# Real-time trajectory generation for omnidirectional vehicles

Tamás Kalmár-Nagy, Pritam Ganguly, Raffaello D'Andrea
Sibley School of Mechanical and Aerospace Engineering
Cornell University
Ithaca, NY 14853, USA

## Abstract

In this paper, we discuss an innovative method of generating near-optimal trajectories for a robot with omnidirectional drive capabilities, taking the second-order dynamics of the vehicle into account. The relaxation of optimality results in immense computational savings, critical in dynamic environments. In particular, a decoupling strategy for each of the three degrees of freedom of the vehicle is presented, along with a method for coordinating the degrees of freedom. A nearly optimal trajectory for the vehicle can typically be calculated in less than 1000 floating point operations, which makes it attractive for real-time control in dynamic and uncertain environments.

## 1 Introduction

Omnidirectional vehicles provide superior maneuvering capability compared to the more common nonholonomic (or "car-like") vehicles. The ability to move along any direction and to simultaneously attain a desired orientation make it an attractive option for dynamic environments. The annual RoboCup competition is an example of where omnidirectional vehicles have been used in computationally intensive, dynamic environments (Asada and Kitano, 1998, Veloso et al., 1999, D'Andrea et al., 2001, see also Frazzoli et al., 2000).

Most papers on trajectory control of omnidirectional vehicles have dealt with relatively static environments (Moore and Flann, 1999, Jung et al., 1999); the trajectory control is essentially performed by first building a geometric path and then by using feedback control to track the path. This strategy is effective when reaching the goal without collisions is much more important than time-optimality. In fast paced environments, however, the dynamic capabilities of the vehicles must be taken into account. Muñoz et al. (1994) presented methods for planning mobile robot trajectories by considering kinematic and dynamic constraints on the motion of the vehicle. Faiz and Agrawal (2000) re-

cently proposed a trajectory planning scheme for open-chain systems that takes the dynamics of the system into account, as well as inequality constraints. Watanabe et al. (1998) demonstrated that with a resolved-acceleration type feedback full omnidirectionality can be achieved with decoupled rotational and translational motion. A novel trajectory generation method based on a time-scaled artificial potential field was put forth by Tanaka et al. (1998).

The objective of this paper is to establish a real-time control strategy (voltages to the motors) that will move the robot to a given location, with zero final velocity, as quickly as possible. The algorithm presented is extremely efficient and robust to measurement and control errors. The organization of the paper is as follows. In section 2, the kinematic and dynamic model of an omnidirectional vehicle is presented. Section 3 shows that the translational and rotational degrees of freedom (DOF) of the vehicle can be independently controlled by imposing constraints on the control efforts. Section 4 describes the construction of one-dimensional minimum time and fixed time trajectories as well as the solution to the relaxed trajectory generation problem. Simulations are presented in section 5. The paper ends with some concluding remarks in section 6.

## 2 Kinematic and Dynamic Modeling of the Omnidirectional Vehicle

The omnidirectional drive consists of three sets of wheel assemblies equally spaced at 120 degrees from one another (see Figure 1). Each of the wheel assembly consists of a pair of "orthogonal wheels" (Pin and Killough, 1994) with an active (the propelling direction of the actuator) and a passive (free-wheeling) direction which are orthogonal to each other. The point of symmetry is assumed to be coincident with the center of mass (CM) of the robot.

### 2.1 Geometry of Vehicle

The schematic arrangement of the wheel assemblies is shown in Figure 2.
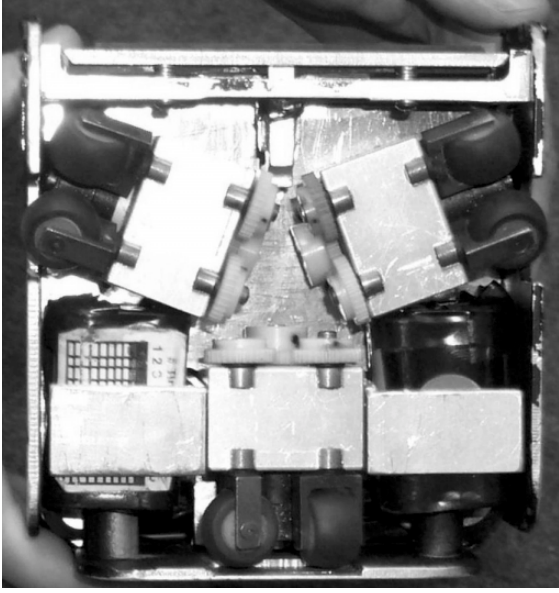
**Figure 1:** Bottom view of omnidirectional drive



