Computationally Efficient Path Planning for Wheeled Mobile Robots in Obstacle Dense Environments

Husnu Turker Sahin and Erkan Zergeroglu

Gebze Institute of Technology {htsahin, ezerger}@bilmuh.gyte.edu.tr

Summary. This paper considers the problem of nonholonomic motion planning for wheeled mobile robots with limited sensing capabilities. A simple yet efficient path planning algorithm is developed, which combines a virtual front steering mechanism with an easy to implement obstacle avoidance method generating smooth motion profiles. Extensions are presented for improved navigation in environments with u-shaped concave barriers or tunnels, which are considered as difficult conditions for most planners in the literature. Our approach is computationally efficient, easy to implement and is suitable for robots with entry level equipment. Feasibility and effectiveness of the proposed method is illustrated by numerous simulations.

1 Introduction

Smooth path generation for nonholonomic wheeled mobile robots (WMRs) is being researched significantly in the recent years. The nonholonomic constraints of WMRs impose difficulties for effective path planning, on top of this presence of static and/or dynamic obstacles in operation environments complicates the problem further. Alternative solutions have been proposed on WMR trajectory planning ranging from graph search methods [1, 2], to the application of potential function based techniques [4, 3, 5, 6]. Many of these methods assume full observability of the operational space [1, 3, 5] and some cannot provide dynamical path planning [3, 5], which is impractical for general applications of WMRs. Recently more advanced methods have been presented that offer better solutions to the path planning problem for obstacle cluttered environments [7, 8]. However, these methods utilize triangulation based mappings of the nearby environment by expensive sensory devices. This increases the computational cost of the planning algorithms, and necessitates more expensive electronics like laser scanners. A qualitative revision on the relation between the sensing abilities of wheeled mobile robots and the topology of their environment can be found in [9, 10].

In this paper we present a computationally efficient approach to the nonholonomic and smooth path planning problem for WMRs based on limited

sensing information. The proposed path planner utilizes a two axle reference robot, the back-wheel of which forms a readily nonholonomic reference trajectory for unicycle robots to follow as the front axle is steered in the direction of the device target. A simple yet effective obstacle avoidance algorithm is used in parallel with this direct steer mechanism to achieve trajectory generation under minimum sensing conditions, ie. in the presence of only a small number of ON-OFF sensors providing a small sensing zone. When an obstacle is sensed, the planner assumes it as a circular body, and readjusts the drag force to the front axle to keep the path distant from the center of the obstacle estimate. The planner has the potential to avoid u-shaped concave blocks of arbitrary size as well as tunnels by varying two parameters of its algorithm. Super-ellipses with minimum number of parameters can be utilized as virtual obstacles to fill the concave sections of u-shaped obstacles, thereby eliminating the need for complicated mapping algorithms. With these properties the proposed method is suitable for implementation on small robots with entry level sensors, and has potential for extension to multi-robot trajectory planning.

The rest of the paper is organized in the following manner: In Section 2 the employed WMR kinematic model and the problem formulation are summarized, while the path planner details are given in Section 3. Simulation results are presented in Section 4, followed by Conclusions in Section 5.

2 Kinematic Model and Problem Formulation

The unicycle WMR kinematics utilized in this paper are modelled by the following relation [12]:

$$\begin{bmatrix} \dot{x_c} \\ \dot{y_c} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_l \\ \dot{\theta} \end{bmatrix}. \tag{1}$$

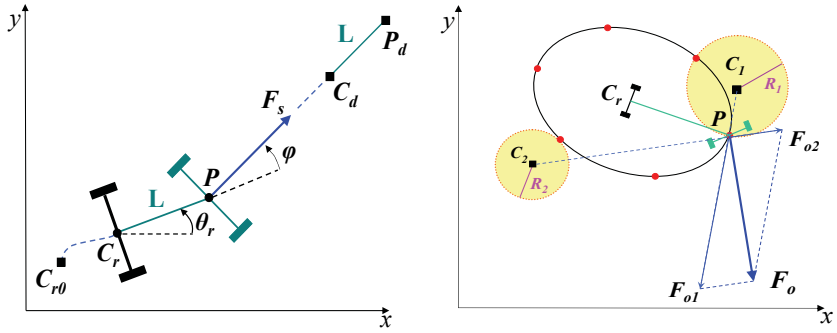
Here $q_c = \begin{bmatrix} x_c, & y_c, & \theta \end{bmatrix}^T \in \mathbb{R}^{3\times 1}$ is the pose vector of the WMR, with $[\, x_c(t),\, y_c(t)\,]^T \in \mathbb{R}^{2\times 1}$ and $\theta(t) \in \mathbb{R}^1$ denoting the robot center of mass (COM) and orientation, respectively. The $v(t) = [\, v_l,\, \dot{\theta}\,]^T$ term is the linear and angular velocity vector. The cartesian WMR velocities are obtained from (1) to be:

