Robotica (2013) volume 31, pp. 323–330. © Cambridge University Press 2012
doi:10.1017/S0263574712000331

Reactive and the shortest path navigation of a wheeled mobile robot in cluttered environments

Andrey V. Savkin and Michael Hoy*

School of Electrical Engineering and Telecommunication, University of New South Wales, Sydney, Australia

(Accepted June 8, 2012. First published online: July 10, 2012)

SUMMARY

We determine the shortest (minimal in length) path on a unicycle-like mobile robot in a known environment with smooth (possibly non-convex) obstacles with a constraint on curvature of their boundaries. Furthermore, we propose a new reactive randomized algorithm of robot navigation in unknown environment and prove that the robot will avoid collisions and reach a steady target with probability 1. The performance of our algorithm is confirmed by computer simulations and outdoor experiments with a Pioneer P3-DX mobile wheeled robot.

KEYWORDS: Obstacle avoidance; Robot navigation; Cluttered environments; Wheeled mobile robots; Minimum length path; Shortest path planning; Reactive navigation; Global path planning.

1. Introduction

Collision-free navigation of a mobile robot in cluttered environments is a fundamental problem of robotics. In order to operate in a cluttered environment, an autonomous unmanned vehicle should be able to detect and avoid the obstacles along the trajectory toward the target. Existing navigation approaches can be generally classified as global or local (reactive).[7] Global path planning algorithms use *a priori* information to build a complete model of the environment and then try to find the best possible solution.[1,4,5,8,15,19] In this approach the environment is assumed to be known *a priori*. On the other hand, local or reactive navigation algorithms use on board sensors to locally observe small fragments of an unknown environment at each time.[2,7,11,17] However, one problem is that these strategies are all heuristic and are not based on mathematical models such as kinematic equations of the vehicles and their non-holonomic constraints.

In this paper, we first consider the problem of global shortest path planning in a known environment. We assume that there is a number of possibly non-convex obstacles with a constrain on their boundaries curvature and a steady target that should be reached by the robot. We prove that the shortest (minimal in length) path consists of edges of the so-called tangent graph introduced in this paper. Therefore, the problem of the shortest path planning is reduced to a finite search problem. Furthermore, we consider also a problem of on-line reactive navigation in unknown environment. We propose a randomized navigation algorithm and prove that the robot will reach the target with probability 1 avoiding collision with obstacles. We assume that the robot studied in this paper is a unicycle like vehicle described by the standard non-holonomic model with a hard constraint on angular velocity. It is well known that the motion of many wheeled robots, missiles and unmanned aerial vehicles can be described by this model; see for example refs. [9–14, 17, 18]. Unlike many other papers on this area of robotics that present heuristic-based navigation strategies, we give mathematically rigorous analysis of our navigation algorithms with complete proofs of the stated theorems. Moreover, it should be pointed out that many other researchers' papers on this topic (see e.g. refs. [4, 5, 8, 15]) do not assume non-holonomic constraints on robot's motion, which is a severe limitation in practice. We confirm the performance of our real-time navigation strategy with extensive computer simulations and outdoor experiments with a Pioneer P3-DX mobile wheeled robot. The reminder of the paper is organized as follows. Section 2 presents the system description and the assumptions. Section 3 considers the problem of the shortest path planning. The problem of reactive navigation in unknown environments is introduced and solved in Section 4. Computer simulations for the proposed reactive navigation algorithm are given in Section 5, whereas Section 6 presents experiments with a Pioneer P3-DX mobile wheeled robot. Section 7 presents brief conclusions. Finally, the proofs of the main results are given in Appendix.

2. System Description and Main Assumptions

We consider a planar vehicle or wheeled mobile robot modeled as a unicycle. It travels with a constant speed and is controlled by the angular velocity limited by a given constant. The model of the vehicle is as follows (see Fig. 1):

$$\begin{array}{ll} \dot{x} = v\cos\theta & x(0) = x_0 \\ \dot{y} = v\sin\theta \quad , & y(0) = y_0\,. \\ \dot{\theta} = u \in [-u_M, u_M] & \theta(0) = \theta_0 \end{array} \tag{1}$$

Here (x, y) is the vector of the vehicle's Cartesian coordinates, θ gives its orientation, v and u are the speed and angular velocity, respectively. The maximal angular velocity u_M is given. The robot satisfies the standard non-holonomic