**Figure 2:** Geometry of the omnidirectional vehicle

The positions ($\mathbf{P}_{0i}$) of these units are easily given in the $(X, Y)$ frame (which is fixed to the center of mass 0 of the robot) with the help of the rotation matrix ($\theta$ is the angle of counterclockwise rotation)

$$\mathbf{R}(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \quad (1)$$

as

$$\mathbf{P}_{01} = L \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad \mathbf{P}_{02} = \mathbf{R}\left(\frac{2\pi}{3}\right)\mathbf{P}_{01} = \frac{L}{2}\begin{pmatrix} -1 \\ \sqrt{3} \end{pmatrix}$$

$$\mathbf{P}_{03} = \mathbf{R}\left(\frac{4\pi}{3}\right)\mathbf{P}_{01} = -\frac{L}{2}\begin{pmatrix} 1 \\ \sqrt{3} \end{pmatrix} \quad (2)$$
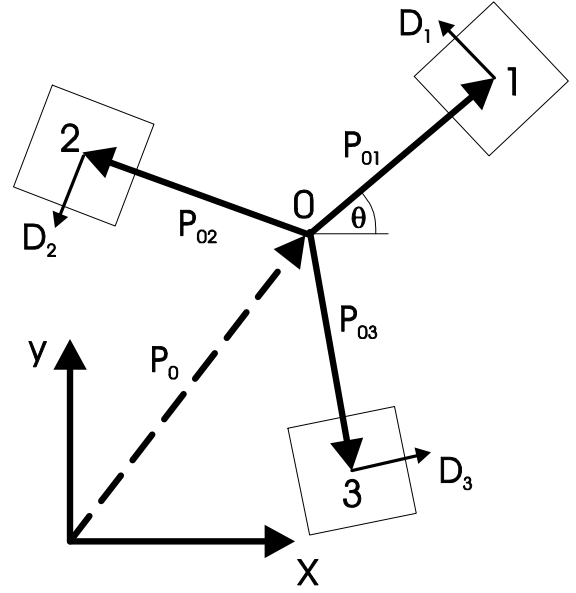
where $L$ is the distance of the drive units from the CM. The unit vectors $\mathbf{D}_i$ that specify the drive direction the $i$th motor (also relative to the CM) are given by

$$\mathbf{D}_i = \frac{1}{L}\mathbf{R}\left(\frac{\pi}{2}\right)\mathbf{P}_{0i}$$

$$\mathbf{D}_1 = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \qquad \mathbf{D}_2 = -\frac{1}{2}\begin{pmatrix} \sqrt{3} \\ 1 \end{pmatrix} \qquad \mathbf{D}_3 = \frac{1}{2}\begin{pmatrix} \sqrt{3} \\ -1 \end{pmatrix}$$

$$(3)$$

## 2.2 The Motor Characteristics

In general, the optimal control problem for independent actuator driven wheels are treated with either bounded velocity (Jung et al., 1999) or bounded acceleration (Renaud and Fourquet, 1997) but not both. With the no-slip condition, the force generated by a DC motor driven wheel (Ogata, 1997) is

$$f = \alpha U - \beta v \quad (4)$$

where $f$ [N] is the magnitude of the force generated by a wheel attached to the motor, and $v$ [m/s] is the velocity of the wheel. The constants $\alpha$ [N/V] and $\beta$ [kg/s] can be determined from experiments.

