larger on average, for the adaptive sampling strategy. This results in a poor approximation of the time interval in sections not near the kinematic singularity which leads to an increase or decrease of the terminal time  $t_T$ . In the case of the straight line, the poor approximation induces an increased terminal time, compared to the equidistant sampling scheme.

Comparing the terminal and computation time of (B) the rectangle with rounded corners, the adaptive sampling strategy barely impacts both. For (C) the meander-shaped EE-path, Fig. 11(a), the behavior is quite the same as for (A) the straight line. The computation time of the optimal solution is reduced for all three numbers of samples, but on the contrary the terminal time is increased when using the non-equidistant sampling scheme. The strategy  $\Delta\sigma_i = \sigma(s_{i+1}) - \sigma(s_i)$  comes with a slight improvement of both the terminal  $t_T$  and computation time  $t_{comp}$ , for (D) the arbitrary curve, Fig. 11(c).

### 6. Conclusion

The inverse kinematics problem was expressed in terms of the arc length  $s$  in joint space when the EE-motion is parameterized by a path parameter  $\sigma$ . This leads to an ODE system for  $\mathbf{q}(s)$  and  $\sigma(s)$  which is solved for  $s \in [0, s_T]$ . It is shown that using the arc length as independent parameter (instead of  $\sigma$ ) allows for coping with kinematic singularities. Moreover, the arc length parameterized velocity inverse path-kinematics solution (10) gives rise to an adaptive sampling scheme.

This derived discretization strategy was used to solve the optimal path following problem (19) with SOCP. The solution was discussed in detail for two exemplary EE-paths. The optimization results, with the non-equidistant sampling, were compared with an equidistant sampling in  $\sigma$ . It is shown that the adaptive sampling strategy leads to smoother trajectories near singularities and geometrically

challenging paths (in joint space). The non-equidistant discretization also copes well with the presence of simple geometries, where the sampling rate is reduced, see Section 5.2.2.

Furthermore, the impact on the terminal and computation time was discussed in detail. It was observed that non-equidistant sampling leads to an overall reduction of the computation time. In return, the terminal time shows slightly increased values.

Finally, it should be mentioned that the derived adaptive sampling strategy has to be investigated in detail for every prescribed EE-path since the inverse path kinematics, and thus, the non-equidistant sampling is highly depending on the shape of the EE-path and position in the workspace.