* Corresponding author. E-mail: mch.hoy@gmail.com

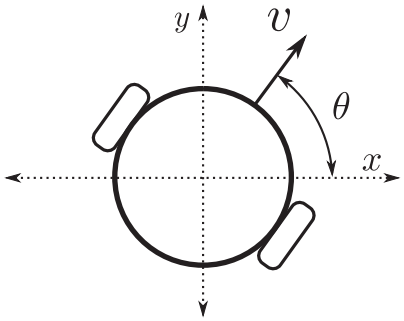

Fig. 1. Unicycle model of a wheeled robot.

constraint

$$|u(t)| \le u_M, \tag{2}$$

and the minimum turning radius of the robot is

$$R_{\min} = \frac{v}{u_M}. \tag{3}$$

Any path $(x(t), y(t))$ of the robot as shown in (1) is a plane curve satisfying the following constraint on its so-called average curvature (see ref. [3]): let $P(s)$ be this path parametrized by arc length, then

$$\|P'(s_1) - P'(s_2)\| \le \frac{1}{R_{\min}}|s_1 - s_2|. \tag{4}$$

Here $\|\cdot\|$ denotes the standard Euclidean vector norm. Note that we use the constraint (4) on average curvature because we cannot use the standard definition of curvature from differential geometry (see e.g. ref. [16]) since the curvature may not exist at some points of the robot's trajectory.

There is a steady point-wise target $\mathcal{T}$ and several disjoint obstacles $D_1, \ldots, D_k$ in the plane. Let the safety margin $d_0 > 0$ be given. The objective is to drive the vehicle to the target through the obstacle-free part of the plane while keeping the safety margin.

Let D be a closed set, p be a point in the plain. Introduce the distance $\rho(D, p)$ as

$$\rho(D, p) := \min_{q \in D} \|p - q\|.$$

Note that min is achieved since D is closed. Also, $\rho(D, p) = 0$ if $p \in D$.

Definition 2.1. *For $d_0 > 0$, the d_0-*neighborhood *of the domain $D \subset \mathbf{R}^2$ is the set formed by all points at the distance $\le d_0$ from D, i.e., $\mathcal{N}[D, d_0] := \left\{p \in \mathbf{R}^2 : \rho(D, p) \le d_0\right\}$.*

Definition 2.2. *A path $p(t) = (x(t), y(t))$ of the robot as shown in (1) is said to be* target reaching with obstacle avoidance *if there exists a time $t_f > 0$ such that $p(t_f) = \mathcal{T}$ and $p(t)$ does not belong to $\mathcal{N}[D_i, d_0]$ for all $i, t \in [0, t_f]$.*

Assumption 2.1. *For all i the set $\mathcal{N}[D_i, d_0]$ is a closed, bounded, connected and linearly connected set.*

Assumption 2.2. *The sets $\mathcal{N}[D_i, d_0]$ and $\mathcal{N}[D_j, d_0]$ do not overlap for any $i \ne j$.*

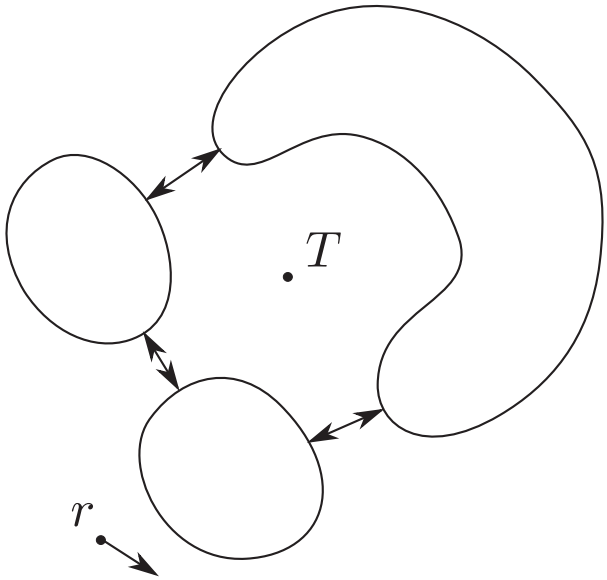

Fig. 2. The target T cannot be safely reached if the minimum distance between obstacles is too small.