## 2.3 Equations of Motion

The vector $\mathbf{P}_0 = \begin{pmatrix} x & y \end{pmatrix}^T$ is the position of the CM in a Newtonian frame as shown in Figure 2. The drive positions and velocities are given by

$$\mathbf{r}_i = \mathbf{P}_0 + \mathbf{R}(\theta)\mathbf{P}_{0i} \quad (5)$$

$$\mathbf{v}_i = \dot{\mathbf{P}}_0 + \dot{\mathbf{R}}(\theta)\mathbf{P}_{0i} \quad (6)$$

while the individual wheel velocities are

$$v_i = \mathbf{v}_i^T\left(\mathbf{R}(\theta)\mathbf{D}_i\right) \quad (7)$$

Substituting equation (6) into equation (7) results in

$$v_i = \dot{\mathbf{P}}_0^T\mathbf{R}(\theta)\mathbf{D}_i + \mathbf{P}_{0i}^T\dot{\mathbf{R}}^T(\theta)\mathbf{R}(\theta)\mathbf{D}_i \quad (8)$$

The second term of the right hand side is just the tangential velocity

$$\mathbf{P}_{0i}^T\dot{\mathbf{R}}^T(\theta)\mathbf{R}(\theta)\mathbf{D}_i = L\dot{\theta} \quad (9)$$

The drive velocities are thus linear functions of the velocity and the angular velocity of the robot

$$\begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} = \begin{pmatrix} -\sin\theta & \cos\theta & L \\ -\sin\left(\frac{\pi}{3}-\theta\right) & -\cos\left(\frac{\pi}{3}-\theta\right) & L \\ \sin\left(\frac{\pi}{3}+\theta\right) & -\cos\left(\frac{\pi}{3}+\theta\right) & L \end{pmatrix}\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix}$$

$$(10)$$

Linear and angular momentum balance can be written as

$$\sum_{i=1}^{3} f_i \mathbf{R}\left(\theta\right) \mathbf{D}_i = m\ddot{\mathbf{P}}_0 \qquad (11)$$

$$L \sum_{i=1}^{3} f_i = J\ddot{\theta} \qquad (12)$$

where $f_i$ is the magnitude of the force produced by the $i$th motor, $m$ is the mass of the robot and $J$ is its moment of inertia.

Using equation (4) together with the balance laws (11, 12) and replacing the $v_i$'s from the kinematic relation (8) results

$$\sum_{i=1}^{3} \left(\alpha U_i - \beta v_i\right) \mathbf{R}\left(\theta\right) \mathbf{D}_i = m\ddot{\mathbf{P}}_0 \qquad (13)$$

$$L \sum_{i=1}^{3} \left(\alpha U_i - \beta v_i\right) = J\ddot{\theta} \qquad (14)$$

This system of differential equations can be put into the following form

$$\begin{pmatrix} m\ddot{x} \\ m\ddot{y} \\ J\ddot{\theta} \end{pmatrix} = \alpha\hat{\mathbf{P}}\left(\theta\right) \mathbf{U}\left(t\right) - \frac{3\beta}{2} \begin{pmatrix} \dot{x} \\ \dot{y} \\ 2L^2\dot{\theta} \end{pmatrix} \qquad (15)$$

with

$$\hat{\mathbf{P}}\left(\theta\right) = \begin{pmatrix} -\sin\theta & -\sin\left(\frac{\pi}{3}-\theta\right) & \sin\left(\frac{\pi}{3}+\theta\right) \\ \cos\theta & -\cos\left(\frac{\pi}{3}-\theta\right) & -\cos\left(\frac{\pi}{3}+\theta\right) \\ L & L & L \end{pmatrix} \qquad (16)$$

$$\mathbf{U}\left(t\right) = \begin{pmatrix} U_1\left(t\right) & U_2\left(t\right) & U_3\left(t\right) \end{pmatrix}^T \qquad (17)$$

The nondimensional equations of motion (see Kalmár-Nagy et al., 2002) become

$$\begin{pmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{\theta} \end{pmatrix} + \begin{pmatrix} \dot{x} \\ \dot{y} \\ \frac{2mL^2}{J}\dot{\theta} \end{pmatrix} = \mathbf{q}\left(\theta,t\right) \qquad (18)$$

where $\mathbf{q}\left(\theta,t\right)$ is the *control action*

$$\mathbf{q}\left(\theta,t\right) = \mathbf{P}\left(\theta\right) \mathbf{U}(t) \qquad (19)$$

where $\mathbf{P}\left(\theta\right)$ is $\hat{\mathbf{P}}\left(\theta\right)$ with $L=1$.

