10.24425/123460

Archives of Control Sciences
Volume 28(LXIV), 2018
No. 2*, pages 269–283*

Motion planning for nonholonomic systems with earlier destination reaching

ADAM RATAJCZAK

The motion planning problem consists in finding a control function which drives the system to a desired point. The motion planning algorithm derived with an endogenous configuration space approach assumes that the motion takes place in an arbitrary chosen time horizon. This work introduces a modification to the motion planning algorithm which allows to reach the destination point in time, which is shorter than the assumed time horizon. The algorithm derivation relies on the endogenous configuration space approach and the continuation (homotopy) method. To achieve the earlier destination reaching a new formulation of the task map and the task Jacobian are introduced. The efficiency of the new algorithm is depicted with simulation results.

Key words: motion planning, nonholonomic, endogenous configuration space, homotopy, continuation, earlier destination reaching

1. Introduction

According to [8, Formulation 14.1] the motion planning task relies on finding a control function which leads the system from a certain initial position to a desired destination with the required accuracy. Moreover, there is an assumption that the time is unbounded $t \in [0, \infty)$ and exists some termination action where all the velocities in the system are equal to zero. Finally, there exists some $t > 0$ where the desired point is reached and the current controls are equal to the termination actions.

The motion planning algorithms derived from the endogenous configuration space approach [15] take the time interval as an input parameter, which have to be bounded. The resultant motion needs the whole time interval to reach the destination point. The selection of the particular value of the time horizon is relatively inconvenient. It should take into account the resultant path length, the

A. Ratajczak is with Department of Control Systems and Mechatronics, Wrocław University of Science and Technology, Janiszewski str. 11/17, 50-372 Wrocław, Poland, E-mail: adam.ratajczak@pwr.edu.pl.

This research was supported by the Wrocław University of Science and Technology under a statutory research project.

Received 26.01.2017.

values of control function etc. In this paper, we present a new motion planning algorithm which is able to reach the destination earlier, namely in shorter time than the selected time interval.

The new algorithm will be also relayed on the endogenous configuration space approach. This method unifies the theory of robotic manipulators with the theory of nonholonomic systems. So, many methods and intuitions from holonomic systems could be transferred to nonholonomic cases. The derivation of the motion planning algorithms based on the endogenous configuration space approach relies on the continuation (homotopy) method [4, 5] which is still an active tool for the construction of the motion/trajectory planning algorithms [1, 2, 7]. Some other recent results of nonholonomic motion planning can be found in [3, 6, 9].

This work is an extension of the conference paper [10], where the earlier destination reaching feature was implemented with an outline of prioritarian multiple task motion planning algorithm [11]. However, this paper introduces a new algorithm which is no longer a kind of a multiple task, but it contains only one task which is able to plan the motion with earlier destination reaching. In the recent years, a little more research has been performed to influence on the solution of planning algorithms based on the endogenous configuration space approach [13, 17, 18].

The remaining part of this paper is the following. In section 2 the motion planning with earlier destination reaching problem is defined. The preliminary material is presented in 3. The section 4 introduces the new motion planning algorithm with earlier destination reaching. The simulation results are presented in 5. The section 6 concludes the paper. There is also an Appendix A which collects the details of the algorithm derivations.

2. Problem definition

Let us consider a control affine system with an output function

$$\begin{cases} \dot{q} = f(q) + G(q)u = f(q) + \sum_{i=1}^{m} g_i(q)u_i, \\ y = k(q), \end{cases} \tag{1}$$

where $q \in \mathbf{R}^n$ is a state space vector, $u \in \mathbf{R}^m$ denotes the control vector and $y \in \mathbf{R}^r$ stands for the task space vector. The $f(q)$ is a drift term, and the control matrix $G(q)$ collects the vector fields $g_i(q)$.

For the system (1) the motion planning problem with earlier destination reaching could be defined in following way. Find a control function $u(t)$ which drives the system from an initial state $q(0) = q_0$ to a desired point y_d in a task space in time shorter than assumed interval $[0,T]$. For the classic motion planning problem derived with the endogenous configuration space approach the desired point is reached at the end of the time horizon $t \in [0,T]$. Here, we introduce a modification to this assumption in which the desired point will be reached at $t_k < T$. Such a problem relies on the minimization of the following integral

$$\min_{u(\cdot)} \int_0^T H\big(\varepsilon(t)\big)\mathrm{d}t, \tag{2}$$

where $\varepsilon(t) = y(t) - y_d$ is an error in the task space. Such definition (2) of a problem could be treated as a vector–valued practical optimization problem. We assume that the problem is solved when all the variables reach the 0 value with assumed accuracy. The particular formula of the function $H\big(\varepsilon(t)\big)$ will be introduced later.