Remark 2.1. *It is obvious that if Assumption 2.2 does not hold, target reaching with obstacle avoidance may be impossible (see e.g. Fig. 2).*

Assumption 2.3. *For all i the boundary ∂D_i of the obstacle D_i is a closed, non-self-intersecting analytic curve.*

Assumption 2.4. *For all i the boundary $\partial D_i(d_0)$ of the set $\mathcal{N}[D_i, d_0]$ is a closed, non-self-intersecting analytic curve with curvature $k_i(p)$ at any point p satisfying $|k_i(p)| \le \frac{1}{R_{\min}}$.*

Remark 2.2. *Note that we do not assume that the sets $\mathcal{N}[D_i, d_0]$ are convex. Therefore, the boundaries $\partial D_i(d_0)$ may have negative curvature at some points. Here we use the standard definition of curvature from differential geometry (see e.g. ref. [16]). Since $\partial D_i(d_0)$ is analytic, the curvature is defined at any point.*

We also assume that the initial position $p(0) = (x(0), y(0))$ of the robot is far enough from the obstacles and the target.

Assumption 2.5. *The following inequalities hold: $\rho(\mathcal{T}, p(0)) \ge 8R_{\min}$ and $\rho(\mathcal{N}[D_i, d_0], p(0)) \ge 8R_{\min}$ for all i.*

Definition 2.3. *There are two circles with the radius $R_{\min}$ that cross the initial robot position $p(0)$ and tangent to the robot initial heading $\theta(0)$. We will call them initial circles.*

Assumption 2.6. *The robot initial heading $\theta(0)$ is not tangent to any boundary $\partial D_i(d_0)$.*

3. Off-Line Shortest Path Planning

In this section we describe the shortest or minimum length target-reaching paths with obstacle avoidance.

Definition 3.1. *A straight line L is said to be a tangent line if one of the following conditions holds:*

1. *The line L is simultaneously tangent to two boundaries $\partial D_i(d_0)$ and $\partial D_j(d_0)$ where $i \ne j$.*

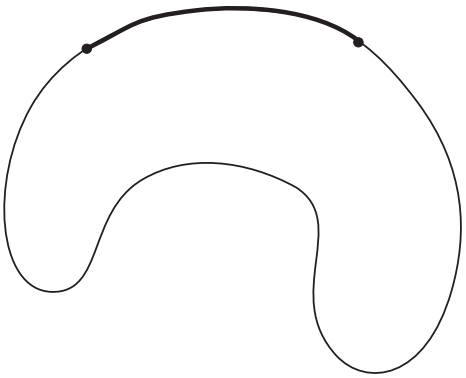

Fig. 3. A (B)-segment.

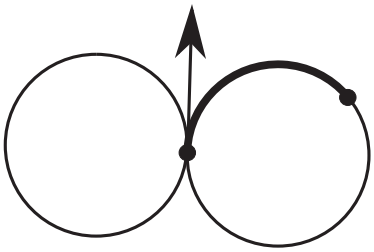

Fig. 4. A (C)-segment.

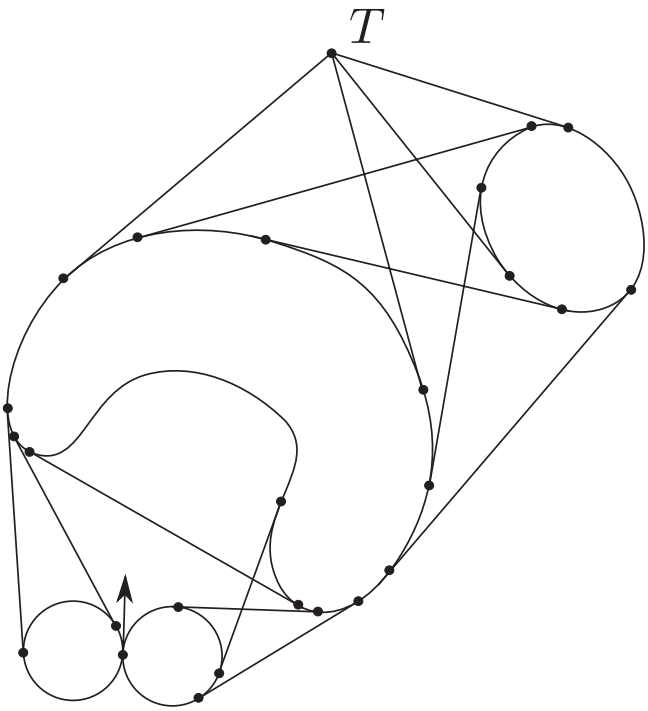

Fig. 5. An example of the tangent graph, with a target point T.