The following boundary conditions are specified

$$z(0) = z_0, \ \ z(t_f) = z_f, \ \ \dot{z}(0) = v_0, \ \ \dot{z}(t_f) = 0 \qquad (20)$$

where $z \in \{x, y, \theta\}$. Note that zero final velocity is specified. For non-zero final velocities the proposed class of solutions do not depend continuously on the boundary conditions, and continuity is essential to ensure robustness to position errors.

## 3 Control Strategy

Moving the robot from a point to another requires specifying the three voltages $U_i\left(t\right)$. Clearly, real-time optimal control of the coupled nonlinear differential equations (18) is not feasible with modest computational resources. There have been several attempts to overcome the complexity emerging from nonlinearity and coupling. d'Andréa-Novel et al. (1992) showed that dynamic feedback linearization can lead to the simplification of the control problem of 3-wheeled robots. Time optimal trajectories were constructed by Balkcom and Mason (2000) for differential drive robots.

The goal of this section is to find a simplified, computationally tractable optimal control problem whose solution yields feasible, albeit sub-optimal, trajectories.

The set of feasible voltages $\mathcal{U}$ is a cube given by

$$\mathcal{U}\left(t\right) = \left\{ \mathbf{U}\left(t\right)| \quad |U_i\left(t\right)| \leq 1 \right\} \qquad (21)$$

The set of allowable controls $\mathbf{P}\left(\theta\right)\mathcal{U}\left(t\right)$ depends on the vehicle orientation $\theta$. Since $\theta$ is responsible for the coupling of equations (18), replacing this set with a set of $\theta$-independent controls would greatly simplify the control problem. The maximal such set is found by taking the intersection of all possible sets of allowable controls

$$\mathcal{Q}\left(t\right) = \underset{\theta\in[0,2\pi)}{\cap} \mathbf{P}\left(\theta\right)\mathcal{U}\left(t\right) \qquad (22)$$

Obviously, any $\mathbf{q}\left(t\right)\in\mathcal{Q}$ is a suitable replacement for the $\theta$-dependent control action. In the following we give the explicit representation of $\mathcal{Q}$.

For a given $\theta$, the linear transformation $\mathbf{P}\left(\theta\right)$ maps the cube $\mathcal{U}\left(t\right)$ into the tilted cuboid (the set of allowable controls) $\mathbf{P}(\theta)\mathcal{U}\left(t\right)$. The matrix $\mathbf{P}\left(\theta\right)$ can be decomposed as a product of a rotation and a $\theta$-independent linear transformation

$$\mathbf{P}\left(\theta\right) = \mathbf{R}_z\left(\theta\right) \mathbf{P}\left(0\right) \qquad (23)$$

where

$$\mathbf{R}_z\left(\theta\right) = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad (24)$$

$$\mathbf{P}\left(0\right) = \frac{1}{2} \begin{pmatrix} 0 & -\sqrt{3} & \sqrt{3} \\ 2 & -1 & -1 \\ 2 & 2 & 2 \end{pmatrix} \qquad (25)$$

The linear transformation $\mathbf{P}\left(0\right)$ maps cube $\mathcal{U}\left(t\right)$ into the tilted cuboid $\mathbf{P}(0)\mathcal{U}\left(t\right)$ with a diagonal $|q_\theta| \leq 3$ along the $q_\theta$ axis. Note that since the mapping $\mathbf{P}\left(0\right)$ is linear, the surface of the cube gets mapped onto that of the cuboid. The transformation $\mathbf{R}_z\left(\theta\right)$ then rotates this cuboid about the $q_\theta$ axis (or equivalently: about its diagonal). The problem is to find the solid of revolution that is the intersection of all possible rotations