3. Preliminaries

The problem stated in the previous section will be solved within an endogenous configuration space approach [15]. For this aim, we will introduce the necessary components.

The admissible control functions $u(t) \in \mathbf{R}^m$ in system (1) will be assumed Lebesque square integrable on an interval $t \in [0,T]$. Then, the control space $\mathscr{U} = L^2_m[0,T]$ together with the inner product $\langle u_1(\cdot), u_2(\cdot)\rangle = \int_0^T u_1^{\mathrm{T}}(t)u_2(t)\,\mathrm{d}t$ and the induced norm forms a Hilbert space called the endogenous configuration space. The endogenous configuration space is sometimes called the control space. Every element of control space is a set of m control functions defined on the interval $[0,T]$, which can be applied to (1) to obtain a system trajectory $q(\cdot)$. It is worth to mention that for the nonholonomic system (1) the trajectory depends not only on the control function but also on the initial condition q_0. For this purpose we introduce a flow of the system (1) as $q(t) = \varphi_{q_0,t}\big(u(\cdot)\big)$ which could be interpreted as the state of the system at time t, started from the initial state $q(0) = q_0$ and driven by the control functions $u(\cdot)$ over the horizon $[0,t]$. Every admissible control function $u(\cdot) \in \mathscr{U}$ corresponds to the state trajectory $q(t)$ and the task space trajectory $y(t) = k\big(q(t)\big)$. Let us call this input–output relation $K_{q_0,t} : \mathscr{U} \to \mathbf{R}^r$ as an instantaneous map and define as follows

$$K_{q_0,t}\big(u(\cdot)\big) = y(t) = k\Big(\varphi_{q_0,t}\big(u(\cdot)\big)\Big). \tag{3}$$

In the robotics nomenclature this relation could be identified with the kinematics of nonholonomic systems [15]. The differentiation of the kinematics yields a Jacobian $J_{q_0,t}: \mathscr{U} \to \mathbf{R}^r$ expressed with the following formula [15]

$$J_{q_0,t}\big(u(\cdot)\big)v(\cdot) = \mathscr{D}K_{q_0,t}\big(u(\cdot)\big)v(\cdot) = \frac{\mathrm{d}}{\mathrm{d}\alpha}\bigg|_{\alpha=0} K_{q_0,t}\big(u(\cdot)+\alpha v(\cdot)\big)$$
$$= C(t)\int_0^t \Phi(t,s)B(s)v(s)\,\mathrm{d}s = \eta. \tag{4}$$

The Jacobian determines how the infinitesimal change of $v(\cdot) \in \mathscr{U}$ in the control space influences on change of $\eta \in \mathbf{R}^r$ in the task space. The matrices appearing in (4) comes from the linear system associated with (1)

$$\begin{cases} \dot{\xi}(t) = A(t)\xi(t) + B(t)v(t), \\ \eta(t) = C(t)\xi(t), \end{cases} \tag{5}$$

which is also a linear approximation of system (1) along a control–trajectory $\big(u(t), q(t)\big)$ pair. The mentioned matrices can be computed from the following formulas

$$\begin{aligned} A(t) &= \frac{\partial\big(f(q)+G(q)u\big)}{\partial q}, \\ B(t) &= \frac{\partial\big(f(q)+G(q)u\big)}{\partial u} = G(q), \\ C(t) &= \frac{\partial\big(k(q)\big)}{\partial q}. \end{aligned} \tag{6}$$

The matrix $\Phi(t,s)$ is a transition matrix [14] of the linear system (5) and satisfies the differential equation $\frac{\partial}{\partial t}\Phi(t,s) = A(t)\Phi(t,s)$ with final condition $\Phi(s,s) = I_n$.

4. Motion planning algorithm with earlier destination reaching

Having defined the kinematics (3) and Jacobian (4) of the nonholonomic system (1) we can move to the derivation of the motion planning algorithm with earlier destination reaching. The problem introduced in section 2 can be now rewritten as

$$\min_{u(\cdot)} \int_0^T H\Big(K_{q_0,t}\big(u(\cdot)\big) - y_d\Big)\,\mathrm{d}t. \tag{7}$$

To minimize the criterion (7) we define the task map $\mathbb{K}_{q_0,T}\colon \mathscr{U} \to \mathbf{R}^r$ as follows

$$\mathbb{K}_{q_0,T}\big(u(\cdot)\big) = \int_0^T H\big(y(t)\big)\,\mathrm{d}t, \quad H\colon \mathbf{R}^r \to \mathbf{R}^r, \quad H(y) = \begin{bmatrix} h_1(y) \\ h_2(y) \\ \vdots \\ h_r(y) \end{bmatrix}. \tag{8}$$

