Automatic reparking of the robot trailer along suboptimal paths

Andrei Ardentov
Ailamazyan Program Systems Institute
Russian Academy of Sciences
Pereslavl-Zalessky, Russia
aaa@pereslavl.ru

Kirill Yefremov
Department of Mechatronic Systems
Kalashnikov Izhevsk State Technical University
Izhevsk, Russia
efremofk@yandex.ru

***Abstract*—The work investigates the experimental problem of reparking trailer for a wheeled robot with a trailer. The geometric model with kinematic constraints leads to the sub-Riemannian problem. We solve this problem via nilpotent approximation. The corresponding solution is close to optimal and locally minimizes the total kinetic energy of the driving wheels. A full-scale model is designed in a way to avoid phase constraints usually appearing in trailer systems. We perform 64 different experiments with reparking trailer and obtain the satisfactory accuracy: for one maneuver we repark the trailer with maximum angle error equal to 4 degrees.**

***Index Terms*—Mobile robot, trailer, motion planning, Vicon, sub-Riemannian problem, nilpotent approximation.**

I. Introduction

Robots with an arbitrary number of connected trailers (n-trailers) consist of a tractor robot equipped with a feedback control device that acts as a driving link and passively connected trailers. Trailers are interconnected to the robot by a passive rotary joint, which can be attached to the robot in different ways: at the center of the driving wheel pair (on-axis) [1], [2], at some distance from it (off-axis or off-hooked) [3]–[6], or in combination [7]. The number of trailers is scaled depending on the capabilities of the control system. The robots with a hinged trailer attached at the center of the tractor wheels pair are most easily controlled.

In this work, we consider a mobile robot with a trailer with a rotary joint attached at the center of the driving wheel pair at the highest point. The connection at the highest point allows rotating the robot relative to the trailer for a full turn, without restrictions.

The parking problem for mobile robots without trailers have two variants of solutions:

1) following a given trajectory [8],
2) moving from one point to another.

For mobile robots with a trailer, following the specified trajectories gives a more significant solution than moving from one point to another, since the mobile robot pulls or pushes the trailer, which can not turn around without additional maneuvers. A significant problem for the movement of mobile robots with a trailer is the *jackknife effect* [1]. During folding, the mobile robot and the trailer can be mechanically damaged. Jackknife effects can appear during a rearward movement, due to the high non-linearity of the process, or during a rapid braking, due to the resulting inertia forces of the trailer.

We consider the problem of tracking the full-scale model of the robot with the trailer along suboptimal trajectories. The accuracy of the reparking algorithms is improved by refining the geometric parameters of the robot and more accurate discretizing of the actual controls (angular velocities of the driving wheels).

The work of A. Ardentov and K. Yefremov is supported by the Russian Science Foundation under grant 17-11-01387-P and performed in Ailamazyan Program Systems Institute of Russian Academy of Sciences.

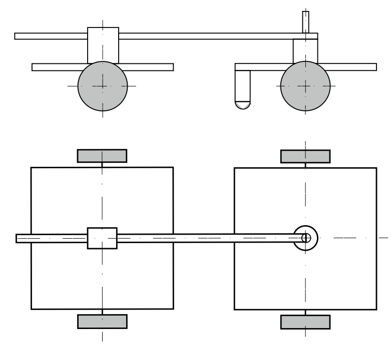

Fig. 1. Scheme of the robot with one trailer: side and top views.

II. The geometric model

We consider the following ideal model for a robot with a trailer moving on a horizontal plane. The mobile wheeled robot consists of a square platform with two driving wheels and one omnidirectional ball castor. We denote the center of the driving wheel pair by $(x, y) \in \mathbb{R}^2$. The robot orientation (or equivalently the driving wheel orientation) is given by an angle $\theta \in S^1$. The trailer consists of another square platform with two passive wheels and it is attached to the robot center (x, y) by a rod, see the geometric scheme in Fig. 1. The distance between the robot and the trailer is denoted by a constant value l_t. The trailer orientation is given by an angle $\varphi \in S^1$. Therefore, a position of the robot and the trailer is defined by the point

$$q = (x, y, \theta, \varphi) \in \mathbb{R}^2_{x,y} \times S^1_\theta \times S^1_\varphi.$$

III. The sub-Riemannian problem and nilpotentization

Kinematics of the described geometric model is described by the nonholonomic system

$$\begin{cases} \dot{x} = u_1 \cos\theta, \\ \dot{y} = u_1 \sin\theta, \\ \dot{\theta} = u_2, \\ \dot{\varphi} = -u_1 \frac{\sin\varphi}{l_t} - u_2, \end{cases} \tag{1}$$

where the controls $u_1 \in \mathbb{R}$, $u_2 \in \mathbb{R}$ are linear and angular velocities. We assume that there are no phase constraints, i.e., every configuration q is admissible.

