Analytical time-optimal trajectories for an omni-directional vehicle

Conference Paper *in* Proceedings - IEEE International Conference on Robotics and Automation · May 2012

DOI: 10.1109/ICRA.2012.6224602

2 authors:

Weifu Wang
University at Albany, The State University of New York

Devin J. Balkcom
Dartmouth College

Analtyical time-optimal trajectories for an omni-directional vehicle

Weifu Wang and Devin J. Balkcom

***Abstract*— We present the first analytical method for solving for the time-optimal trajectory between any pair of configurations of a kinematic model of an omni-directional vehicle with bounds on wheel speeds, in an obstacle-free plane. It has been previously shown that time-optimal trajectories for this system are sequences of controls that are constant in the frame of the robot. There are three different types of trajectories that may be optimal, known as roll, shuffle, and singular trajectories. Singular trajectories have previously been solved analytically; this paper solves roll and shuffle trajectories, and also tightens the bound on the number of switches between controls in roll trajectories to six. Using the derived algorithm, we sampled the configuration space densely in the region near the origin to determine how the time and structure of optimal trajectories changes across configurations. We also compared the time to a simple driving strategy: turn until the fastest translation direction faces the goal, drive to the goal, and turn to the correct angle. The result shows that the time-optimal trajectory tends to be ten to twenty percent faster.**

I. INTRODUCTION

This paper presents the first analytical method to find the time-optimal trajectory between any two configurations of a particular symmetric three-wheeled model of an omni-directional vehicle, without obstacles in the workspace. The model is shown in figure 1. The wheels are *omniwheels* – the wheels are powered in the direction that they are driven, but can slide freely in the perpendicular direction. We assume that the speeds of the wheels can be controlled, and that these wheel speeds are each bounded to fall in the range $[-1, 1]$ for some choice of units of distance.

The configuration of the robot is given by $q = (x, y, \theta)$, the location and orientation of a frame attached to the center of the robot, with respect to some world coordinate system. Although the wheel speeds are directly controlled by the motors, choose the generalized velocity of the robot in its own frame as the controls. There is a one-to-one mapping between wheel speeds and the generalized velocity of the robot. The motion of the robot is then given by

$$q(T) = q(0) + \int_0^T \mathcal{R}(\theta(t))u(t)dt, \tag{1}$$

where $u = ({}^R\dot{x}, {}^R\dot{y}, {}^R\dot{\theta})$ is the generalized velocity in the robot frame, and $\mathcal{R}$ is the matrix that transforms the velocity into the world frame (formed by replacing the upper left block of a 3x3 identity matrix with a 2x2 rotation matrix).

We have studied this model in previous work [1], and in that work, derived a complete geometric description of the types of trajectories that might be optimal. However, until now, the problem of determining the precise trajectory that connects any two particular configurations has evaded solution. This solution is the primary contribution of the current paper; the current paper also presents a sampled mapping that shows regions of configuration space where each trajectory type is optimal. We will use several results from [1]:

1) Trajectories with piecewise-constant controls are sufficient to achieve optimality for bounded wheel speed model; these controls are chosen from 14 discrete controls, including clockwise and counterclockwise spins in place about robot center C_s, forward or backward translation in three directions, and clockwise or counter-clockwise rotations about three points (C_1, C_2, C_3 in figure 1) placed in an equilateral triangle about the robot.
2) We classify possibly-optimal trajectories into three types: singular trajectories (which may include translation), roll trajectories (which include only rotations all in the same direction), and shuffle trajectories (which include only rotations; every fourth rotation is in the opposite direction of the others).
3) Both *roll* and *shuffle* trajectories are periodic, in the sense that for each trajectory there exists a duration T_p such that any two configurations along a trajectory separated by T_p represent effectively a pure translation parallel to a fixed line in the plane, called the control line. (In [7], it was shown that no more than one period of a shuffle may be optimal; we show additionally in this paper that no more than one period of a roll may ever be optimal.)

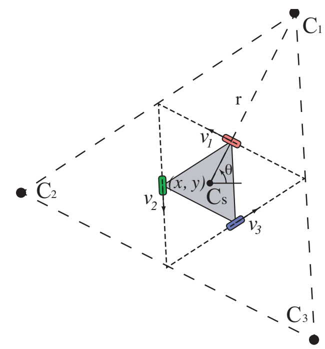


Fig. 1: The shape and the notation of the model

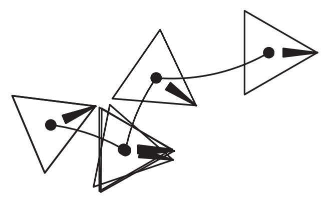


Fig. 2: A sample time-optimal trajectory

The recent Ph.D. thesis of Andrei Furtuna [6] attacks a more-general problem: solving the time-optimal trajectories for a rigid body in the plane with polyhedral bounds on generalized velocity controls; the omni-directional robot problem is a special case. In [6], a complete analytical method is presented for solving for the time-optimal singular trajectory between a pair of configurations; therefore, the current paper focuses entirely on the problem of solving *roll* and *shuffle* trajectories.