Moreover, we also choose the task error equal to the task map (8), namely $e(T) = \mathbb{K}_{q_0,T}\big(u(\cdot)\big)$. To derive the algorithm formula we will need also the task Jacobian which we obtain by a differentiation of the task map (8)

$$\begin{aligned}
\mathbb{J}_{q_0,T}\big(u(\cdot)\big)v(\cdot) &= \mathscr{D}\mathbb{K}_{q_0,T}\big(u(\cdot)\big)v(\cdot) \\
&= \frac{\mathrm{d}}{\mathrm{d}\alpha}\Big|_{\alpha=0} \mathbb{K}\big(u(\cdot)+\alpha v(\cdot)\big) = \frac{\mathrm{d}}{\mathrm{d}\alpha}\Big|_{\alpha=0} \int_0^T H\Big(K_{q_0,t}\big(u(\cdot)\big) - y_d\Big)\mathrm{d}t \\
&= \int_0^T \frac{\partial H\big(y(t)\big)}{\partial y} \underbrace{C(t)\int_0^t \Phi(t,s)B(s)v(s)\,\mathrm{d}s}_{J_{q_0,t}\left(u(\cdot)\right)v(\cdot)}\,\mathrm{d}t.
\end{aligned} \tag{9}$$

To solve the motion planning problem with earlier destination reaching we enroll the homotopy (continuation) method [4, 5]. In order to achieve this aim we choose in the endogenous configuration space a smooth curve $u_\vartheta(\cdot)$ parametrized with $\vartheta \in \mathbf{R}$ and passing through the initial configuration (control) $u_0(\cdot)$ at $\vartheta = 0$. Along this curve we can compute the task error $e_\vartheta(T) = K_{q_0,T}\big(u_\vartheta(\cdot)\big)$. We also assume, that the task error $e_\vartheta(T)$ is vanishing exponentially when the continuation parameter $\vartheta \to \infty$. This fact could be expressed with the differential equation

$$\frac{\mathrm{d}e_\vartheta(T)}{\mathrm{d}\vartheta} = -\gamma e_\vartheta(T). \tag{10}$$

The parameter $\gamma > 0$ can be used to control the decay rate. The substitution of the task map (8) into the (10), and the enrolling the task Jacobian (9) leads us to the Ważewski–Davidenko equation

$$\mathbb{J}_{q_0,T}\big(u_\vartheta(\cdot)\big)\frac{\mathrm{d}u_\vartheta(\cdot)}{\mathrm{d}\vartheta} = -\gamma e_\vartheta(T). \tag{11}$$

The equation (11) can be solved using any right Jacobian inverse, in particular with a Moore–Penrose inverse

$$\left(\mathbb{J}^{\#}_{q_0,T}\big(u_\vartheta(\cdot)\big)\eta\right)(t) = B^{\mathrm{T}}(t)M^{\mathrm{T}}(t)P^{-1}(T)\eta, \tag{12}$$

where $\eta \in \mathbf{R}^r$ is an element from the task space, $M(t)$ can be obtained from the differential equation

$$\dot{M}(t) = -\frac{\partial H\big(y(t)\big)}{\partial y} - M(t)A(t) \tag{13}$$

with the final condition $M(T) = 0$. The matrix $P(T)$ is a solution of the following differential equation

$$\dot{P}(t) = M(t)B(t)B^T(t)M^T(t), \tag{14}$$

with the initial condition $P(0) = 0$. The detailed derivations of the above formulas are attached in the Appendix A. Finally, the equation (11) together with the Jacobian inverse (12) takes the form of dynamical system

$$\frac{\mathrm{d}\,u_\vartheta(t)}{\mathrm{d}\,\vartheta} = -\gamma\left(\mathbb{J}^{\#}_{q_0,T}\left(u_\vartheta(\cdot)\right)e_\vartheta(T)\right)(t) = B^{\mathrm{T}}(t)M^{\mathrm{T}}(t)P^{-1}(T)e_\vartheta(T), \tag{15}$$

which is a motion planning algorithm with earlier destination reaching. The solution, i.e. the wanted control function $u^*(t)$, is a limit $u^*(t) = \lim_{\vartheta\to\infty} u_\vartheta(t)$ of the resultant trajectory of (15).

The motion planning algorithm with earlier destination reaching task in [10] was performed as a task priority scheme, where the main task was the proper motion planning and as a secondary task was chosen an earlier destination reaching task. Contrary to [10], here, thanks to a specific structure of the function $H(y)$, there is only one task which reach earlier the destination point with assumed accuracy.