$\mathbf{R}_z\left(\theta\right)\mathbf{P}\left(0\right)\mathcal{U}\left(t\right)$ of the cuboid. This solid of revolution is characterized by (see Kalmár-Nagy et al., 2002 for details)

$$q_x^2\left(t\right) + q_y^2\left(t\right) \le r^2\left(q_\theta\right) \tag{26}$$

where the radius $r\left(q_\theta\right)$ is

$$r\left(q_\theta\right) = \frac{3 - |q_\theta|}{2} \tag{27}$$

With this result the equations of motion decouple

$$\ddot{x} + \dot{x} = q_x \tag{28}$$

$$\ddot{y} + \dot{y} = q_y \tag{29}$$

$$\ddot{\theta} + \frac{2mL^2}{J}\dot{\theta} = q_\theta \tag{30}$$

While these equations are linear, the control efforts are coupled, i.e. the constraint

$$q_x^2 + q_y^2 \le \left(\frac{3 - |q_\theta|}{2}\right)^2 \tag{31}$$

has to be satisfied. To simplify further discussion we focus on controlling only the translational degrees of freedom. In other words, the amount of control effort used for rotation control will be bounded by $|q_\theta| = 1$. The following results can be generalized for variable $q_\theta$ (Kalmár-Nagy et al., 2002).

# 4 Trajectory Generation

We are concerned with finding the time-optimal solution to the following system of linear equations

$$\ddot{x} + \dot{x} = q_x\left(t\right) \tag{32}$$

$$\ddot{y} + \dot{y} = q_y\left(t\right) \tag{33}$$

with the boundary conditions

$$x(0) = x_0, \ \ x(t_f) = x_f, \ \ \dot{x}(0) = v_{x0}, \ \ \dot{x}(t_f) = 0 \tag{34}$$

$$y(0) = y_0, \ \ y(t_f) = y_f, \ \ \dot{y}(0) = v_{y0}, \ \ \dot{y}(t_f) = 0 \tag{35}$$

and the constraint on the control inputs

$$q_x^2\left(t\right) + q_y^2\left(t\right) \le 1 \tag{36}$$

It can be shown (see Kalmár-Nagy et al., 2002) that the optimal control strategy is achieved when

$$q_x^2\left(t\right) + q_y^2\left(t\right) = 1 \quad t \in [0, t_f] \tag{37}$$

## 4.1 Minimum Time Trajectory
We first focus on solving the optimal control problem (32, 33, 34, 35) separately for $x$ and $y$. These separate solutions will in general result different final times

and so they must be synchronized to ensure that the boundary conditions (34, 35) of the composite problem are satisfied. It will be shown in section 4.3 that such a solution always exists.

The minimum time problem consists of finding a solution of the boundary value problem (32, 34) or (33, 35) with as small a $t_f$ as possible. It can be shown (for example Pierre, 1986), that this boundary value problem always has a solution, and that the control which minimizes the final time $t_f$ consists of two piecewise constant segments of magnitude 1. This type of control strategy is commonly referred to as "bang-bang" control. Koh and Cho (1999) formulated a path tracking problem for a two-wheeled robot based on bang-bang control. In our case, the following must be solved for $u$ ($|u| = 1$), $t_1$ and $t_2$:

$$\ddot{z} + \dot{z} = u, \ \ 0 < t \le t_1 \tag{38}$$

$$\ddot{z} + \dot{z} = -u, \ \ t_1 < t \le t_1 + t_2 = t_f, \tag{39}$$

$$z(0) = z_0, \ \ z(t_f) = z_f, \ \ \dot{z}(0) = v_0, \ \ \dot{z}(t_f) = 0 \tag{40}$$

where $z$ represents either $x$ or $y$. In this non-dimensional form the maximum achievable velocity has magnitude 1 (it also means $|v_0| < 1$). Subject to the boundary conditions (40), equations (38, 39) can be solved to yield

$$z\left(t\right) = \begin{cases} e^{-t}\left(u - v_0\right) + u\left(t - 1\right) + v_0 + z_0 & 0 \le t < t_1 \\ u\left(t_f - t - e^{t_f - t} + 1\right) + z_f & t_1 \le t \le t_f \end{cases} \tag{41}$$

$$v\left(t\right) = \begin{cases} \left(v_0 - u\right)e^{-t} + u & 0 \le t < t_1 \\ \left(e^{t_f - t} - 1\right)u & t_1 \le t \le t_f \end{cases} \tag{42}$$