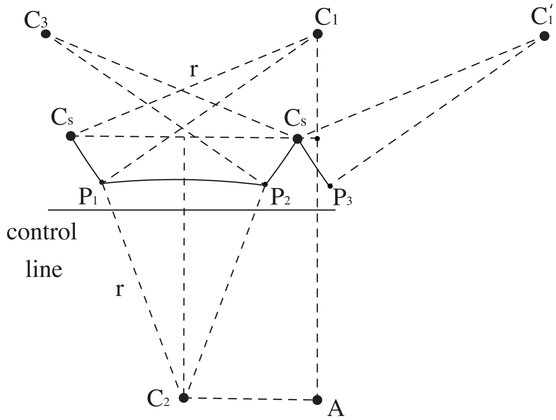


Fig. 3: Geometry of a shuffle trajectory

We assume the angular speed for spins is 1. The angular velocity for turns is $1/3$. Those facts imply that distance from C_i to C_s is four times the distance from C_s to each wheel. For details of derivations of these controls, see [1]. Each roll or shuffle trajectory is described as a sequence of rotation centers, with a plus or minus superscript to denote the direction of rotation (+ is counterclockwise). For example, one period of a *shuffle* trajectory might be of the form $C_s^+C_1^+C_2^-C_3^+C_s^+$ in figure 3. As we will prove later, no optimal shuffle trajectory can be longer than one period. So, no control can appear more than two times, and at most one control may be repeated.

A. Related Work

Dynamic models of omni-directional vehicle with or without slip have been developed by Jung and Kim [11], and Williams *et al.* [10]. Optimal trajectories for omni-directional vehicle using a dynamic model have been studied by Choi *et al.* [19] and Kalmár-Nagy *et al.* [12]. However, because of the complexity of the model, the resulting algorithms have been numerical rather than analytical, and assume certain restrictions on the structure of trajectories considered. The kinematic model we study is less sophisticated, but does admit complete analytical solution. Other methods to study the motion of mobile robots and other mobile vehicles have also been studied. Optimal dynamic models for an underwater vehicle are studied by Chyba and Haberkorn [4]. A new distance function using nonholonomic constraints were used to find the time-optimal trajectory between Dubin's car and polygonal obstacles [18], [9].

Time-optimal trajectories for several other kinematic models have been found. A car that can only drive forwards with bounded forward velocity and steering angle was studied by Dubins [5]. Reeds and Shepp [14] found the shortest paths for a steered car that can drive both forwards and backwards. Sussmann and Tang [17] further refined the results by developing a general methodology for solving this type of problem. Souères, Boissonnat and Laumound [15], [16] mapped the pairs of configurations to optimal trajectories for the Reeds-Shepp car. Balkcom and Maon [2] discovered the time-optimal trajectories for a differential-drive robot with bounds on the wheel speeds; Chitsaz *et al.* [3] determined the trajectories for a differential-drive that minimizes the sum of the rotation of the two wheels.

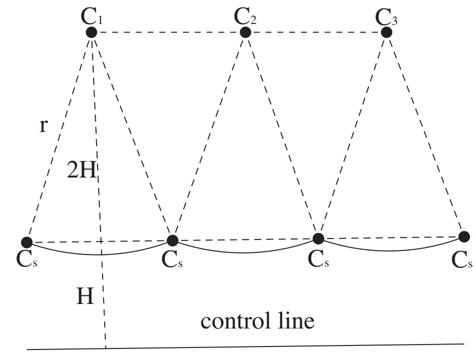


Fig. 4: Detailed roll trajectory.

II. Necessary conditions for optimality

We have shown in [7] that a necessary condition for time-optimal motion of a rigid-body with polyhedral bounds on generalized velocity controls, derived by application of Pontryagin's Principle [13], is that for any time-optimal trajectory of the robot, there must exist constants k_1, k_2, and k_3, not all zero, such that at any time along the trajectory, the generalized velocity of the robot $(\dot{x}, \dot{y}, \dot{\theta})$ must maximize the quantity H called *Hamiltonian*, given by:

$$H = k_1\dot{x} + k_2\dot{y} + \dot{\theta}\left(k_1 y - k_2 x + k_3\right). \tag{2}$$

Furthermore, H must be constant and positive over the trajectory.

The omni-directional robot model is a special case of such a rigid body. There is a geometric interpretation of this necessary condition. Without loss of generality, assume $k_1^2 + k_2^2 = 1$. (For this vehicle, we have shown optimality may be achieved without use of the $k_1 = k_2 = 0$ case, and non-zero scaling of the constants does not affect the maximization condition.) Let there be a line in the plane with direction given by the vector (k_1, k_2); let the signed distance of the line from the origin be k_3. Call this line the control line.

For trajectories containing no translations (such as the roll and shuffle trajectories we study in this paper), the Hamiltonian equation can also be given by:

$$H = {}^L y_c \dot{\theta}, \tag{3}$$

where ${}^L y_c$ is the distance of the current rotation center from the control line. The choice of constants determines the placement of the control line in the plane. Geometrically, we can say that for any optimal trajectory, we must be able to draw a particular line in the plane, such that at every time during the trajectory, the current rotation center, scaled by the angular velocity, is, among all possible rotation centers used to control the robot, as far as possible from the line.

Although we omit the technical details in the current paper, this geometric condition directly leads to the classification of trajectory types as roll, shuffle, or singular. If the robot is quite far from the control line, the Hamiltonian is maximized by a simple spin in place, since $|\dot{\theta}|$ is greatest for a spin. If the robot is somewhat closer to the control line, the robot may spin until one of the rotation centers C_1, C_2, or C_3 is three times as far from the line as the center of the robot; the robot then turns about the center until another center is far enough from the control line to cause another control switch; such a trajectory may be a roll or a shuffle, depending on how close the robot is to the control line. Singular trajectories may occur when two of the rotation centers C_1, C_2, and C_3 are equidistance from the control line, and on opposite sides of the control line; this situation permits translation.

A. Time-optimal trajectories never exceed one period

A *switch* is a time during a trajectory at which the control changes. A *segment* of trajectory is the subsection of the trajectory between two switches.