The parking problem is defined by system (1) and by the initial and final configurations

$$q(0) = q_0 = (x_0, y_0, \theta_0, \varphi_0), \tag{2}$$

$$q(T) = q_1 = (x_1, y_1, \theta_1, \varphi_1). \tag{3}$$

The optimal control problem corresponds to the sub-Riemannian distance minimization

$$l = \int_0^T \sqrt{u_1^2 + \mu^2 u_2^2}\, dt \to \min, \tag{4}$$

where the constant $\mu > 0$ sets a compromise between linear and angular movement.

The optimal synthesis for problem (1)–(4) is unknown. We use the method of nilpotent approximation [12] in order to obtain a sub-optimal (i.e., close to optimal) solution of problem (1)–(4). System (1) is approximated by the control system for the nilpotent sub-Riemannian problem on the Engel group [14]. The transformation formula from boundary conditions (2) (3) to the boundary conditions of the nilpotent problem was obtained in [15] and then was clarified to the canonical form in [13]. The iterative algorithm developed in [13] constructs an approximate solution to problem (1)–(4). For far points q_0, q_1 this algorithm requires too many iterations and, consequently, high computing power is necessary to obtain such solutions.

We consider the special case when initial position of the robot coincides with the final one, i.e.,

$$q(0) = q_0 = (x_0, y_0, \theta_0, \varphi_0), \tag{5}$$

$$q(T) = q_1 = (x_0, y_0, \theta_0, \varphi_1), \tag{6}$$

This case admits a one-parameter family of approximate solutions [13]. This property helps to find a more precise solution. Moreover, changing the value of the parameter μ brings more variations of approximate solutions. Numerical calculations show that it is possible to find an approximate solution with one maneuver starting from (5) and arriving to a small enough neighborhood of the terminal point, i.e., $q(T) \in O_\varepsilon(q_1)$.

We apply the specialized algorithm for reparking trailer [13] to the full-scale model and obtain the precision of the algorithm, i.e., describe how big $O_\epsilon(q_1)$ can be for arbitrary positions of the trailer φ_0, φ_1.

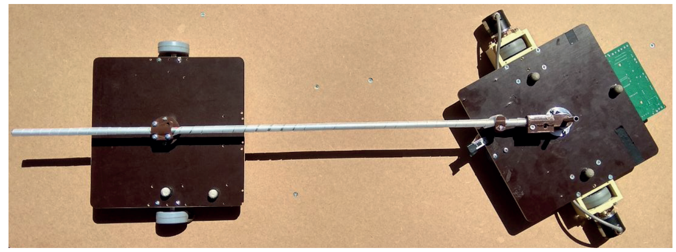

Fig. 2. Prototype of the mobile robot with the trailer.

IV. The full-scale model

According to the given geometric model (see Fig. 1), a full-scale prototype of the wheeled robot with the trailer was constructed for experimental research, see Fig. 2. Each of the driving robot wheels is equipped with individual stepper motors and incremental optical encoders. The feedback of the control system is provided by angular velocity sensors at the driving wheels. Stepper motors are connected directly to the robot wheels and have a resolution of 200 steps per revolution. A stepper motor controller (driver) with a 10-part division increases the sampling of the angular rotation of the wheel up to 2000 pulses per revolution. The optical incremental encoders have a resolution of 2000 pulses per revolution and are connected directly to the robot driving wheels. Such a design of the robot decreases the maximum speed of the robot, but it avoids the influence of inaccuracies appearing in the gearboxes manufacture. The full-scale model of the robot is 0.1m high and $0.24 \times 0.24\text{m}^2$ wide and long. The wheel radii are approximately equal to $r^0 = 0.025$m. The distance between the driving wheels is approximately equal to $l^0 = 0.278$m. A stepper motor controller is Gecodrive G210. A control board is manufactured as the control system for the robot, it is based on the STM32F7 microcontroller operating at a frequency of 180MHz. The frequency of the control system is 25Hz. The HC-05 module with a wireless Bluetooth interface is used to transmit the input data to the robot and to receive the output data from it.