**Author contributions.** All authors contributed equally to this article.

**Financial support.** This research was funded in whole, or in part, by the Austrian Science Fund (FWF) [I 4452-N].

**Competing interests.** The authors declare that they have no conflicts of interest.

**Ethical approval.** Not applicable.

## References

- [1] D. Kaserer, H. Gattringer and A. Müller, “Nearly optimal path following with jerk and torque rate limits using dynamic programming,” *IEEE Trans. Robot.* **35**(2), 521–528 (2019).
- [2] M. Oberherber, H. Gattringer and A. Müller, “Successive dynamic programming and subsequent spline optimization for smooth time optimal robot path tracking,” *Mech. Sci.* **6**(2), 245–254 (2015).
- [3] K. Shin and N. McKay, “A dynamic programming approach to trajectory planning of robotic manipulators,” *IEEE Trans. Automat. Contr.* **31**(6), 491–500 (1986).
- [4] F. Pfeiffer and R. Johanni, “A concept for manipulator trajectory planning,” *IEEE J. Robot. Automat.* **3**(2), 115–123 (1987).
- [5] J.-J. E. Slotine and H. S. Yang, “Improving the Efficiency of Time-optimal Path-following Algorithms,” **In:** *American Control Conference 1988*(1988) pp. 2129–2134.
- [6] D. Verscheure, B. Demeulenaere, J. Swevers, J. De Schutter and M. Diehl, “Time-optimal path tracking for robots: A convex optimization approach,” *IEEE Trans. Automat. Contr.* **54**(10), 2318–2327 (2009).
- [7] F. Debrouwere, *Optimal Robot Path Following Fast Solution Methods for Practical Non-convex Applications* (KU Leuven, 2015). <https://lirias.kuleuven.be/1693024?limo=0>
- [8] T. Marauli, H. Gattringer and A. Müller, “Singularity Robust Inverse Kinematics of Serial Manipulators by Means of a Joint Arc Length Parameterization,” **In:** *Advances in Service and Industrial Robotics 2022* (A. Müller and M. Brandstötter, eds.) (Springer International Publishing, 2022).
- [9] M. Breivik and T. Fossen, “Path Following for Marine Surface Vessels,” **In:** *Oceans '04 MTS/IEEE Techno-Ocean '04 (IEEE Cat. No.04CH37600) 2004*, vol. 4, pp. 2282–2289.
- [10] A. Hladio, C. Nielsen and D. Wang, “Path following for a class of mechanical systems,” *IEEE Trans. Contr. Syst. Technol.* **21**(6), 2380–2390 (2013).
- [11] B. Siciliano and O. Khatib, *Springer Handbook of Robotics*, Chapter Kinematics,” **In:** (Springer, 2016) pp. 11–36.
- [12] A. Aristidou and J. Lasenby, *Inverse Kinematics: A Review of Existing Techniques and Introduction of a New Fast Iterative Solver* (2009).
- [13] A. A. Goldenberg, B. Benhabib and R. Fenton, “A complete generalized solution to the inverse kinematics of robots,” *IEEE J. Robot. Automat.*, **1**(1), 14–20 (1985).
- [14] A. Reiter, A. Müller and H. Gattringer, “On higher-order inverse kinematics methods in time-optimal trajectory planning for kinematically redundant manipulators,” *IEEE Trans. Ind. Inform.* **14**(4), 1681–1690 (2018).
- [15] S. C. Thomopoulos and R. Y. Tam, “An iterative solution to the inverse kinematics of robotic manipulators,” *Mech. Mach. Theory* **26**(4), 359–373 (1991).
- [16] A. Ben-Israel and T. N. Greville. *Generalized Inverses: Theory and Applications*, vol. 15 (IEEE, Manhattan, NY, 2003).
- [17] R. Penrose. “A Generalized Inverse for Matrices,” **In:** *Mathematical Proceedings of the Cambridge Philosophical Society*(1955) pp. 406–413.
- [18] B. Siciliano, L. Sciacivico, L. Villani and G. Oriolo. *Robotics* (IEEE, Manhattan, NY, 2009).
- [19] B. Siciliano, “A closed-loop inverse kinematic scheme for on-line joint-based robot control,” *Robotica* **8**(3), 231–243 (1990).
- [20] A. Lastra. *Parametric Geometry of Curves and Surfaces* (University of Cambridge, Department of Engineering, Cambridge, 2021).
- [21] M. P. Do Carmo. *Differential Geometry of Curves and Surfaces: Revised and Updated Second Edition* (IEEE, Manhattan, NY, 2016).
- [22] Z. Shiller and S. Dubowsky. “On the Optimal Control of Robotic Manipulators with Actuator and End-effector Constraints,” **In:** *Proceedings of the 1985 IEEE International Conference on Robotics and Automation*, vol. 2 (1985) pp. 614–620.

- [23] J. Mattmüller and D. Gisler, “Calculating a near time-optimal jerk-constrained trajectory along a specified smooth path,” *Int. J. Adv. Manufact. Technol.* **45**(9), 1007–1016 (2009).
- [24] F. Debrouwere, W. Van Loock, G. Pipeleers, Q. T. Dinh, M. Diehl, J. De Schutter and J. Swevers. “Time-Optimal Path Following for Robots with Trajectory Jerk Constraints using Sequential Convex Programming,” **In:** *IEEE International Conference on Robotics and Automation 2013* (2013) pp. 1916–1921, ISSN: 1050-4729.
- [25] T. P. Dinh and H. A. L. Thi, “A D.C. optimization algorithm for solving the trust-region subproblem,” *SIAM J. Optim.* **8**(2), 476–505 (1998).
- [26] J. Löfberg. “Yalmip : A Toolbox for Modeling and Optimization in Matlab,” **In:** *Proceedings of the CACSD Conference 2004*, Taipei, Taiwan (2004).
- [27] M. ApS, The MOSEK Optimization Toolbox for MATLAB manual. Version 9.3.21 (2022).