Theorem 1: No time-optimal roll or shuffle trajectory for the omni-directional vehicle can be longer than one period. (One period of shuffle may contain at most four switches, and one period of roll may contain at most six switches.)

Proof: The result for shuffle trajectories has been previously proven in [7]. The proof for rolls can be divided into three cases. In each case, the method is, given a trajectory longer than one period, to construct a new trajectory that takes the same time, but does not satisfy the Pontryagin Principle. Since the new trajectory does not satisfy necessary conditions and cannot be optimal, the original equal-time trajectory also cannot be optimal.

Case 1: A period of a roll that starts and ends with spin, and the turn is longer than $60°$. Construct a new trajectory by reflecting the $60°$ segments of each turn across the line passing P_1 and P_2. This is trajectory $C_sC_1C_1'C_1C_s\cdots$ in figure 5. Denote $C_1A = d_1$, and $C_1'A = d_2$. In this case, $H = \frac{1}{3}d_1 \neq \frac{1}{3}d_2$. So the new trajectory violates the necessary condition for optimality, because in this trajectory the Hamiltonian is not constant.

Case 2: A period of a roll that starts and ends with spin or turns, in which the maximum turn is shorter than $60°$. Reverse all the spins and replace turn controls with symmetric controls following the original path. Denote the distance from C_s to control line is d. Then the original $H = d = \frac{1}{3}d_1$, while the Hamiltonian for the reversed spin is $-d$. This new trajectory violates the necessary condition for optimality because the Hamiltonian is not constant during the trajectory.

Case 3: A period of roll that starts and ends with a turn longer than $60°$. The initial point could be on the original curve along P_1P_2 or P_2C_s. If it is on P_1P_2, follow the rest of period by $C_1C_1'C_s'C_2'C_2C_2'C_s'\cdots$ till the end of the period. The arc length of P_3P_4 and P_4P_5 are $60°$. This new trajectory violates the necessary condition for optimality. Denote the distance form $C_s^{'}$ to control line is d'. Because $d' \neq d$, the Hamiltonian is not constant during the trajectory. If the initial point is on P_2C_s, reflect the $60°$ part $P_iP_{i+1}, i = 3, 5, 7$ across the line passing S_is. This new trajectory also violates the necessary condition of optimality for the same reason as in Case 1.

■

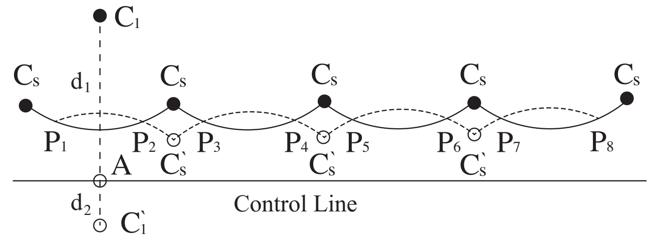


Fig. 5: Figure for proof that roll trajectory cannot be longer than one period.

III. Algorithm

The Hamiltonian is a constant over a time-optimal trajectory. We will see that for roll and shuffle trajectories, knowing the first and last control, as well as the start and goal configurations of the robot, this value of the Hamiltonian can be computed. Once the Hamiltonian has been computed, the complete evolution and time cost of the trajectory can be computed exactly.

Our algorithm loops over possible control choices for the first and last control of the trajectory, and for each choice computes a candidate Hamiltonian value and time cost. The algorithm can be described in algorithm 1. Equations for the computation of the Hamiltonian depend on whether the trajectory is a roll or shuffle, and whether the first and last controls are spins or turns; we will now consider each case.

Algorithm 1: Main algorithm flow

```
Input: t_min = t_singular
for control i at initial do
    for control j at goal do
        if i is spin && j is spin then
            Compute H (eq. 7), time t_r (eq. 6, 9)
            Compute H (eq. 17), time t_s (eq. 6, 19)
        if one of i and j is spin and the other is turn
        then
            Compute H (eq. 13), time t_r (eq. 6, 15, 16)
            Compute H (eq. 29), time t_s
             (eq. 6, 31 or 6, 34) (2 cases)
        if i is turn && j is turn then
            Compute H (eq. 10), time t_r (eq. 6, 12)
            Compute H (eq. 20, 26), time t_s
             (eq. 6, 22 or 6, 28, 25) (2 cases)
        t_min = min(t_min, t_r, t_s)
```

A. Calculating time for each trajectory

Each trajectory that takes the robot from initial configuration to the goal configuration contains n segments. All the segments except the first and the last segments are complete, in the sense that the duration between control switches can be directly computed from the necessary conditions and the value of the Hamiltonian.

Denote the angular velocity and time for each segment by ω_i and $t_i, i = 1, 2, \ldots, n$. Assume initial and goal orientation are θ_s and θ_g. We have:

$$\begin{aligned} \sum_{i=1}^{n} t_i\omega_i + \theta_s &= \theta_g \\ t_{total} &= \sum_{i=1}^{n} t_i. \end{aligned} \tag{4}$$

In equation 5, the equal sign means the same angle on the circle. Since both initial and goal orientation are in $[-\pi, \pi]$, we can construct a function $S_\theta(\theta_s, \theta_g)$ that calculates the total change of orientation through the trajectory; denote $\sum_{i=2}^{n-1} t_i\omega_i$ by θ_{inner}, $\sum_{i=2}^{n-1} t_i$ as t_{inner}, then we have:

$$\begin{aligned} t_1\omega_1 + \theta_{\text{inner}} + t_n\omega_n &= S_\theta(\theta_s, \theta_g) \\ t_1\omega_1 + t_n\omega_n &= S_\theta(\theta_s, \theta_g) - \theta_{\text{inner}} \quad (5) \\ t_{\text{total}} &= t_1 + t_{\text{inner}} + t_n. \quad (6) \end{aligned}$$

B. Algorithm for each case

1) Roll: spin-spin: Assume a trajectory is a roll, and the first and last controls are both spins (take figure 4 as an example). In this case, the control line should be parallel to the line passing all the spin centers C_s, in which one is the first and the other is the last (order is not important). Denote the distance between first and last rotation centers by Δx.