4.1. Specific form of the function $H\big(y(t)\big)$

Let us focus on the selection of the function $H\big(y(t)\big)$. As we previously assumed, we want the computed control $u^*(t)$ to drive the system (1) in such a way which minimizes the criterion (7). Moreover, as long as the algorithm is a motion planning algorithm we primarily want the desired destination y_d to be reached, at least at the end of time horizon T. The specific selection of the function $H\big(y(t)\big)$ should guarantee that the mentioned two features, namely: the destination reaching and a minimization of the criterion will be met. To do so the elements $h_i(y)$ of function $H\big(y(t)\big)$ should take positive values and have only one minimum at $y_i - y_{d_i}$. Regarding to those assumptions we chose the following three forms of the $H\big(y(t)\big)$ function

- Gaussian function

$$h_i(y) = 1 - e^{\frac{-(y_i - y_{d_i})^2}{2\sigma^2}}, \tag{16}$$

- Lorentzian function

$$h_i(y) = 1 - \frac{\sigma^2}{\sigma^2 + (y_i - y_{d_i})^2}, \tag{17}$$

- Quadratic function

$$h_i(y) = \frac{1}{2}\left(y_i - y_{d_i}\right)^2. \tag{18}$$

The plots of the above three types of selected functions $H\big(y(t)\big)$ are depicted in Fig. 1.

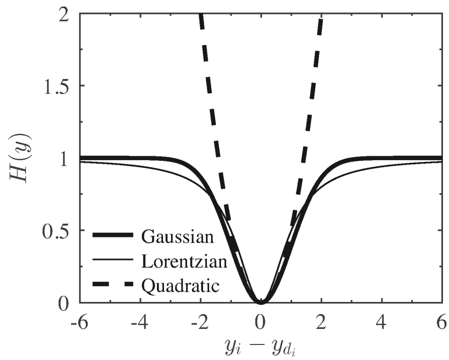

Figure 1: The three types of function $H(y)$

5. Simulations

In this section we will present the results of numerical simulation of the motion planning algorithm with earlier destination reaching for a model of the unicycle.

5.1. Unicycle

The schematic view and the meaning of all the variables are shown in the Fig. 2. The kinematics of such system can be expressed with following equations

$$\begin{cases} \dot{q} = G(q)u = \begin{bmatrix} \cos(q_3) & 0 \\ \sin(q_3) & 0 \\ 0 & 1 \end{bmatrix} u \\ y = k(q) = q, \end{cases} \tag{19}$$

where $q = (q_1, q_2, q_3) = (x, y, \theta) \in \mathbf{R}^3$ is a state space vector, x, y denote the position and θ the orientation of the unicycle. The controls belong to $u \in \mathbf{R}^2$ and have the meaning of the longitudinal and rotation speeds. The output function is set as an identity so $y = q$.

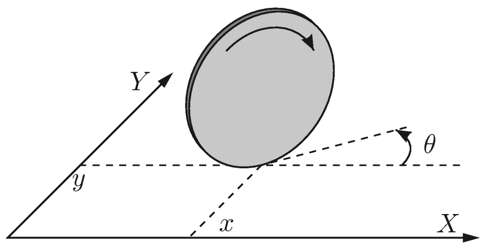

Figure 2: Unicycle

5.2. Organization of computation

To solve the motion planning problem with earlier destination reaching it is necessary to find a solution of the differential–algebraic system of equations composed of (19), (6), (13), (14) and (8), namely

$$\begin{cases} \dfrac{\mathrm{d}q_\vartheta(t)}{\mathrm{d}t} = f(q_\vartheta(t)) + G(q_\vartheta(t))u_\vartheta(t), \\ A_\vartheta(t) = \dfrac{\partial\Big(f\big(q_\vartheta(t)\big) + G\big(q_\vartheta(t)\big)u_\vartheta(t)\Big)}{\partial q}, \quad B_\vartheta(t) = G(q_\vartheta(t)), \\ \dfrac{\mathrm{d}M_\vartheta(t)}{\mathrm{d}t} = -\dfrac{\partial H\big(y(t)\big)}{\partial y} - M_\vartheta(t)A_\vartheta(t), \\ \dfrac{\mathrm{d}P_\vartheta(t)}{\mathrm{d}t} = M_\vartheta(t)B_\vartheta(t)B_\vartheta^{\mathrm{T}}(t)M_\vartheta^{\mathrm{T}}(t), \\ e_\vartheta(T) = \mathbb{K}_{q_0,T}(u_\vartheta(\cdot)), \end{cases} \tag{20}$$

for each value of ϑ, with initial conditions $q(0) = q_0$, $P(0) = 0$ and final condition $M(T) = 0$. The subscripts $(\cdot)_\vartheta$ in (20) mean that the function depends on the current trajectory of the system $\varphi_{x_0,t}\big(u_\vartheta(\cdot)\big)$ driven by the control function $u_\vartheta(\cdot)$. For every ϑ the resultant trajectories of (20) are applied to solve the algorithm equation (15), which now depends on particular trajectories

$$\frac{\mathrm{d}u_\vartheta(t)}{\mathrm{d}\vartheta} = -\gamma\left(\mathbb{J}^{\#}_{q_0,T}\big(u(\cdot)\big)e_\vartheta(T)\right)(t) = B_\vartheta^{\mathrm{T}}(t)M_\vartheta^{\mathrm{T}}(t)P_\vartheta^{-1}(T)e_\vartheta(T). \tag{21}$$