The input data is given by discrete values of the angular velocities of the driving wheels, which are associated with the controls u_1, u_2 in the following way:

$$w_l = \frac{u_1 - \frac{l}{2} u_2}{r_l}, \qquad w_r = \frac{u_1 + \frac{l}{2} u_2}{r_r}, \tag{7}$$

where r_l, r_r are the actual radii of the left and right wheel; l is the actual distance between the centers of the driving wheels.

The trailer model replicates the robot model, but the wheel axle of the trailer does not have motors and feedback sensors. The connection between the robot and the trailer has one degree of freedom — rotation around the center of the robot (the connection point). The connection point is located at the top point of the robot and the angle of rotation is not limited. Thus, there are no phase constraints, so we avoid the so-called *jackknife effect* when the robot and the trailer collide.

We use the motion capture system called Vicon to track the movement of the robot with the trailer. To do this, reflective markers are placed at certain points of the robot and the trailer, see details in [9], [10]. The Vicon motion capture system recognizes the visible markers during experiments. The coordinates of the markers are transmitted to the Matlab program from the motion capture system. The Matlab program exports the corresponding array with a sampling rate of 100 Hz to a file. The file is further processed by the Mathematica program which calculates the real trajectory and compares it with the ideal one.

In order to accurately control the robot movement, we clarify the precise values of the driving wheel radii $r_l = r^0 + a_l, r_r = r^0 + a_r$ and wheel distance $l = al^0$, here a_l, a_r, a are correction parameters satisfying

$$(-\varepsilon \leq a_r \leq \varepsilon) \& (-\varepsilon \leq a_l \leq \varepsilon) \& (1 - \hat{\varepsilon} \leq a \leq 1 + \hat{\varepsilon}),$$

where the constants $\varepsilon = 0.005$m and $\hat{\varepsilon} = 0.05$ express a possible error in the initial measurement of the values r^0, l^0.

We use trajectories of a constant curvature to verify accuracy of the obtained model. The ideal linear and angular velocities are $u_{10}^i \equiv 1$, $u_{20}^i \equiv \mathrm{const}$ (straight line segments with $u_{20}^i \equiv 0$, arcs of circles with $u_{20}^i \neq 0$), where i is a number of the trajectory.

Ideal values of the angular velocities for the driving wheels w_l^i, w_r^i are calculated by the following formulas:

$$w_l^i = \frac{u_{10}^i - \frac{l^0}{2} u_{20}^i}{r^0}, \qquad w_r^i = \frac{u_{10}^i + \frac{l^0}{2} u_{20}^i}{r^0}.$$

The real linear and angular velocities are expressed by the values

$$u_{1r}^i = u_{10}^i + \frac{a_r w_r^i + a_l w_l^i}{2}, \qquad u_{2r}^i = \frac{u_{20}^0 + \frac{a_r w_r^i - a_l w_l^i}{l^0}}{a}.$$

Note that the real velocities of the driving wheels are time-reparametrized in such a way that the robot at the beginning of motion accelerates from the zero velocity to the maximum $v_m = 0.12$m/s and decelerates to zero at the end. Since such reparametrization does not affect the calibration process, we use constant velocities for simplicity.

We refine the formula for the method of least squares used in [11]:

$$\begin{aligned} \sum_{i=1}^{n} \Big(& \left(L_r^i - u_{1r}^i L_0^i\right)^2 + \left(\mathrm{mod}_{[-\pi,\pi]}(\theta_1^i - \theta_0^i - u_{2r}^i L_0^i)\right)^2 \\ & + \left((x_1^i - x_0^i) u_2^r - u_{1r}^i (\sin\theta_1^i - \sin\theta_0^i)\right)^2 \qquad (8) \\ & + \left((y_1^i - y_0^i) u_{2r}^i + u_{1r}^i (\cos\theta_1^i - \cos\theta_0^i)\right)^2 \Big) \to \min, \end{aligned}$$

where L_0^i and L_r^i are the ideal and actual distances traveled by the robot center; $(x_0^i, y_0^i, \theta_0^i, \varphi_0^i)$, $(x_1^i, y_1^i, \theta_1^i, \varphi_1^i)$ are the real initial and final configurations of the mobile robot with the trailer for i-th trajectory.