Denote the segment length by Δl; the orientation change for each segment by $\Delta\alpha$. Δl and $\Delta\alpha$ of roll in the control line can be calculated as follows:

$$\begin{aligned}\Delta l_{\text{spin}} &= 0\\ \Delta l_{\text{turn}} &= 2\sqrt{r^2-4H^2}\\ \Delta\alpha_{\text{spin}} &= 2\pi/3 - 2\text{acos}(2H/r)\\ \Delta\alpha_{\text{turn}} &= 2\text{acos}(2H/r),\end{aligned}$$

since the distances from C_1 and C_s to the control line are $3H$ and H, which can easily be derived from the fact that Pontryagin's Principle indicates that H must be constant over a trajectory.

Recall that there are n segments, and let $k = \lfloor n/2 \rfloor$ be the number of turns between the first and last control

$$\begin{aligned}\Delta x &= k \times \Delta l_{\text{turn}} = 2k\sqrt{r^2-4H^2}\\ H &= \frac{1}{2}\sqrt{r^2 - \frac{\Delta x^2}{4k^2}}.\end{aligned} \tag{7}$$

The Hamiltonian value H can be calculated by looping over integer values of k, which is less or equal to 3, since optimal roll trajectory cannot contain more than six switches. If $H \in \left(2/\sqrt{3}, 2\right]$, we can then calculate the time for this roll trajectory by using equation 6, with $\omega_1 = \omega_2$ in this case.

$$\theta_{\text{inner}} = k * \Delta\alpha_{\text{turn}} + (k-1) * \Delta\alpha_{\text{spin}} \tag{8}$$
$$t_{\text{inner}} = 3k * \Delta\alpha_{\text{turn}} + (k-1) * \Delta\alpha_{\text{spin}}. \tag{9}$$

2) Roll: turn-turn: Assume the trajectory is roll, and both first and last control are turns (take figure 4 as an example). For this case, we consider rotation centers C_1 and C_3. Denote $\|C_1 - C_3\| = \Delta c$. Also, $\omega_1 = \omega_n$, using equation 6 to solve for the time.

$$\begin{aligned}\Delta c &= k \times \Delta l_{\text{turn}} = 2k\sqrt{r^2-4H^2}\\ H &= \frac{1}{2}\sqrt{r^2 - \frac{\Delta x^2}{4k^2}}\end{aligned} \tag{10}$$

$$\begin{aligned}\theta_{\text{inner}} &= k\Delta\alpha_{\text{spin}} + (k-1)\Delta\alpha_{\text{turn}}\\ t_{\text{inner}} &= k\Delta\alpha_{\text{spin}} + 3(k-1)\Delta\alpha_{\text{turn}} \qquad (11)\\ t_1 + t_n &= (S_\theta(\theta_i, \theta_g) - \theta_{\text{inner}})/\omega_1. \qquad (12)\end{aligned}$$

3) Roll: turn-spin: Assume the trajectory is a roll, and the first and last controls are different; one is a turn and the other is a spin. Take figure 6 as an example, with start at A and goal at the rightmost C_s. (The geometry also works if the positions are reversed, with A the goal.)

Define $d = \|C_1 - C_s\|$, and define $d_w = \|A - C_s\|$. For n total segments, there are $k = \lfloor n/2 \rfloor - 1$ spins between the first and last control. In figure 6, $\|C_1 - C_2\| = \Delta l_{\text{turn}} = 2\|C_3 - B\|$.

$$\begin{aligned}\sqrt{d^2 - 4H^2} &= k \times \Delta l_{\text{turn}} = (2k+1)\sqrt{r^2-4H^2}\\ H &= \sqrt{\frac{(2k+1)^2 r^2 - d^2}{2(2k+1)^2 - 4}}\end{aligned} \tag{13}$$

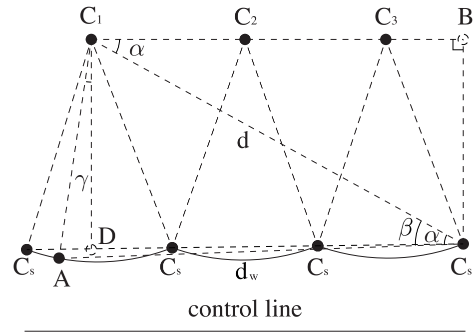


Fig. 6: A simplified case of a roll trajectory with different types of first and last controls

$$\begin{aligned}\theta_{\text{inner}} &= k\Delta\alpha_{\text{spin}} + k\Delta\alpha_{\text{turn}} = 2k\pi/3\\ t_{\text{inner}} &= k\Delta\alpha_{\text{spin}} + 3k\Delta\alpha_{\text{turn}}\\ t_1\omega_1 + t_n\omega_n &= S_\theta(\theta_i, \theta_g) - \theta_{\text{inner}}.\end{aligned} \tag{14}$$

To calculate the time, since $\omega_1 \neq \omega_n$, we need to calculate the time for the turn. Denote the angle $\angle AC_sC_1$ by β, the angle $\angle C_sC_sC_1$ by α, the angle $\angle AC_1D$ by γ. If the projection of d_w onto the control line is longer than $\|C_1 - B\|$, then the first control is longer than half of a complete turn and vice versa. Therefore, the angle for the first control is $\Delta\alpha_{\text{ratate}}/2 \pm \gamma$. Use equation 6 to find the total time.