As it can be seen, there are two independent variables, t and ϑ, so the whole problem could be treated as a kind of partial differential equations. However, based on the specific structure of equations (20) and (21) they can be solved using two nested ordinary differential equation solvers. The inner solver gives a solution of the (20), while the outer computes the algorithm equation (21). Both solvers are built-in MATLAB solvers with variable step, and the demanded control function is computed in nonparametric version [12].

5.3. Simulation results

To show the efficiency of the new algorithm we will present the results of the numerical computations. The simulations were performed for three “earlier destination reaching” algorithms with all types of the $h\big(y(t)\big)$ functions introduced in section 4.1. Additionally, the result of the “classic” motion planning algorithm [16], based on the task map defined by (3) with Moore-Penrose Jacobian inverse of (4) is added for comparison purposes. The simulatively investigated problem is to find a control function which drives the unicycle (19) from an initial state $q_0 = (0,0,0)$ to a desired point $y_d = q_d = (5,5,0)$ in a shorter time than the horizon $T = 5$. The decay rate in (10) is set $\gamma = 1$, and the parameter of the $h_i\big(y(t)\big)$ function in (16) and (17) is equal $\sigma = 1$.

The results of the simulation are depicted in Figs. 3–7. The path in XY plane is shown in Fig. 3. One can observe, that the motion path is similar for each algorithm. However, only the “classic” solution does not overshoot the desired point.

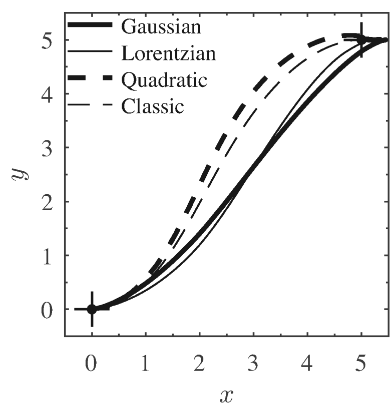

Figure 3: Path of the motion in XY plane

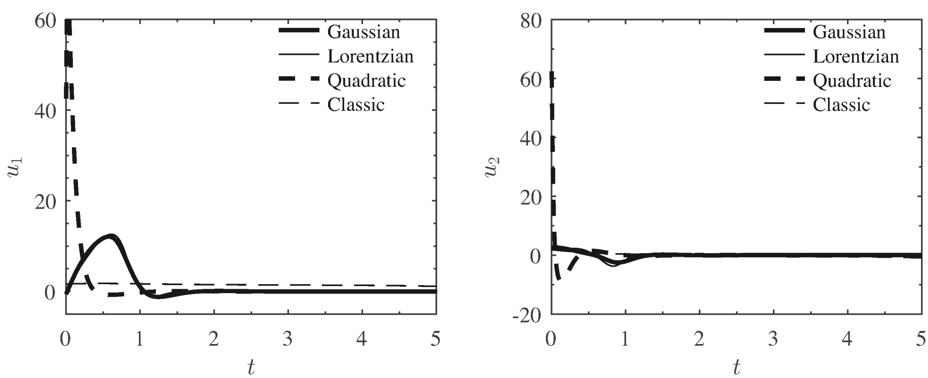

Figure 4: Control functions

It is related with the higher motion speeds for all “earlier destination reaching” algorithms. The control function plots are presented in Figs. 4 and 5 where the interesting regions are emphasized. As it can be expected, the earlier reaching of the destination results with the higher values of control functions. The control values for the quadratic function are very high, while the values for Gaussian or Lorentzian function take more practical values. For comparison, the controls ob-