2. *The line L is tangent to a boundary $\partial D_i(d_0)$ on two different points.*
3. *The line L is simultaneously tangent to a boundary $\partial D_i(d_0)$ and an initial circle.*
4. *The line L is tangent to a boundary $\partial D_i(d_0)$ and crosses the target $\mathcal{T}$.*
5. *The line L is tangent to an initial circle and crosses the target $\mathcal{T}$.*

Points of boundaries $\partial D_i(d_0)$ and initial circles belonging to tangent lines are called tangent points. Now we consider only finite segments of tangent lines such that their interiors do not overlap with any boundary $\partial D_i(d_0)$ The segments of tangent lines of types (1) *and* (2) *are called (OO)-segments, the segments of tangent lines of type* (3) *are called (CO)-segments, the segments of tangent lines of type* (4) *are called (OT)-segments and the segments of tangent lines of type* (5) *are called (CT)-segments. Furthermore, we consider only segments that do not intersect the interiors of the initial circles.*

Definition 3.2. *A segment of a boundary $\partial D_i(d_0)$ between two tangent points is called (B)-segment if curvature is non-negative at any point of this segment (see Fig. 3). A segment of an initial circle between the initial robot position $p(0)$ and a tangent point is called (C)-segment (see Fig. 4).*

Now we are in a position to present the main result of this section.

Theorem 3.1. *Suppose that Assumptions* 2.1–2.6 *hold. Then there exists the shortest (minimum length) target-reaching path with obstacle avoidance. Furthermore, the shortest target-reaching path consists of $n \geq 2$ segments $S_1, S_2, \ldots, S_n$ such that if $n = 2$ then S_1 is a (C)-segment and S_2 is a (CT)-segment. If $n \geq 3$ then S_1 is a (C)-segment, S_2 is a (CO)-segment and S_n is an (OT)-segment. If $n > 3$ and $3 \leq k \leq n-1$ then any S_k is either (OO)-segment or (B)-segment.*

The proof of Theorem 3.1 is given in Appendix.

Remark 3.1. *Consider the graph whose vertices are the robot's initial position, the target $\mathcal{T}$, and tangent points, and whose edges are segments of (OO), (CO), (OT), (CT) and (B) types. This graph has a finite number of vertices and edges since according to Assumption 2.4 all the boundaries $\partial D_i(d_0)$ are analytic. This graph is called the tangent graph. The tangent graph is defined by robot's initial position, target position and obstacles geometry. Figure 5 shows an example of a tangent graph. Theorem 3.1 reduces the problem of finding the shortest target-reaching path to a search among all paths in the tangent graph.*

4. On-Line Navigation

In this section we consider a case when the robot does not know the location of the target and the obstacles *a priori*. The robot is equipped with a vision-type sensor, which is able to determine coordinates of the target and points of the boundaries $\partial D_i(d_0)$ if the straight line segment connecting the robot current coordinates and the point of interest does not intersect any obstacle D_i (see Fig. 6).

For the case of simplicity, we will introduce the following assumption.

Assumption 4.1. *Any tangent point belongs to only one tangent line.*

Definition 4.1. *When the robot moves along the boundary of an obstacle or an initial circle and reaches a tangent point, it can leave the boundary and move along the corresponding*

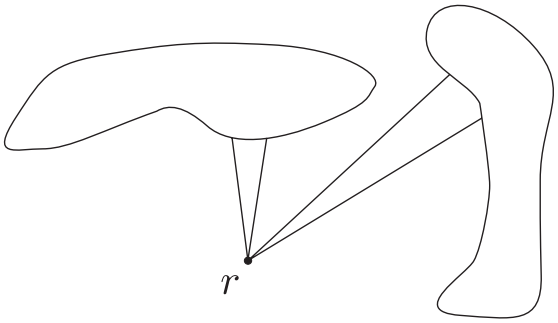

Fig. 6. The robot r is equipped with some type of vision sensor that is able to detect obstacles, tangents and the target so long as they are in line-of-sight.