$$\begin{aligned}\alpha &= \text{asin}(2H/d)\\ \beta &= \text{acos}((d_w^2 + d^2 - r^2)/(2 \times d_w \times d))\\ \Delta x &= d_w\cos(\beta - \alpha) - (k + 1/2)\Delta l_{turn}\\ \text{sign} &= \Delta x/|\Delta x|\\ \gamma &= \text{acos}((2H + d_w\sin(\beta-\alpha))/r)\\ t_1 &= \frac{1}{2}\alpha_{\text{turn}} + \text{sign} \times \gamma \qquad (15)\\ t_n &= (S_\theta(\theta_i, \theta_g) - \theta_{\text{inner}} - t_1\omega_1)/\omega_n. \qquad (16)\end{aligned}$$

4) Shuffle: spin-spin: Calculating the value of Hamiltonian for a shuffle trajectory is almost the same as for the roll, except the segment lengths are different. Consider an example shuffle trajectory in figure 3 that starts with a spin at C_s on the left and ends with spin at C_s on the right. At P_1, the control switches directly from C_1 to C_2, so $\|C_1 - C_2\| = \sqrt{3}R$ and $\|C_1 - A\| = 6H$. Denote the distance between initial and goal by d_w. We know from the structure of the shuffle trajectory that $\omega_1 = \omega_n$, and the sign of ω_3 (the angular velocity corresponding to the turn around C_2) is different from all the other angular velocities. Then, we have:

$$\begin{aligned}\Delta\alpha_{\text{spin}} &= 2\pi/3 - 2\text{acos}(2H/r)\\ \Delta\alpha_{\text{turn1}} &= \text{acos}\frac{2H}{r} - \frac{\pi}{6} - \text{acos}\frac{6H}{\sqrt{3}r}\\ \Delta\alpha_{\text{turn2}} &= 2(\frac{\pi}{6} - \text{acos}\frac{6H}{\sqrt{3}r})\end{aligned}$$

$$d_w = 2\sqrt{r^2-4H^2} - 2\sqrt{3r^2 - 36H^2} \tag{17}$$

$$\theta_{\text{inner}} = 2 \times 3\Delta\alpha_{\text{turn1}}\omega_2 + 3\Delta\alpha_{\text{turn2}}\omega_3$$
$$t_{\text{inner}} = 3 \times 2\Delta\alpha_{\text{turn1}} + 3\Delta\alpha_{\text{turn2}} \quad (18)$$
$$t_1 + t_n = (S_\theta(\theta_i, \theta_g) - \theta_{\text{inner}})/\omega_1. \quad (19)$$

5) Shuffle: turn-turn: Assume the trajectory is shuffle and both the first and last controls are turns (take figure 3 as an example). There are two cases in this type of trajectory.
Case 1: The first and last turns have the same sign of angular velocity. Consider the case where the first control is C_1 and the last control is C_3 (the geometry also works with the reverse order, and gives the same time cost). Denote the distance between the first and last rotation center by Δc. Then $\|C_1 - C_3\| = \Delta c$, $\omega_1 = \omega_n$, and the sign of ω_2 is different from all the other angular velocities. Use equation 6 to calculate the total time:

$$c_w = 2\sqrt{3r^2 - 36H^2}$$
$$H = \frac{1}{6}\sqrt{3r^2 - \frac{c^2}{4}} \quad (20)$$

$$\theta_{\text{inner}} = 3\Delta\alpha_{\text{turn2}}\omega_2$$
$$t_{\text{inner}} = 3\Delta\alpha_{\text{turn2}} \quad (21)$$
$$t_1 + t_n = (S_\theta(\theta_i, \theta_g))/\omega_1. \quad (22)$$

Case 2: The first and last turns have different signs of angular velocity. First, consider the case where the control sequence is from C_1 to C_2 (the control sequence from C_2 to C_3 is symmetric). Denote the distance between the two rotation centers by c_w. For the sequence from C_1 to C_2, $\Delta c = \sqrt{3}r$. Assume the start is on the arc C_sP_1, denoted by P, let $\|C_3 - P\| = d_1$, and use equations 6 to calculate total time.

$$\Delta c = \sqrt{3}r \quad (23)$$
$$t_1\omega_1 + t_2\omega_2 = S_\theta(\theta_i, \theta_g) \quad (24)$$
$$t_1 = \text{acos}\left(\frac{r^2 + 3r^2 - d_1^2}{2 \times r \times \sqrt{3}r}\right). \quad (25)$$

(In this case, the Hamiltonian value is not needed.)

Now consider the control sequence from C_2 to C_4. Define $\Delta c = \|C_2 - C_4\|$, $d_2 = \|C_2 - C_s\|$. Assume the goal is on arc C_sS_3 and denoted by B. Define $d_3 = \|C_2 - B\|$, use equations 6 to calculate total time.

$$\sqrt{\Delta c^2 - 36H^2} = 2\sqrt{r^2 - 4H^2} - \sqrt{3r^2 - 36H^2} \quad (26)$$

$$\sqrt{d_2 - 16H^2} = \sqrt{r^2 - 4H^2} - \sqrt{3r^2 - 36H^2}$$
$$\theta_{\text{inner}} = 3\alpha_{\text{turn1}}\omega_2 + \alpha_{\text{spin}}\omega_3$$
$$t_{\text{inner}} = 3\alpha_{\text{turn1}} + \alpha_{\text{spin}} \quad (27)$$
$$t_1\omega_1 + t_n\omega_n = S_\theta(\theta_i, \theta_g) - \theta_{\text{inner}}$$
$$t_1 = [\text{acos}\left(\frac{r^2 + \Delta c^2 - d_2^2}{2 \times r \times \Delta c}\right) - \text{acos}\left(\frac{r^2 + \Delta c^2 - d_3^2}{2 \times r \times \Delta c}\right)]/\omega_1. \quad (28)$$