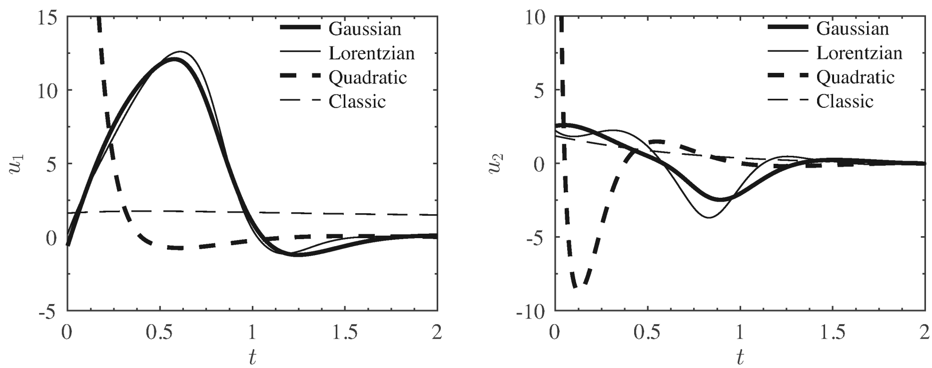

Figure 5: Control functions (interesting regions)

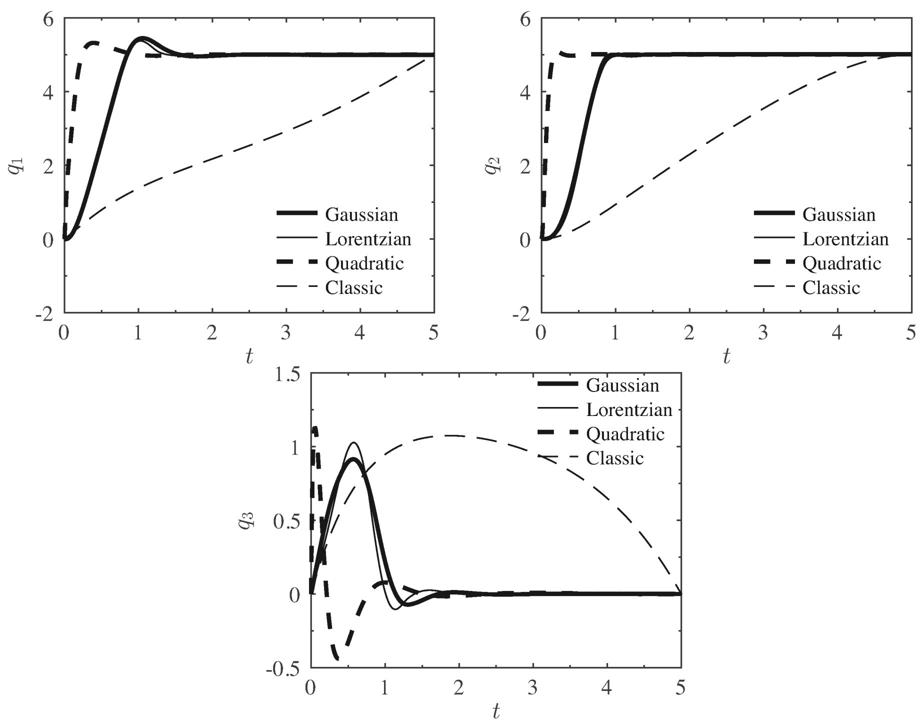

Figure 6: State trajectories

tained from "classic" algorithm take very small values. It is noteworthy that the controls for all "earlier destination reaching" algorithms take the zero value at the end of the motion, which means that the unicycle stops at the destination point. Fig. 6 presents the trajectories of the state vector elements. As it can be seen in the plots, the destination is reached earlier by the algorithm with quadratic function than by the algorithms with Gaussian or Lorentzian function. The same fact can be observed in Fig. 7 where the norm $\|q(t)-q_d\|$ is depicted. The algorithm with quadratic function reaches the destination in time below 1 s, while the other two algorithms need about 1.5 s. In comparison, the "classic" algorithm uses the whole time horizon, by default.

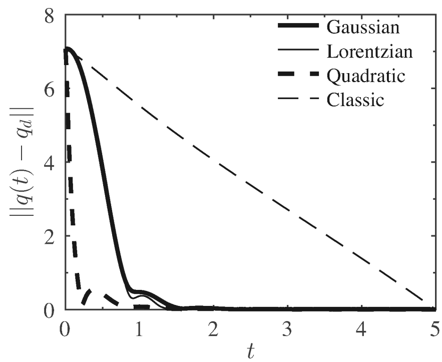

Figure 7: Norm $\|q(t)-q_d\|$ in time

6. Conclusion

The paper introduces the new algorithm which is able to plan the motion in a shorter time than the predefined time horizon, which is an update to classic motion planning algorithms derived by means of the endogenous configuration space approach.

The presented algorithms could be tuned by a specific selection of the $H\big(y(t)\big)$ function. The efficiency of the algorithm has been presented with numerical simulation, where the three different functions have been enrolled. The results for the Gaussian function (16) and for the Lorentzian function (17) are comparable. All algorithms successfully solve the motion planning problem and reach the destination earlier than the time interval.