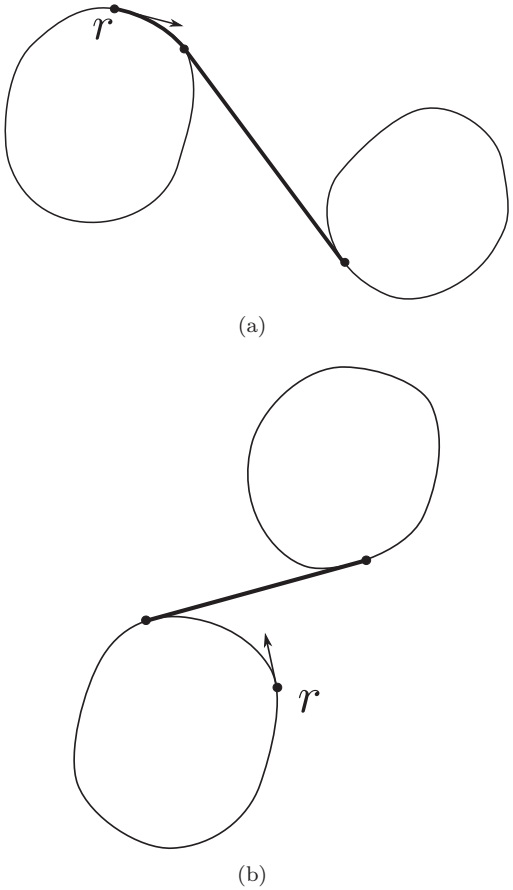

Fig. 7. A tangent segment can only be a segment of our path if it is approached by the robot r in the correct direction.

straight line edge of the tangent graph if its heading is equal to direction of this edge (see Fig. 7(a)). In this case, we say that the robot reaches an exit tangent point. (A case when the robot cannot leave the boundary at a tangent point is shown in Fig. 7(b)). Furthermore, if the straight line edge corresponding to this exit tangent point is an (OT) *or* (CT) *segment, then we call this point an exit tangent point of* T*-type. Otherwise, if the straight line edge corresponding to this exit tangent point is an* (OO) *or* (CO) *segment, we call this point an exit tangent point of* O*-type.*

Let $0 < p < 1$ be a given number. To solve this problem, we propose the following probabilistic navigation algorithm:

A1: The robot starts to move along any of two initial circles.

A2: When the robot moving along an obstacle boundary or an initial circle reaches an exit tangent point of T-type, it starts to move along the corresponding (OT) or (CT) segment.

A3: When the robot moves along an obstacle boundary or an initial circle and reaches an exit tangent point of O-type, with probability p, it starts to move along the corresponding (CO) or (OO) segment, and with probability $(1-p)$ it continues to move along the boundary.

A4: When the robot moves along a (CO) or (OO) segment and reaches a tangent point on an obstacle boundary, it starts to move along the obstacle boundary.

Now we are in a position to present the main result of this section.

Theorem 4.1. *Suppose that Assumptions* 2.1–2.6 *and 4.1 hold. Then for any* $0 < p < 1$*, the algorithm* **A1–A4** *with probability* 1 *defines a target-reaching path with obstacle avoidance.*

The proof of Theorem 4.1 is given in Appendix.

5. Computer Simulations

In this section we present computer simulations of the reactive navigation algorithm **A1–A4**. This navigation strategy was realized as a sliding mode control law by switching between the boundary following approach proposed in ref. [11] and the pure pursuit navigation approach; see e.g. ref. [14]. Our navigation law can be expressed as follows:

$$u(t) = \begin{cases} \pm u_M & R1 \\ \Gamma sgn\left[\phi_{tan}(t)\right] u_M & R2 \\ \Gamma sgn\left[\dot{d}_{\min}(t) + X(d_{\min}(t) - d_0)\right] u_M & R3 \end{cases}. \quad (5)$$

$$\begin{aligned} &R1 \rightarrow R2 : CO \text{ or } CT \text{ detected}, \\ &R2 \rightarrow R3 : d_{\min}(t) < d_{\text{trig}}, \dot{d}_{\min}(t) < 0, \\ &R3 \rightarrow R2 : \begin{cases} OT \text{ detected} \\ OO \text{ detected, probability } p \end{cases}. \end{aligned} \quad (6)$$