6) Shuffle: turn-spin: **Case 1**: Initial and final turn and spin have same sign of angular velocity. Take figure 3 as example. Consider the control sequence from C_1 to C_3, denote the distance between them by Δc. Assume initial A is on arc C_sP_1, Define $d_1 = \|A - C_2\|$. Use equations 6 to calculate total time.

$$\sqrt{\Delta^2 c - 4H^2} = 2\sqrt{3r^2 - 36H^2} - \sqrt{r^2 - 4H^2} \quad (29)$$

$$\theta_{\text{inner}} = 3\Delta\alpha_{\text{turn1}}\omega_2 + 3\Delta\alpha_{\text{turn2}}\omega_3$$
$$t_{\text{inner}} = 3\Delta\alpha_{\text{turn1}} + 3\Delta\alpha_{\text{turn2}} \quad (30)$$
$$t_1\omega_1 + t_n\omega_n = S_\theta(\theta_i, \theta_g)$$
$$t_1 = \frac{\text{acos}\left(\frac{r^2+3r^2-d_1^2}{2\times r\times\sqrt{3}r}\right) - \frac{\pi}{6}}{\omega_1}. \quad (31)$$

Case 2: Initial and final turn and spin have different sign of angular velocity. Take figure 3 as example, consider the case that control sequence from C_2 to C_s. Define $\Delta c = \|C_2 - C_s\|$, assume initial in on arc P_1P_2 denoted by A, define $d_1 = \|C_s - A\|$. Use equations 6 to calculate total time.

$$\sqrt{\Delta c^2 - 16H^2} = \sqrt{r^2 - 4H^2} - \sqrt{3r^2 - 36H^2} \quad (32)$$

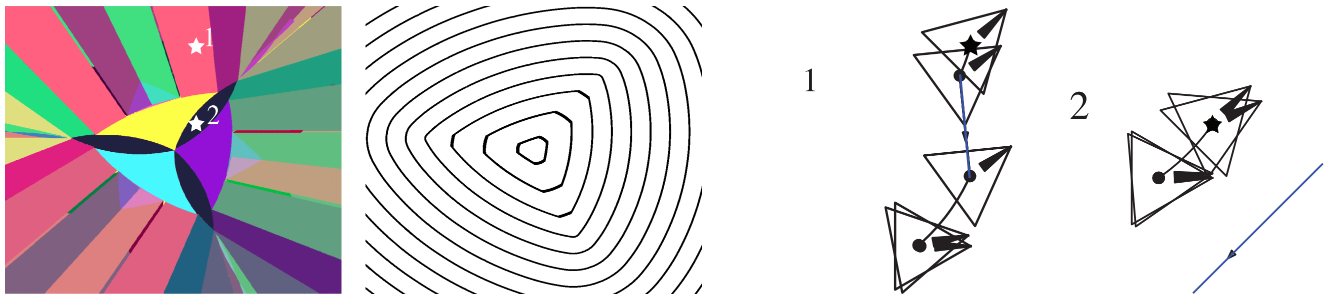
$$\theta_{\text{inner}} = 3\Delta\alpha_{\text{turn1}}\omega_2$$
$$t_{\text{inner}} = 3\Delta\alpha_{\text{turn1}} \quad (33)$$
$$d_2 = 2r\sin(\Delta\alpha/2)$$
$$t_1 = [\text{acos}\left(\frac{r^2 + \Delta c^2 - d_1^2}{2 \times r \times \Delta c}\right) - \text{acos}\left(\frac{r^2 + \Delta c^2 - d_2^2}{2 \times r \times \Delta c}\right)]/\omega_1. \quad (34)$$

IV. Simulation, Experiments, and Comparison

We implemented this algorithm in C. Compared to the numerical method in [8], the analytical method is more precise, and computes about 70 configurations per second which is faster by a factor of about 100 than the numerical method on a relative modern 2008 desktop. We sampled a few slices of the configuration space at $\theta_s = \pi/4$, $\pi/2$, 0 and π with x_s, y_s in the range $[-5, 5]$ and goal begin origin. Figure 7 is the result for trajectories with $\theta_s = \pi/4$ together with corresponding time contour plots and sample trajectories at star 1 and 2. The contour map shows isocost curves for time costs in increments of 0.5 seconds. Figures 8, 9 and 10 show similar results for other slices.

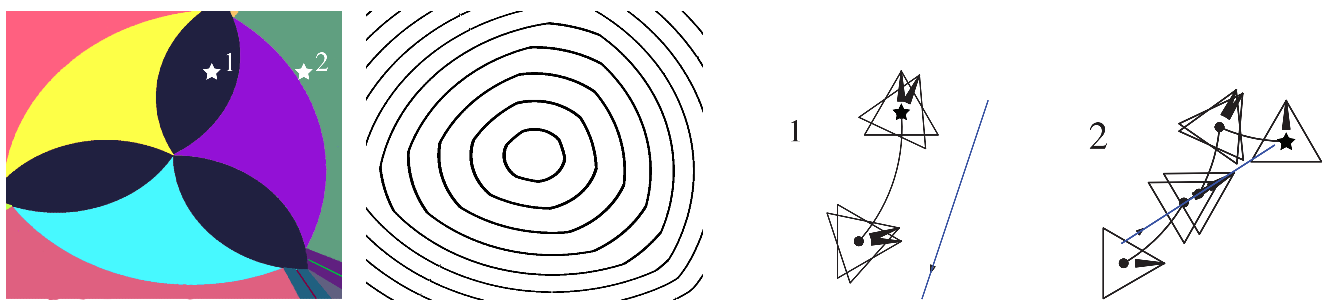
The result shows that the time-optimal trajectories for an omni-directional vehicle are mostly roll and shuffle for initial configurations not far (combine the distance and the angle difference) from the goal, and singulars otherwise. For initial angle close to the goal angle, three of six fastest translation directions that may be close to starting angle may be used by time-optimal trajectories depending on starting location (figures 7, 8). When the starting angle is very close to the goal orientation, all six fastest translation directions may be used (figure 9). And if the angle difference between initial and goal are big enough, the angle cost dominates (figure 10). Figure 9 and 10 also showed that there are equivalent trajectories for some configurations, because the goal orientation can be attained by rotating either in the positive or negative direction with the same time cost. An example is shown in figure 10c.