The solution to this problem is given by (Kalmár-Nagy et al., 2002)

$$u = \text{sgn}\left(v_0 - \text{sgn}\left(c\right)\left(e^{|c|} - 1\right)\right) \tag{43}$$

$$t_2 = \ln\left(1 + \sqrt{D}\right) \tag{44}$$

$$t_1 = t_2 - c/u \tag{45}$$

where

$$c = z_0 + v_0 - z_f, \qquad D = 1 + e^{c/u}\left(\frac{v_0}{u} - 1\right) \tag{46}$$

Once $u$, $t_1$ and $t_2$ are determined, $z\left(t\right)$ is given by (41). The execution time for this trajectory is

$$t_{f\,\text{min}} = t_1 + t_2 = 2\ln\left(1 + \sqrt{D}\right) - c/u \tag{47}$$

## 4.2 Fixed Time Bang-Bang Control
If the control effort $u$ is decreased one intuitively expects the execution time $t_f$ to increase. Indeed, the following proposition holds (Kalmár-Nagy et al., 2002)

**Proposition 1** *For all* $t_f \geq t_{f\,\min}$ *there exists a* $\bar{u} \in (0, 1]$ *such that*

$$u = \bar{u}\,\mathrm{sgn}\left(\frac{v_0}{\bar{u}} - \mathrm{sgn}\,(c)\left(e^{\left|\frac{c}{\bar{u}}\right|} - 1\right)\right) \qquad (48)$$

*satisfies (38, 39). Further, the execution time*

$$\bar{t}_f\,(\bar{u}) = t_f\left(\bar{u}\,\mathrm{sgn}\left(\frac{v_0}{\bar{u}} - \mathrm{sgn}\,(c)\left(e^{\left|\frac{c}{\bar{u}}\right|} - 1\right)\right)\right) \quad (49)$$

*is a strictly monotonously decreasing function of* $\bar{u}$ *with* $\lim_{\bar{u}\to 0}\bar{t}_f \to \infty$ *and* $\bar{t}_f\,(1) = t_{f\,\min}$.

This result means that reaching the desired final position in the prescribed amount of time (provided this time is greater then the minimum time to reach this position with zero final velocity) is always possible with a reduced control effort bang-bang control.

## 4.3 Trajectory Synchronization

Generally, execution times for the minimum time problems for the different degrees of freedom will be different. To find a solution to the boundary value problem (32, 33, 34, 35), these solutions must be synchronized, that is $t_{fx} = t_{fy}$ should hold. Note that the execution time depends on the boundary conditions, as well as on the control effort, i.e.

$$t_{fz}\,(q_z) = t_f\,(z_0, z_f, v_{z0}, q_z) \qquad (50)$$

With this notation, we want to find the control efforts $q_x$ and $q_y$ for which

$$t_{fx}\,(q_x) = t_{fy}\,(q_y) \qquad (51)$$

Using the constraint (37) this is written as

$$t_{fx}\,(q_x) = t_{fy}\left(\pm\sqrt{1 - q_x^2}\right) \qquad (52)$$

or

$$\bar{t}_{fx}\,(\bar{q}_x) = \bar{t}_{fy}\left(\sqrt{1 - \bar{q}_x^2}\right) \qquad (53)$$

Since $\bar{t}_{fx}\,(\bar{q}_x)$ is a strictly monotonously decreasing function on $\bar{q}_x \in (0, 1]$, $\bar{t}_{fy}\left(\sqrt{1 - \bar{q}_x^2}\right)$ is strictly monotonously *increasing* there. Hence there exists a unique $\bar{q}_x$ satisfying (53).