The modification introduced to the motion planning algorithm derived within an endogenous configuration space approach brings the new algorithm closer to the definition in [8, Formulation 14.1]. In the new algorithm the time horizon is no longer important, and the system reaches the termination action, wherein the velocities are equal to zero.

A. Derivation of the Jacobian inverse

Let us start from the Jacobian equation

$$\mathbb{J}_{q_0,T}\big(u(\cdot)\big)v(\cdot) = \int_0^T \frac{\partial H\big(y(t)\big)}{\partial y} J_{q_0,t}\big(u(\cdot)\big)v(\cdot)\,\mathrm{d}t = \eta, \tag{22}$$

where $v(\cdot) \in \mathscr{U}$ and $\eta \in \mathbf{R}^r$. To find a Moore–Penrose inversion $\mathbb{J}_{q_0,T} : \mathscr{U} \to \mathbf{R}^r$ we will use the Lagrange multiplier method, together with the minimization of the squared norm of the control function

$$\min_{u(\cdot)} \frac{1}{2} \int_0^T v^{\mathrm{T}}(t)v(t)\,\mathrm{d}t. \tag{23}$$

The Lagrange function takes the following form

$$\begin{aligned}\mathscr{L}(v(\cdot),\lambda) &= \frac{1}{2}\int_0^T v^{\mathrm{T}}(t)v(t)\,\mathrm{d}t \\ &\quad + \lambda^{\mathrm{T}}\left(\int_0^T \frac{\partial H\big(y(t)\big)}{\partial y} J_{q_0,t}\big(u(\cdot)\big)v(\cdot)\,\mathrm{d}t - \eta\right) \\ &= \frac{1}{2}\int_0^T v^{\mathrm{T}}(t)v(t)\,\mathrm{d}t + \lambda^{\mathrm{T}}\left(\int_0^T \frac{\partial H\big(y(t)\big)}{\partial q}\int_0^t \Phi(t,s)B(s)v(s)\,\mathrm{d}s\,\mathrm{d}t - \eta\right).\end{aligned} \tag{24}$$

The differentiation of the Lagrangian (24) and equating to zero, together with equality

$$\int_0^T\int_0^t f(t,s)\,\mathrm{d}s\,\mathrm{d}t = \int_0^T\int_s^T f(t,s)\,\mathrm{d}t\,\mathrm{d}s \tag{25}$$

give us

$$v(t) = -B^{\mathrm{T}}(t)\int_t^T \Phi^{\mathrm{T}}(s,t)\left(\frac{\partial H\big(y(s)\big)}{\partial q}\right)^{\mathrm{T}}\mathrm{d}t\,\lambda. \tag{26}$$

The substitution of (26) into (22) allows us to find the Lagrange multiplier

$$\lambda = -\left(\int_0^T \frac{\partial H\big(y(t)\big)}{\partial q} \int_0^t \Phi(t,s)B(s)B^{\mathrm{T}}(s) \int_s^T \Phi^{\mathrm{T}}(z,s)\left(\frac{\partial H\big(y(t)\big)}{\partial q}\right)^{\mathrm{T}} \mathrm{d}z\,\mathrm{d}s\,\mathrm{d}t\right)^{-1} \eta = -P^{-1}(T)\eta. \qquad (27)$$

Substituting the Lagrange multiplier (27) into (26) we solve the Jacobian equation (22) and obtain the pseudo inverse

$$v(t) = B^{\mathrm{T}}(t)\int_t^T \Phi^{\mathrm{T}}(s,t)\left(\frac{\partial H\big(y(s)\big)}{\partial q}\right)^{\mathrm{T}} \mathrm{d}t\, P^{-1}(T)\eta = \left(\mathbb{J}^{\#}_{q_0,T}\big(u(\cdot)\big)\eta\right)(t). \qquad (28)$$

Let us focus on the $P(T)$ function

$$P(T) = \int_0^T \frac{\partial H\big(y(t)\big)}{\partial q} \int_0^t \Phi(t,s)B(s)B^{\mathrm{T}}(s) \int_s^T \Phi^{\mathrm{T}}(z,s)\left(\frac{\partial H\big(y(t)\big)}{\partial q}\right)^{\mathrm{T}} \mathrm{d}z\,\mathrm{d}s\,\mathrm{d}t. \qquad (29)$$

If we make a following substitution

$$M(t) = \int_t^T \frac{\partial H\big(y(s)\big)}{\partial q}\Phi(s,t)\,\mathrm{d}s \qquad (30)$$

and use again the identity (25), then the equation (29) takes the form

$$P(T) = \int_0^T M(t)B(t)B^{\mathrm{T}}(t)M^{\mathrm{T}}(t)\,\mathrm{d}t. \qquad (31)$$

The trajectory of $M(t)$ from (30) can be computed from (13) with suitable final condition and the solution of (31) coincides with the solution of (14) with appropriate initial condition. Including (30) and (31) into (28) we obtain

$$\left(\mathbb{J}^{\#}_{q_0,T}\big(u(\cdot)\big)\eta\right)(t) = B^{\mathrm{T}}(t)M^{\mathrm{T}}(t)P^{-1}(T)\eta, \qquad (32)$$

what is just (12).