This navigation law is a rule for switching between three separate modes $R1 - R3$. Initially, mode $R1$ is active, and transitions to other modes are determined by Eq. (6). Mode $R1$ describes motion along the initial circle with maximal actuation. Mode $R2$ describes pursuit navigation, where $\phi_{tan}(t)$ is defined as the angle between the vehicle's heading and a line segment connecting the vehicle and currently tracked tangent edge (see Eq. (8)). Mode $R3$ describes boundary following behavior, where the control calculation is based on the minimum distance to the nearest obstacle, defined as $d_{\min}(t)$. This control law is subject to some restrictions, which are inherited from ref. [11]; however, these are satisfied due to the assumptions in Section 2. The variable Γ is defined as $+1$ if the obstacle targeted is on the left of the tangent being tracked, and -1 if it is on the right.

A constant $d_{\text{trig}} > d_0$ is also introduced, which determines when the control system transitions to boundary following mode.[11] The saturation function X is defined as follows:

$$X(r) = \begin{cases} lr & |r| < k \\ lk\ sgn(r) & otherwise \end{cases}, \tag{7}$$

where l and k are tunable constants. Because of any potential chattering in the output, once it is decided to not pursue a tangent, there is a short pause until tangent following can potentially be engaged again.

Remark 5.1. *Note that it follows from (1) that the robot's trajectory is differentiable. This and Assumption 2.3 imply that the function $d_{min}(t)$ is continuous; however, in the case on a non-convex obstacle, the function $d_{min}(t)$ may be non-differentiable for some t. It does not really matter in practice, in our computer simulations and experiments with a real robot, we use a first-order difference approximation of $\dot{d}_{min}(t)$ in Eq. (5).*

In our simulations and experiments we assume that we have a LIDAR-type device which informs the vehicle of the distance from the vehicle to the obstacle in a finite number of directions around the obstacle. The length of these detection rays are defined by η_i, where η_0 refers to a detection ray directly in front of the vehicle. The angular spacing between successive rays is defined by $\Delta\theta$. In this case a tangent in front of the vehicle is detected by monitoring $|\Delta\eta_0|$ for any changes beyond some threshold d_{thresh}. The transversal direction can be determined by comparing the immediately adjacent obstacle detections, $\Gamma := \text{sign}(\eta_1 - \eta_{-1})$.

To calculate the error parameter ϕ_{tan}, a point T_{int} is defined as an intermediate target. This is calculated by iteration of the following steps:

1. Initially when $R2$ is engaged, the T_{int} is set to be a constant offset from the detected tangent point:

$$T_{\text{int}} = \begin{bmatrix} x(t) + cos(\theta(t) + i\Delta\theta + \Gamma \tan^{-1}(d_0/d_{\text{cen}}(t))) \\ y(t) + sin(\theta(t) + i\Delta\theta + \Gamma \tan^{-1}(d_{\text{tar}}/d_{\text{cen}}(t))) \end{bmatrix} \tag{8}$$

with $i := 0$.

2. ϕ_{tan} is calculated by finding the angle between the vehicle heading and a line connecting the vehicle and T_{int}.

3. In subsequent time steps we find a successive tangent point; a search occurs for appropriate i so that $\eta_i - \eta_{i+\Gamma} > d_{\text{thresh}}$ (so we maintain the same transversal direction as the stored intermediate target), and also has the smallest Euclidean distance to T_{int}. The point is calculated using Eq. (8).

Remark 5.2. *While this calculation calls for an estimate of position to be available to the robot, this estimate only needs to be accurate for a relatively short time between control updates. Thus, in the studied case, robot odometry is sufficient since robot odometry gives an accurate estimate over short time intervals.*

The following simulation was carried out with the unicycle model (1) and the navigation law updated at 10 Hz. The control parameters used for simulations may be found in Table I. In Figs. 8–10, it can be seen that the robot converges to the target without any problems, as expected. Different sequences of random numbers would of course lead to different paths around the obstacles.

Table I. Control parameters used for simulations.