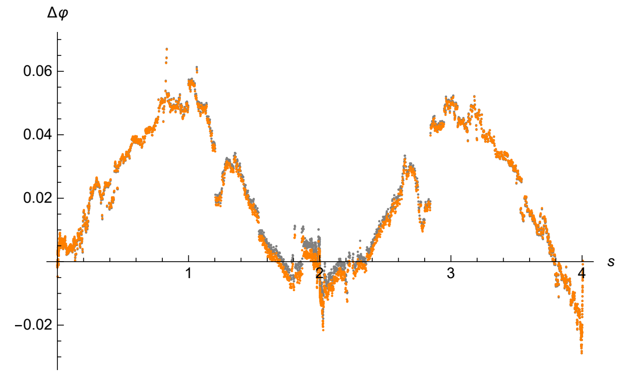

Fig. 3. The difference $\Delta\varphi$ between the real and ideal trailer angle, when the robot is moving along the full circle forward and backward with the refined geometric parameters

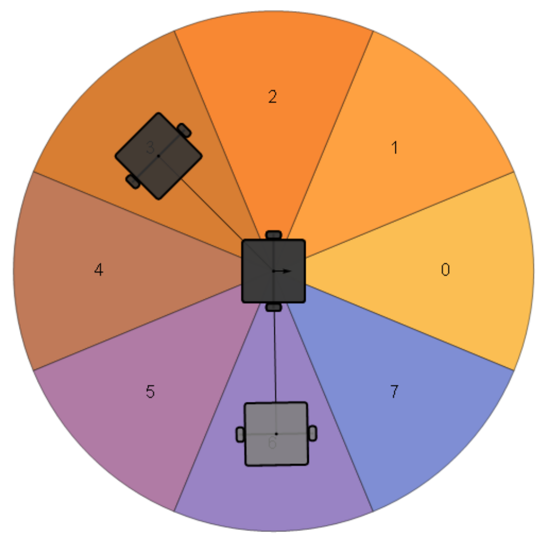

Fig. 4. The reparking problem from octant 6 to octant 3.

The Mathematica software is used for a numerical minimization of (8) by the standard methods (NMinimize, FindMinimum). Calibrated values are

$$r_l = 0.0243288, \quad r_r = 0.0245978, \quad l = 0.2773358. \quad (9)$$

Obtained values (9) are verified by similar experiments when the robot is moving along straight lines and circles. We extend these trajectories with the inverse movements to the initial point q_0. Fig. 3 shows the correspondence of the full-scale model with the geometric one. In this example, the maximum deviation of the trailer angle along the way is 0.067, the deviation at the endpoint $\Delta q_1 = (\Delta x_1, \Delta y_1, \Delta\theta_1, \Delta\varphi_1) = (0.00188\text{m}, 0.0006258\text{m}, 0.00486, 0.0002628)$ confirms satisfactory control accuracy which allows us to perform experiments with more complex maneuvers.

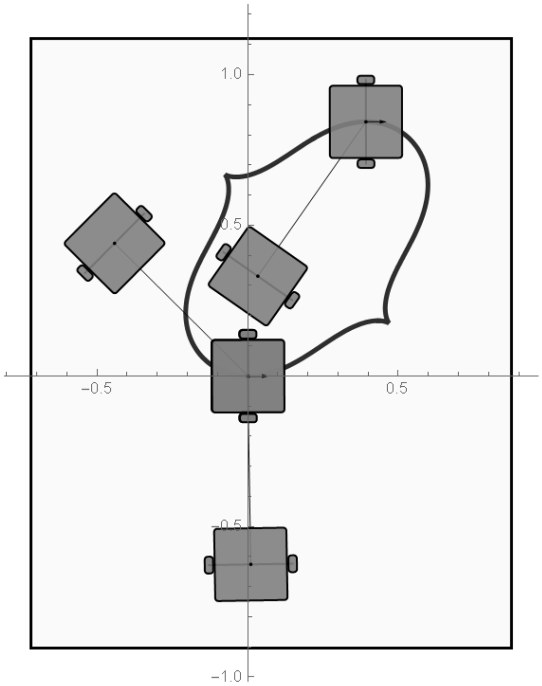

Fig. 5. A solution for the reparking problem from octant 6 to octant 3.