For configurations that are far away from the goal, we observe that a simple control strategy, which is a special case of singular sometimes turns out to be time-optimal: turn the fastest translation direction to face the goal; drive to the goal; turn to the correct angle. Under these circumstances, is it worthwhile to implement the complete algorithm described here? We compared the time-optimal trajectory with the simple turn-drive-turn strategy for some configurations closer to the origin. As the distance from the start to the goal change



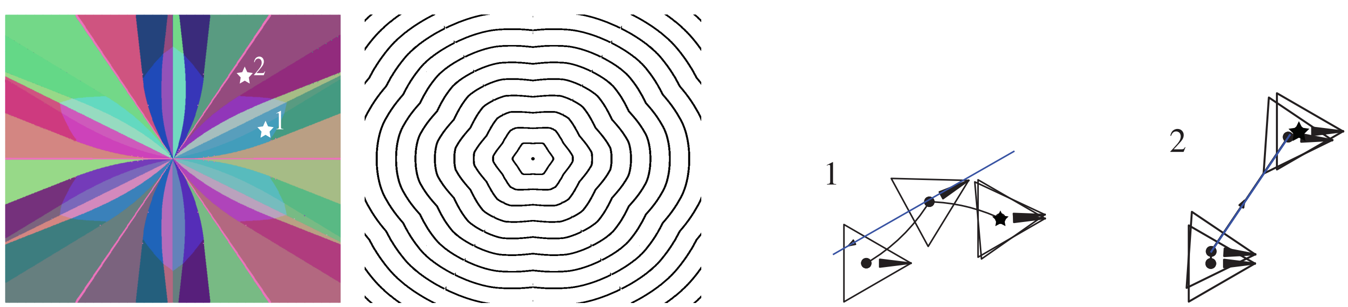
(a) trajectory types for $\pi/4$ slice (b) isocost curves (c) singular from $(1, 4, \pi/4)$ to origin (d) roll from $(1, 1, \pi/4)$ to origin

Fig. 7: Trajectories for $\theta_s = \pi/4$



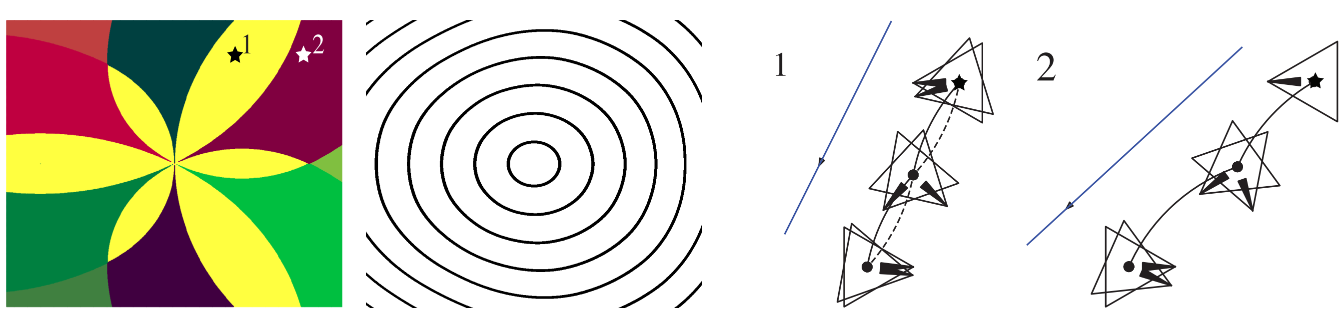
(a) trajectory types for $\pi/2$ slice (b) time for $\pi/2$ slice (c) $(1, 3, \pi/2)$ (d) $(4, 3, \pi/2)$

Fig. 8: Trajectories for $\theta_s = \pi/2$



(a) trajectory types for 0 slice (b) time for 0 slice (c) $(3, 1, 0)$ (d) $(2, 3, 0)$

Fig. 9: Trajectories for $\theta_s = 0$



(a) trajectory types for π slice (b) time for π slice (c) $(2, 4, \pi)$ (d) $(4, 4, \pi)$

Fig. 10: Trajectories for $\theta_s = \pi$

TABLE I: comparison of time optimal trajectory and turn - drive - turn

distance to origin	better in percentage
0 - 1	10.1697%
1 - 2	17.9340%
2 - 3	22.5781%
3 - 4	23.6106%
4 - 5	22.9535%
5 - 6	21.8520%
6 - 7	20.7536%
7 - 8	19.7472%
8 - 9	18.5558%
9 - 10	17.1047%

TABLE II: all types of trajectory for three or more segments

segment length	number of singular types	number of roll types	number of shuffle types
3	24	12	24
4	33	12	24
5	47	12	24
6	51	12	0
7	20	12	0
8	18	0	0

from 1 to 10, the optimal control averagely beats the turn-drive-turn by different percentages, shown in the table I. It turns out for configurations that are far away or extremely close to the goal, the turn-drive-turn strategy are almost as good as optimal trajectories. However, for the configurations that are not very far away from the goal, the time optimal trajectories are faster by about 20% on average.