$$\dot{x}_c = v_l \cos\theta, \quad \dot{y}_c = v_l \sin\theta. \tag{2}$$

Planned paths must satisfy the following nonholonomicity constraint for accurate tracking by WMRs,

$$\dot{x}_c \cos\theta - \dot{y}_c \sin\theta = 0. \tag{3}$$

The main objective of our path planner is to generate a nonholonomic collision free path in an unstructured obstacle dense environment with limited sensor data. The utilized reference robot is the 2-axle device (Figure 1(a)), having the current front axle COM position $P = [\, x_e,\, y_e\,]^T \in \mathbb{R}^2$. If the robot is steered by a force F_s from point P to a desired front end location P_d, its back wheel should follow generating a readily nonholonomic trajectory between



(a) Front steer mechanism model. (b) Obstacle avoidance model.

Fig. 1. The path planner reference robot models.

the current WMR position $C_r = [\, x_r,\, y_r \,]^T$ and its target $C_d = [\, x_d,\, y_d \,]^T$. Avoidance of obstacles that may be encountered in the robot path is achieved by changing the steer force F_s in the direction away from these blocks.

3 The Proposed Path Planner

3.1 Nonholonomic Steering Towards a Desired Target

The general equations modeling the kinematics of the bicycle reference robot, depicted in Figure 1(a), are derived from the extension of the unicycle kinematics of (1), (2) via suitable steer force-velocity relations as:

$$\dot{x}_r = v_l \cos(\theta_r), \quad \dot{y}_r = v_l \sin(\theta_r), \quad \dot{\theta}_r = v_l \sin(\varphi)\,/\,L, \quad v_l = F\cos(\varphi). \tag{4}$$

The modified terms in (1) are the linear velocity which is the projection of the steer force F, on the robot direction via steer angle φ; and the angular velocity $\dot{\theta}_r$ related to L, the distance between the front axle and back axle positions P and C_r in figure 1(a).

The steering force F_s is applied from the front axle center P towards desired WMR COM position C_d as depicted in Figure 1(a). This force is not declined until point P_d is reached. Hence as the front axle reaches P_d, the back wheel center generates a smooth reference trajectory from the initial position C_{r_0} to the desired target C_d suitable for a unicycle WMR to track. To keep the steer force as simple as possible, while satisfying these properties, the following formula is selected:

$$F_s = K \frac{e}{\sqrt{\|e\|^2 + \varepsilon}} + B\dot{e}. \tag{5}$$

Here $K, B \in \mathbb{R}^{2\times 2}$ are diagonal positive definite scaling matrices, and $\|e\|$ denote the Euclidean norm of the error term e, defined as the difference between the desired front axle COM P_d and its current position P as follows:

$$e = P_d - P = [\; x_{e_d} - x_e,\; y_{e_d} - y_e \;]^T. \tag{6}$$

In equation (5), a steering force normalization is applied by the division of the position error by its norm. The error norm is inversely proportional to the distance between the terminal point and current front end location P, thereby counterbalancing any decline in the forcing function arising from reductions in system error. Also the $\varepsilon \in \mathbb{R}^+$ term in (5) is to prevent possible instability after the desired target P_d is reached.

3.2 Obstacle Detection and Avoidance

During most robotic missions, WMRs have access to information from a limited region of the task space via their sensors. As unicycle WMRs are of circular structure, the sensing zone is assumed to be of ellipsoidal shape in our work. When an obstacle is detected by one of the sensors, the path planner switches to the obstacle avoidance phase. Our obstacle avoidance algorithm operates effectively with commonly accessible hardware such as ON/OFF sensors. The properties of such a setup are depicted in Figure 1(b), where the red dots show the sensors. Depending on which sensors send the signals, the WMR assumes a "circular blocking object" is encountered about that location. The planner then calculates estimates for the centers of the obstacles $C_i(t)$; and hence for their radii $R_i(t) \in \mathbb{R}^+$ as a measure of the size of the encountered block. In these calculations the number of neighboring ON sensors and the duration of their signals are utilized. Two such obstacle estimates are depicted by the shaded circles in Figure 1(b). After the obstacle estimates are computed, a force component $F_{o_i} = [F_{ox_i},\ F_{oy_i}]^T,\ \ i = 1, \ldots N$, pointing from obstacle center $C_i(t)$ to the reference robot front axle center $P(t)$ is formed for each excited neighboring sensor group. Then the vector sum of these components forms the overall obstacle avoidance force F_o as follows:

$$F_o(t) = \sum_{i=1}^{N} F_{o_i}(t) = \sum_{i=1}^{N} w_i\, R_i(t)\, [P(t) - C_i(t))], \quad i = 1, \ldots, N, \qquad (7)$$

where R_i, C_i, and P are the terms defined above, and $w_i \in \mathbb{R}^+$ denote additional weights selected higher for front sensors to emphasize avoidance in the robot forward path. The ratio of each force component F_o in (7), increases according to the impact time of the related obstacle. However, the overall avoidance force is kept constant by preserving the magnitude inherited from the steer towards the target mode force in (5) to ensure generation of bumpless reference velocities imperative for the nonholonomic controllers. If the WMR is encircled by some fixed or mobile obstacles, such that 5 or more of the device sensors are excited, the planner ceases the steer forces until some of the blocks are removed to avoid the high risk of collision.

Remark 1. The obstacle avoidance forces F_{o_i} should not be ceased immediately after the encountered obstacles are out of the sensing zone. Otherwise, the path planner may start to switch instantaneously between the forward steer and obstacle avoidance modes, resulting in undesirable chattering in the overall steer force $F(t)$. For this reason the cross-over from obstacle avoidance to the

steering toward the target mode is carried out in a 2 stage period. In the first stage, a small virtual margin is added to the estimated radius of the obstacles. Thus the planner to continues to execute the block avoidance mode for an additional time period $\triangle t$ after the actual obstacle is out of the sensing zone. The second stage is implemented via a first order spline like transition. The overall obstacle steer force function $F = [F_x,\, F_y]^T$ in this period is as follows:

$$F = F_s \frac{(t - t_s)}{\delta t} + F_o \frac{(t_s + \delta t - t)}{\delta t}, \tag{8}$$

where F_s and F_o are the front steering and normalized obstacle avoidance forces in (5) and (7). This cross-over phase is confined to $t \in [t_s,\, t_s + \delta t]$ interval, where t_s is the time instant the obstacle is out of sensing zone, and δt is the duration of the transition. This two stage period eliminates chattering in F in addition to local smoothness loss arising from steer angle $\varphi > 90^0$.

3.3 Extension for Large U-Blocks and Complicated Tunnels

The proposed obstacle avoidance algorithm achieves good performance for virtually all convex and many moderately concave obstacles. However, some extremely concave blocks such as larger u-shaped barriers may cause the WMRs to be entrapped. This is a common problem for most path planners containing a target weight in their algorithms [7]. The advantage of our method is local obstacle topology can be emphasized over target bias by increasing the values of the parameters δt and $\triangle t$ of the spline period. Moreover this modification also improves planner performance in complex tunnels. Thus if the algorithm detects such obstacles (by monitoring the distance function to the WMR destination, or continuous excitations from more than one sensor, respectively), the values of δt and $\triangle t$ are increased to higher values. After the obstacle avoidance mode is over, these parameters are reset to their default lower entries for increasing manoeuvrability and enhancing detection of short-cuts.

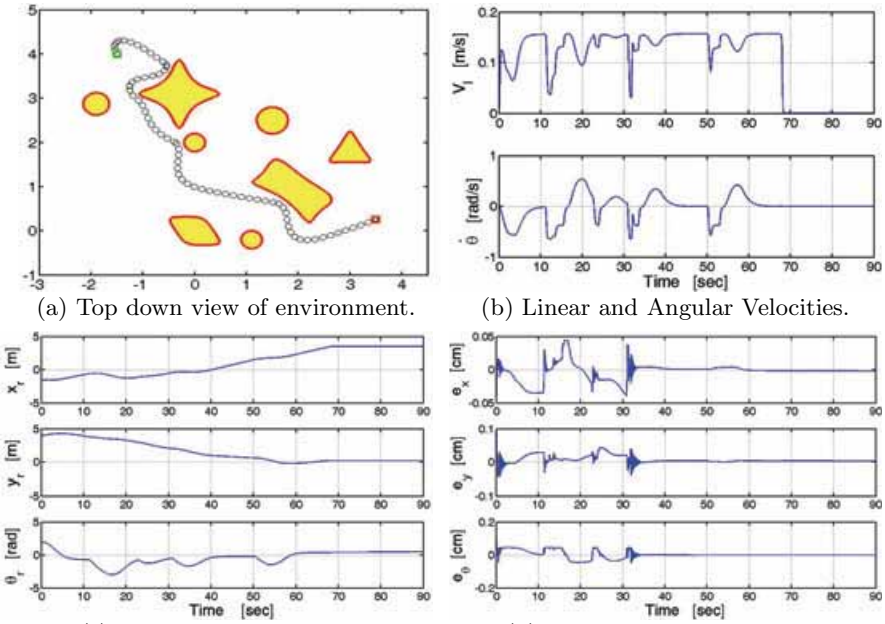
Remark 2. To ensure the WMR does not re-enter traps of u-blocks, we recommend the utilization of 4th degree super-ellipses as virtual obstacles to fill the concave sections. These ellipses are planar, square like curves with minimum number of parameters in the form:

$$\frac{(x - x_0)^4}{a^4} + \frac{(y - y_0)^4}{b^4} = 1.$$

On encountering u-blocks, the δt and $\triangle t$ parameters can be incremented until the generated path leaves the super-ellipse zone. Then the virtual obstacle is activated so that the parameters are reset to their initial values without any risk for the WMR to re-enter the concave trap any more.

4 Simulation Results

We have simulated the proposed path planning method in Matlab© / Simulink© environment using C mex S-functions. Simulations for path planning in various



(a) Top down view of environment. (b) Linear and Angular Velocities.

(c) The Generated Path. (d) The pose tracking errors.

Fig. 2. The first simulation environment and generated path related plots.

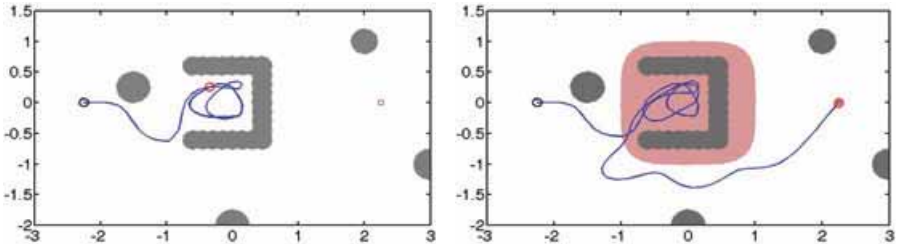
operational spaces with obstacles of different topologies have been performed. ON-OFF sensors located at the robot sensing zone perimeter are utilized in our simulations. We set the number of these sensors to 6 in parallel with the sensing equipments of small WMR configurations. Likewise the major and minor axis lengths of the resulting sensing region and the parameters of the steering force function in (5) are set to:

$$a = 22.5\,cm,\ b = 20\,cm,\ K = diag(0.165, 0.165),\ B = diag(0.01, 0.01). \quad (9)$$

The spline period parameters are assigned the values $\triangle t = 0.6$, $\delta t = 2$ [s], except where otherwise stated for improved performance. The unified non-holonomic controller in [11] is utilized in our simulations.

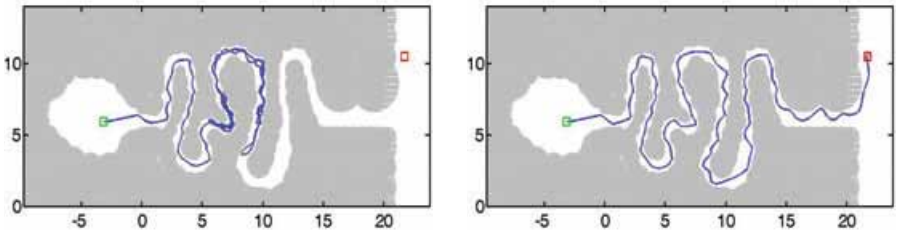
The first simulation is on navigation of a WMR in an environment with moderate obstacles as depicted in Figure 2(a) with the initial and desired WMR positions are marked by green and red rectangles respectively. Figure 2(b) depicts the WMR linear and angular velocities, which are functions with no chattering owing to the spline transition in Remark 1. Also the envelope of the velocity does not decline in parallel with the steer force selection in (5). Despite numerous obstacles the robot reaches its desired position as depicted in Figure 2(c). Finally from Figure 2(d) we observe the tracking errors to be negligible except small fluctuations during obstacle manoeuvres.

Next are the simulations verifying the positive influence of readjusting $\triangle t$ and δt parameters systematically for u-blocks and tunnels. The behavior of path planner versus concave blocks is shown in Figures 3(a) and 3(b). In



(a) Low and fixed $\triangle t = 0.6$, $\delta t = 2$s (b) Virtual Obstacle $\triangle t$, δt incremented

Fig. 3. Simulations on u-blocks.



(a) Short spline times of $\triangle t = 0.8$, $\delta t = 4$s. (b) Longer transition $\triangle t = 1$, $\delta t = 7.5$s.