## 5 Simulations

To demonstrate the computational efficiency and robustness of the algorithm, simulations were performed with the following parameters (so that the dimensional and nondimensional values are the same)

$$a_{\max} = 1\,\frac{\mathrm{m}}{\mathrm{s}^2}, \quad v_{\max} = 1\,\frac{\mathrm{m}}{\mathrm{s}} \qquad (54)$$

The following initial and final conditions were used

$$x_0 = y_0 = 0\,\mathrm{m}, \quad x_f = y_f = 1\,\mathrm{m} \qquad (55)$$

$$v_{x0} = 1\,\frac{\mathrm{m}}{\mathrm{s}}, \quad v_{y0} = 0\,\frac{\mathrm{m}}{\mathrm{s}} \qquad (56)$$

The position of the vehicle is updated sixty times a second ($dt = 0.017$ s). System latency is not considered. To account for errors present in the real system (arising from slippage, vision error, controller tracking error, etc.) noise was added to the actual position and velocity of the robot at the beginning of every simulation step. The disturbances were modeled as white noise, with amplitude of 1 cm and 3 cm/s from a uniform distribution for positions and velocities, respectively.

First, the equation

$$\bar{t}_{fx}\,(\bar{q}_x) = \bar{t}_{fy}\left(\sqrt{1 - \bar{q}_x^2}\right) \qquad (57)$$

is solved for $0 < \bar{q}_x < 1$ by bisection (with the stop condition $|\bar{t}_{fx} - \bar{t}_{fy}| < dt$). Then $q_x$ and $q_y$ are calculated. These control efforts are used for the next time step even if the corresponding times $t_{1x}$, $t_{1y}$ are less than $dt$. The velocities and positions are updated according to (41, 42) with $t_{1x} = t_{1y} = dt$. This is then repeated until both coordinates are within 5 centimeters of the target position and the velocities are less than 5 cm/s in absolute value. The results are shown in Figure 3. The trajectory is tracked excellently, showing the robustness of the proposed method. Note that the most pronounced effect of the noise on the trajectory is in the vicinity of the destination. The position error coupled with the discretization effect gives rise to jerky motion. This can be avoided by switching to open-loop control near the target. The computa-
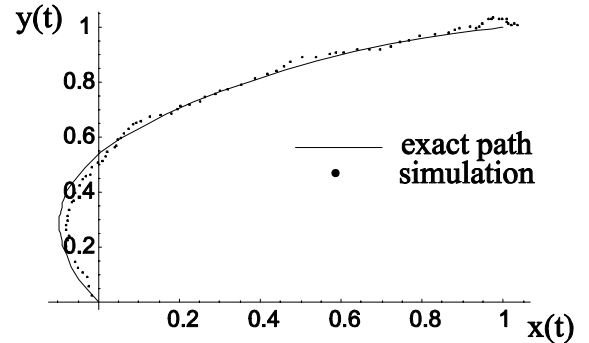


**Figure 3:** Simulation with noise

tional cost of the proposed algorithm is extremely low, around 300 FLOPS (floating point operations) for one timestep. The optimal control problem (32, 33, 34, 35) with coupled controls (37) was numerically solved in Matlab using the optimization routine *fminunc* with computational cost around 50000 FLOPS. Note that with our algorithm the whole trajectory does not have

to be generated (this is due to the fact that we restrict the form of the controls to bang-bang type), this is not the case with the usual optimization algorithms.

## 6 Conclusions

The main benefit of the omnidirectional drive mechanism is a simplification of the resulting control problem, which *greatly* reduces the computation required for generating nearly optimal trajectories. Current research addresses dynamics and control issues, such as estimation, coping with system latency, robustness, and optimal control. The proposed algorithms provide an efficient, yet high performance, method of path planning. The algorithms are in general conservative; this conservatism is small, however, and is justified by the extremely reduced computational costs. The extremely low computational cost means that these nearly optimal trajectories can be used extensively as low overhead primitives by higher level decision making strategies, allowing a large number of possible scenarios to be explored in real time. In addition, these algorithms can readily be used as a basis for obstacle avoidance.

These trajectory generation algorithms were used in the 2000 and 2001 RoboCup competition (D'Andrea et al., 2001), and were in large part responsible for the success of the Cornell team.