At a high level, we classify trajectories as “roll”, “shuffle”, or “singular”, but we might also simply describe a trajectory type based on the sequence of controls that occur in the trajectory. Using this counting method, we found 459 types of trajectory that are optimal for at least some configurations. This number compares to 40 trajectory types for differential-drive and 46 for Reed-Shepp car.

Trajectories with only one or two segments are hard to categorize as a roll or a shuffle, but we may explicitly enumerate all 134 different types. For trajectories with more than three segments, enumeration is more complicated. For rolls and shuffles, because of the strong constraints on the control sequences, the exact number of types can be enumerated; the results are listed in table II. Singulars have less constraints, and the situation is more complicated. We sample the space of starting configurations in an area near the origin and found 193 different types of singular trajectories occurred; this is only a lower bound.

V. CONCLUSION

We developed an analytical method to efficiently find the time-optimal trajectory between configurations of an omni-directional vehicle. We sampled the space of starting configurations and used the algorithm to explore the distribution of the trajectory structures over the configuration space. We also explicitly counted the trajectory types for rolls and shuffles, and used the sampling to find a lower bound on the singular trajectory types. We also compared the time-optimal trajectories with other driving methods, and it turns out the the time optimal trajectories are somewhat faster for start configurations that are not too close or too far away from the goal.

REFERENCES

[1] Devin J. Balkcom, Paritosh A. Kavathekar, and Matthew T. Mason. Time-optimal trajectories for an omni-directional vehicle. *I. J. Robotic Res.*, 25(10):985–999, 2006.
[2] Devin J. Balkcom and Matthew T. Mason. Time optimal trajectories for bounded velocity differential drive robots. In *IEEE International Conference on Robotics and Automation*, 2000.
[3] Hamidreza Chitsaz, Steven M. LaValle, Devin J. Balkcom, and Matthew T. Mason. Minimum wheel-rotation paths for differential-drive mobile robots. In *IEEE International Conference on Robotics and Automation*, 2006. To appear.
[4] M. Chyba and T. Haberkorn. Designing efficient trajectories for underwater vehicles using geometric control theory. In *24rd International Conference on Offshore Mechanics and Artic Engineering*, Halkidiki, Greece, 2005.
[5] L. E. Dubins. On curves of minimal length with a constraint on average curvature and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, 79:497–516, 1957.
[6] A. A. Furtuna. *Minimum time kinematic trajectories for self-propelled rigid bodies in the unobstructed plane*. PhD thesis, Dartmouth College, 2011.
[7] Andrei A. Furtuna and Devin J. Balkcom. Generalizing Dubins curves: Minimum-time sequences of body-fixed rotations and translations in the plane. *I. J. Robotic Res.*, 29(6):703–726, 2010.
[8] Andrei A. Furtuna, Wenyu Lu, Weifu Wang, and Devin J. Balkcom. Minimum-time trajectories for kinematic mobile robots and other planar rigid bodies with finite control sets. In *IROS*, 2011.
[9] Paolo Robuffo Giordano and Marilena Vendittelli. Shortest paths to obstacles for a polygonal dubins car. *IEEE Transactions on Robotics*, 25(5):1184–1191, 2009.
[10] Robert L. Williams II, Brian E. Carter, Paolo Gallina, and Giulio Rosati. Dynamic model with slip for wheeled omnidirectional robots. *IEEE Transactions on Robotics*, 18(3):285–293, 2002.
[11] Myung-Jin Jung and Jong-Hwan Kim. Development of a fault-tolerant omnidirectional wheeled mobile robot using nonholonomic constraints. *I. J. Robotic Res.*, 21(5-6):527–540, 2002.
[12] Tamás Kalmár-Nagy, Raffaello D'Andrea, and Pritam Ganguly. Near-optimal dynamic trajectory generation and control of an omnidirectional vehicle. *Robotics and Autonomous Systems*, 46(1):47–64, 2004.
[13] L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze, and E. F. Mishchenko. *The Mathematical Theory of Optimal Processes*. John Wiley, 1962.
[14] J. A. Reeds and L. A. Shepp. Optimal paths for a car that goes both forwards and backwards. *Pacific Journal of Mathematics*, 145(2):367–393, 1990.
[15] P. Souères and J.-D. Boissonnat. Optimal trajectories for nonholonomic mobile robots. In J.-P. Laumond, editor, *Robot Motion Planning and Control*, pages 93–170. Springer, 1998.
[16] Philippe Souères and Jean-Paul Laumond. Shortest paths synthesis for a car-like robot. *IEEE Transactions on Automatic Control*, 41(5):672–688, May 1996.
[17] Héctor Sussmann and Guoqing Tang. Shortest paths for the Reeds-Shepp car: a worked out example of the use of geometric techniques in nonlinear optimal control. SYCON 91-10, Department of Mathematics, Rutgers University, New Brunswick, NJ 08903, 1991.
[18] M. Vendittelli, J. P. Laumond, and C. Nissoux. Obstacle distance for car-like robots. *IEEE Transactions on Robotics and Automation*, 15:678–691, 1999.
[19] Ji wung Choi, Renwick E. Curry, and Gabriel Hugh Elkaim. Obstacle avoiding real-time trajectory generation and control of omnidirectional vehicles. In *American Control Conference*, 2008.