V. Experiments

A series of experiments were carried out on a flat horizontal polygon with the size 1.5×3 m^2. The main parameters of the experiments are $\varphi_0, \varphi_1 \in S^1$. The circle for possible trailer angles is divided into 8 octants, see Fig. 4. We consider all possible variants for reparking the trailer from octant i to octant j, where $i, j = 0, \dots, 7$. The final trailer position is fixed $\varphi_1 = \pi(j/4 - 1)$ for octant j. The initial position is an arbitrary value satisfying $\varphi_0 \in [\pi(2i-9)/8, \pi(2i-7)/8]$ for octant i. We manually set a random initial position, determine its value by the Vicon system and write it in a file.

Mathematica software reads this file and calculates an approximate solution for the corresponding reparking problem, see Fig. 5. Since each solution has two cusp points, we decompose the trajectory into 3 pieces, naturally parameterize each piece and apply reparametrization with acceleration to the maximum speed.

The control functions for the angular velocities w_l, w_r of the driving wheels are discretized by the integration into stepper motor steps. The discrete values are switching in order to minimize the integral difference between the continuous and discretized velocities, see an example on Fig. 6. The Mathematica software constructs the ideal trajectory and the ideal trajectory corresponding to the discrete velocities. For each test, these trajectories are not distinguishable by eye.

Further, the discrete velocity values are transmitted to the robot in order to solve the reparking problem. The Vicon system monitors the experiment and saves the data in a file. The comparison of the ideal trajectory with the experimental one is illustrated on Fig. 7.

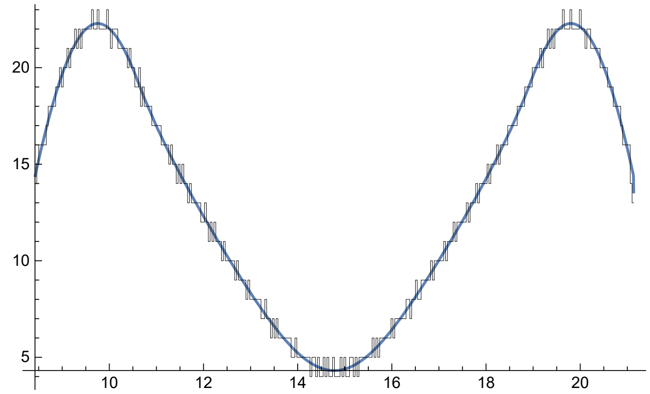

Fig. 6. Discretization of drive wheel speeds

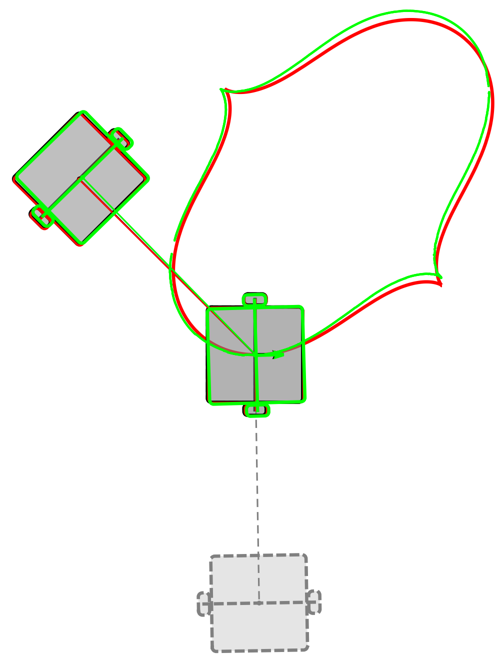

Fig. 7. Comparison of the ideal trajectory (red) with the real one (green) from the experiment with the dashed gray initial position.