Fig. 4. Route Planning in a demanding tunnel.

the simulation of Figure 3(a), the WMR encounters a u-barrier, where it is trapped. This is caused by the small, fixed default values of spline parameters $\triangle t = 0.6$ and $\delta t = 2$ [s]. The simulation is repeated in Figure 3(b). However, this time after the trap is sensed, $\triangle t$ and δt is incremented until the WMR leaves the virtual obstacle $x^4 + y^4 = 1$ covering the u-gap in parallel with Remark 2. After the WMR is outside the virtual obstacle, these parameters are reset to their default values. Thus we observe the WMR reach its final location with no further delay.

Final simulations are on path planning in a complex tunnel as depicted in Figure 4. The objective is to guide the WMR on the left of the tunnel to the clearance on the right. For the first simulation on Figure 4(a), the $\triangle t$ and δt parameters are set to smaller values of 0.6 and 2 [s] respectively, implying a quicker exit from the spline transmission period. The resulting target bias causes the robot to change its course many times in the tunnel and so that it cannot reach its target. In the next simulation in Figure 4(b) the parameters are set to higher values of 1 and 7.5 [s]. Thus the guidance of the planner is improved, and the WMR reaches its destination with no direction reversals. As a result we can conclude that our planner can also perform effectively for large concave blocks and complex tunnels.

5 Conclusions

We have presented a simple yet effective algorithm for collision free, smooth path generation for unicycle type WMRs. The proposed path planner inte-

grates a virtual front steering axle to the actual kinematic wheel enforcing a nonholonomic path for the (actual) rear axis, when the front axle is steered towards a desired location. To tackle the obstacle avoidance problem, obstacles detected by sensors are assumed as of circular geometry, and repulsion forces from their estimated centers are applied to the virtual front axle for an alternative collision free route. Extension for planning in the cases of obstacles of problematic topology such as u-shaped barriers and tunnels without the necessity for computationally expensive mappings is also proposed. Simulation results confirm the computational efficiency and performance of our planner as an alternative to the existing complex high performance planners.

References

1. P. Jacobs and J. Canny, "Planning Smooth Paths for Mobile Robots", *IEEE International Confernce on Robotics and Automation,* 1, 2-7, May 1989.
2. J-P. Laumond, P. E. Jacobs, M. Taix and R. M. Murray, "A Motion Planner for Nonholonomic Mobile Robots", *IEEE Transactions on Robotics and Automation,* 10(5), October 1994.
3. R. M. Murray and S. S. Sastry, "Nonholonomic Motion Planning: Steering Sinusoids", *IEEE Transactions on Automatic Control,* 38(5), 700-716, May 1993.
4. E. Rimon and D. E. Koditschek, "Exact Robot Navigation using Artificial Potential Function", *IEEE Transactions on Robotics and Automation,* 8(5), 501-518, 1992.
5. J. Chen, W. E. Dixon, D. M. Dawson and T. Galluzzo, "Navigation and control of a Wheeled Mobile Robot", *IEEE/ASME International Conference on Advanced Intelligent Mechatronics,* 1145-1150, 2005.
6. F. Lamiraux and D. Bonnafous, "Reactive Trajectory Deformation for Nonholonomic Systems: Applications to Mobile Robots", *IEEE International Confernce on Robotics and Automation,* 3099-3104, May 2002.
7. J. Minguez and L. Montano, "Nearness diagram (ND) navigation: collision avoidance in roublesome scenarios", *IEEE Transactions on Robotics and Automation,* 20(1), 45-59, Feb. 2004.
8. P. Krishnamurthy and F. Khorrami, "GODZILLA: A Low-resource Algorithm for Path Planning in Unknown Environments", *American Control Conference,* 110-115, 2005.
9. J.M. O'Kane and S.M. LaValle, "On Comparing the Power of Mobile Robots", *Robotics: Science and Systems,* August 2006.
10. B. Tovar, A. Yershova, J.M. O'Kane and S.M. LaValle, "Information Spaces for Mobile Robots", *International Workshop on Robot Motion and Control RoMoCo'05,* 11-20, June 2005.
11. W. E. Dixon, D. M. Dawson, E. Zergeroglu, and F. Zhang, "Robust Tracking and Regulation Control for Mobile Robots", *International Journal of Robust and Nonlinear Control,* 10, 199-216, 2000.
12. R. M'Closkey and R. Murray, "Exponential Stabilization of Driftless Nonlinear Control Systems Using Homogeneous Feedback", *IEEE Transactions on Automatic Control,* 42(5), 614-628, May 1997.