References

[1] F. ALOUGES, Y. CHITOUR and R. LONG: A motion-planning algorithm for the rolling-body problem, *IEEE ransactions on Robotics*, **26**(5), (2010), 827–836.

[2] S.C. AMISS and M. GUAY: Motion planning by the homotopy continuation method for control-affine systems, In *American Control Confere (ACC)*, pp. 1767–1772, Montreal, June 2012.

[3] N. BOIZOT and J. GAUTHIER: Motion planning for kinematic systems, *IEEE Transactions on Automatic Control*, **58**(6), (2013), 1430–1442.

[4] Y. CHITOUR: A continuation method for motion-planning problems. *ESAIM: COCV*, **12**, (2006) 139–168.

[5] Y. CHITOUR and H.J. SUSSMANN: Line-integral estimates and motion planning using the continuation method, In J. Baillieul, S.S. Sastry, and H.J. Susmann, Eds., *Essays on Mathematical Robotics*, pages 91–125. Springer-Verlag, New York, 1998.

[6] A.J. HÄUSLER, A. SACCON, A.P. AGUIAR, J. HAUSER and A.M. PASCOAL: Energy-optimal motion planning for multiple robotic vehicles with collision avoidance, *IEEE Transactions on Control Systems Technology*, **24**(3), (2016), 867–883.

[7] J. KARPIŃSKA and K. TCHOŃ: Continuation method approach to trajectory planning in robotic systems, In *16th International Conference on Methods Models in Automation Robotics (MMAR)*, pages 51–56, Aug 2011.

[8] S.M. LAVALLE: Planning Algorithms, Cambridge University Press, Cambridge, 2006. Available at http://planning.cs.uiuc.edu/.

[9] G. PAJAK and I. PAJAK: Planning of a point to point collision-free trajectory for mobile manipulators, In *10th Int. Workshop on Robot Motion and Control*, pages 142–147, Poznan, 2015.

[10] A. RATAJCZAK: Planowanie ruchu układów nieholonomicznych z wcześniejszym osiąganiem celu, In K. Tchoń and C. Zieliński, Eds., *Postępy Robotyki*, **2**, pages 585–594, Oficyna Wydawnicza Politechniki Warszawskiej, 2016.

[11] A. RATAJCZAK: Egalitarian vs. prioritarian approach in multiple task motion planning for nonholonomic systems, *Nonlinear Dynamics*, **88**(3), (2017), 1733–1747.

[12] A. RATAJCZAK and K. TCHOŃ: Parametric and non-parametric Jacobian motion planning for non-holonomic robotic systems, *J. Intell. Robot. Syst.*, **77**(3), (2015), 445–456.

[13] J. RATAJCZAK and K. TCHOŃ: Dynamically consistent Jacobian inverse for mobile manipulators, *Int. Journal of Control*, **89**(6), (2016), 1159–1168.

[14] E.D. SONTAG: Mathematical control theory: Deterministic finite dimensional systems, Springer-Verlag, New York, 2 edition, 1998.

[15] K. TCHOŃ: Endogenous configuration space approach: An intersection of robotics and control theory, In N. van de Wouw, E. Lefeber, and I. Lopez Arteaga, [Eds.], *Nonlinear Systems – Techniques for Dynamical Analysis and Control*, volume 470 of *Lecture Notes in Control and Information Sciences*, pages 209–234. Springer, 2017.

[16] K. TCHOŃ and J. JAKUBIAK: Endogenous configuration space approach to mobile manipulators: a derivation and performance assessment of Jacobian inverse kinematics algorithms, *Int. Journal of Control*, **76**(14), (2003), 1387–1419.

[17] K. TCHOŃ, A. RATAJCZAK and I. GÓRAL: Lagrangian Jacobian inverse for nonholonomic robotic systems. *Nonlinear Dynamics*, **82**(4), (2015), 1923–1932.

[18] K. ZADARNOWSKA: Switched modeling and task–priority motion planning of wheeled mobile robots subject to slipping, *J. Intell. Robot. Syst.*, **85**(3), (2017), 446–469.