## References

[1]    Asada, M. and Kitano, H., editors, *RoboCup-98: Robot Soccer World Cup II*, In Lecture Notes in Computer Science, Springer, 1999.

[2]    d'Andréa-Novel, B., Bastin, G. and Campion, G., Dynamic feedback linearization of nonholonomic wheeled mobile robots, in *Proceedings of the IEEE International Conference on Robotics and Automation*, **3**, pp. 2527-2532, 1992.

[3]    D'Andrea, R., Kalmár-Nagy, T., Ganguly, P., Babish, M.: *The Cornell Robocup Team*, in Kraetzschmar, G., Stone P., Balch T. editors: Robot Soccer WorldCup IV, Lecture Notes in Artificial Intelligence. Springer, Berlin, Heidelberg, New York, 2001.

[4]    Balkcom, D. and Mason, M. Time Optimal Trajectories for Bounded Velocity Differential Drive Robots, in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '00)*, **3**, pp. 2499 - 2504, 2000.

[5]    Faiz, N. and Agrawal, S. K., Trajectory planning of robots with dynamics and inequalities in *Proceedings of the 2000 IEEE International Conference on Robotics and Automation*, **4**, pp. 3976-3982, 2000.

[6]    Frazzoli, E., Dahleh, M. A. and Feron, E., Robust Hybrid Control for Autonomous Vehicle Motion Planning, In *Conference on Decision and Control*, 2000.

[7]    Jung, M., Shim, H., Kim, H. and Kim, J., The Miniature Omni-directional Mobile Robot OmniKity-I (OK-I), in *Proceedings of the International Conference on Robotics and Automation*, **4**, pp. 2686-2691, 1999.

[8]    Kalmár-Nagy, T., Ganguly, P., D'Andrea, R., Near-Optimal Dynamic Trajectory Generation and Control of an Omnidirectional Vehicle, submitted to International Journal of Robotics Research, 2002.

[9]    Koh, K. C. and Cho., H. S., A Smooth Path Tracking Algorithm for Wheeled Mobile Robots with Dynamic Constraints, Journal of Intelligent and Robotic Systems, **24**, pp. 367-385, 1999.

[10]    Moore, K. L. and Flann, N. S., Hierarchial Task Decomposition Approach to Path Planning and Control for an Omni-Directional Autonomous Mobile Robot, in *Proceedings of the International Symposium on Intelligent Control/Intelligent Systems and Semiotics*, pp. 302-307, 1999.

[11]    Muñoz, V., Ollero, A., Prado, M., Simón, A., Mobile robot trajectory planning with dynamics and kinematics constraints, in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 342-351, 1994.

[12]    Ogata, K., *Modern Control Engineering.* Prentice-Hall, third edition, 1997.

[13]    Pierre, D. A., *Optimization Theory with Applications.* Dover, second edition, 1986.

[14]    Pin, F. G. and Killough, S. M., A New Family of Omnidirectional and Holonomic Wheeled Platforms for Mobile Robots, In *IEEE Transactions on Robotics and Automation*, **10**(4), pp. 480-489, 1994.

[15]    Renaud, M. and Fourquet, J. Y., Minimum time motion of a mobile robot with two independent, acceleration-driven wheels, In *Proceedings of the 1997 IEEE, International Conference on Robotics and Automation*, pp. 2608-2613, 1997.

[16]    Tanaka, Y., Tsuji, T., Kaneko, M., Morasso, P.G., Trajectory generation using time scaled artificial potential field, in *Proceedings of the 1998 IEEE/RSJ International Conference on Intelligent Robots and Systems*, **1**, pp. 223-228, 1998.

[17]    Veloso, M., Pagello, E. and Kitano, H., editors, *RoboCup-99: Robot Soccer World Cup III*, In Lecture Notes in Computer Science. Springer, 2000.

[18]    Watanabe, K., Shiraishi, Y., Tzafestas, S. G., Tang, J. and Fukuda, T., Feedback Control of an Omnidirectional Autonomous Platform for Mobile Service Robots, *Journal of Intelligent and Robotic Systems*, **22**(3), pp. 315-330, 1998.