VI. Conclusion

The algorithm of nilpotent approximation [13] for reparking robot with a trailer is adapted for the full-scale model. We improve the method for refining geometric parameter values for the prototype of the mobile robot with the trailer. We consider 64 various boundary conditions for the reparking problem given by values φ_0, φ_1. For each boundary condition, we calculate discrete sub-optimal controls to solve the corresponding problem for the full-scale model. Further, we perform the corresponding 64 experiments to verify our approach and estimate how precisely we solve the reparking problem. We obtain the average and maximum angle errors equal to 1.8 and 4 degrees correspondingly.

REFERENCES

[1] M. Yue, X. Hou, M. Fan, R. Jia, “Coordinated trajectory tracking control for an underactuated tractor-trailer vehicle via MPC and SMC approaches,” In 2017 2nd IEEE International Conference on Advanced Robotics and Mechatronics (ICARM), 2017, pp. 82–87.

[2] D. H. Kim, J. H. Oh, “Experiments of backward tracking control for trailer system,” In Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C), vol. 1, 1999, pp. 19–22.

[3] M. Beglini, L. Lanari, G. Oriolo, “Anti-Jackknifing Control of Tractor-Trailer Vehicles via Intrinsically Stable MPC,” In 2020 IEEE International Conference on Robotics and Automation (ICRA), 2020, pp. 8806–8812.

[4] J. Chiu, A. Goswami, “Driver assist for backing-up a vehicle with a long-wheelbase dual-axle trailer,” In International Symposium on Advanced Vehicle Control (AVEC). Seoul, Korea: KSAE, 2012.

[5] K. Yoo and W. Chung, “Pushing motion control of n passive off-hooked trailers by a car-like mobile robot,” 2010 IEEE International Conference on Robotics and Automation, 2010, pp. 4928–4933.

[6] Bozek P., Karavaev Y. L., Ardentov A. A., Yefremov K. S., “Neural network control of a wheeled mobile robot based on optimal trajectories,” International Journal of Advanced Robotic Systems, 2020, pp. 1-10

[7] N. Evestedt, O. Ljungqvist, D. Axehill, “Path tracking and stabilization for a reversing general 2-trailer configuration using a cascaded control approach,” In 2016 IEEE Intelligent Vehicles Symposium (IV), 2016, pp. 1156–1161.

[8] L. Tang, S. Dian, G. Gu, K. Zhou, S. Wang, & X. Feng, “A novel potential field method for obstacle avoidance and path planning of mobile robot,” In 2010 3rd IEEE International Conference on Computer Science and Information Technology, vol. 9, 2010, pp. 633–637.

[9] Y. L. Karavaev, A. A. Kilin, “Nonholonomic Dynamics and Control of a Spherical Robot with an Internal Omniwheel Platform: Theory and Experiments,” Proceedings of the Steklov Institute of Mathematics, vol. 295, 2016, pp. 158–167.

[10] A. A. Kilin, Y. L. Karavaev, “Experimental research of dynamic of spherical robot of combined type,” Russian Journal of Nonlinear Dynamics, vol. 11, no. 4, 2015, pp. 721–734.

[11] A. A. Ardentov, Y. L. Karavaev, K. S. Yefremov, “Euler elasticas for optimal control of the motion of mobile wheeled robots: the problem of experimental realization,” Regular and Chaotic Dynamics, vol. 24, no. 3, 2019, pp. 312–328.

[12] A. Bellaiche, J.-P. Laumond, J. J. Riser, “Nilpotent infinetisimal approximations to a control Lie algebra,” IFAC NCSDS, Bordeaux, 1992, pp. 174–181.

[13] A. A. Ardentov, A. P. Mashtakov, “Control of a mobile robot with a trailer based on nilpotent approximation,” Autom. Remote Control, vol. 82, no. 1, 2021, pp. 73–92.

[14] A. A. Ardentov, Yu. L. Sachkov, “Cut time in sub-riemannian problem on engel group,” ESAIM: COCV, vol. 21, no. 4, 2015, pp. 958–988.

[15] A. A. Ardentov, “Controlling of a mobile robot with a trailer and its nilpotent approximation,” Regul. Chaot. Dyn., vol. 21, 2016, pp. 775–791.