$u_{\max}$	1.3 rads^{-1}
v	1.5 ms^{-1}
d_{tar}	5 m
d_{trig}	10 m
l	0.33
k	15 m
p	0.7
d_{thresh}	10 m

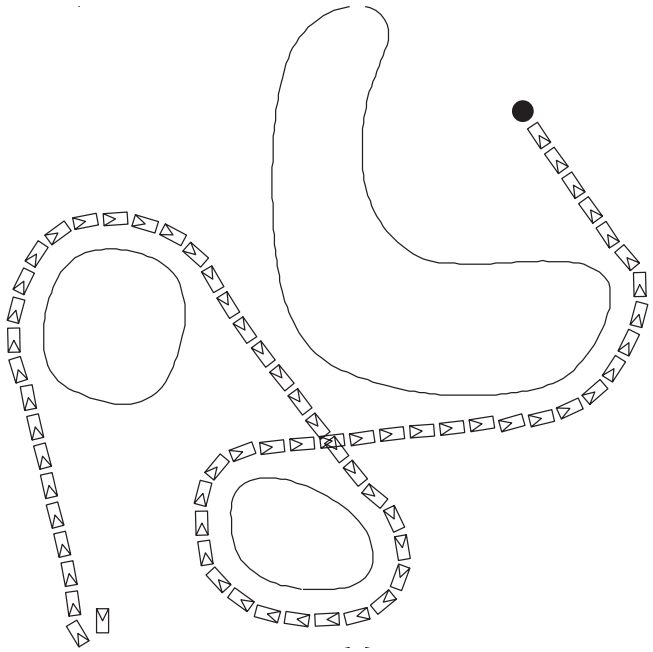

Fig. 8. Simulation with a simple environment.

6. Experiments with a Real Robot

In this section we present an implementation of the reactive navigation algorithm **A1–A4**. Experiments were carried out with a Pioneer P3-DX mobile robot. A LIDAR device with an angular resolution of 0.5° was used to detect tangents directly ahead of the robot as well as obstacles in the vicinity of the robot. In the scenario tested we did not provide the vehicle with a target, rather it was allowed to patrol the area indefinitely. Odometry information available from the robot was used over a single time step to compensate for the movement of the previously calculated tangent point relative to the vehicle, as explained in the previous section (see also Remark 5.1). Range readings over a maximum threshold $R_{\max}$ were truncated in order to prevent any object outside our test area from influencing the results.

Measures were also included to reduce control chattering, which can cause detrimental effects when using real robots.[17] The signum function of Eq. (5) was replaced by a saturation function (Eq. (7) with $l = k = 1$), and the standard low-level

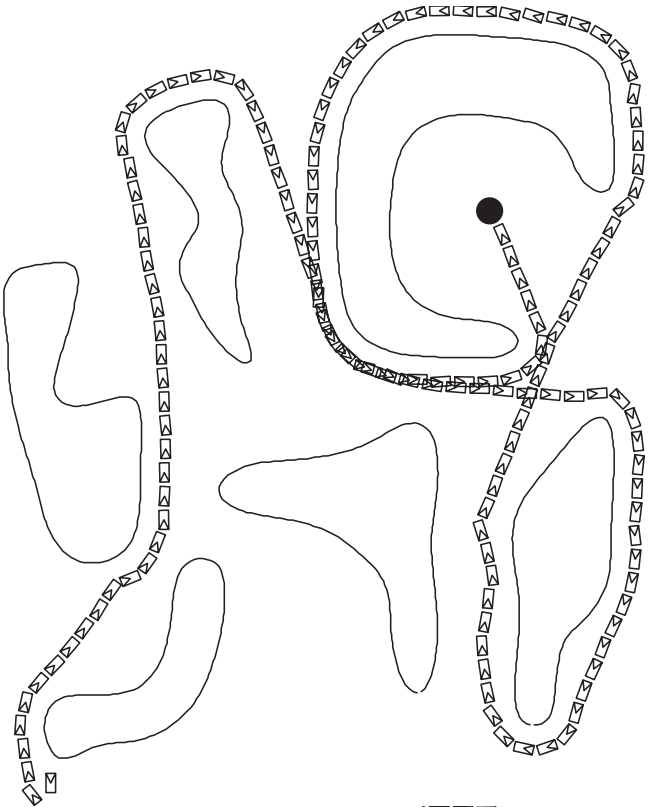

Fig. 9. Simulation with a challenging environment.

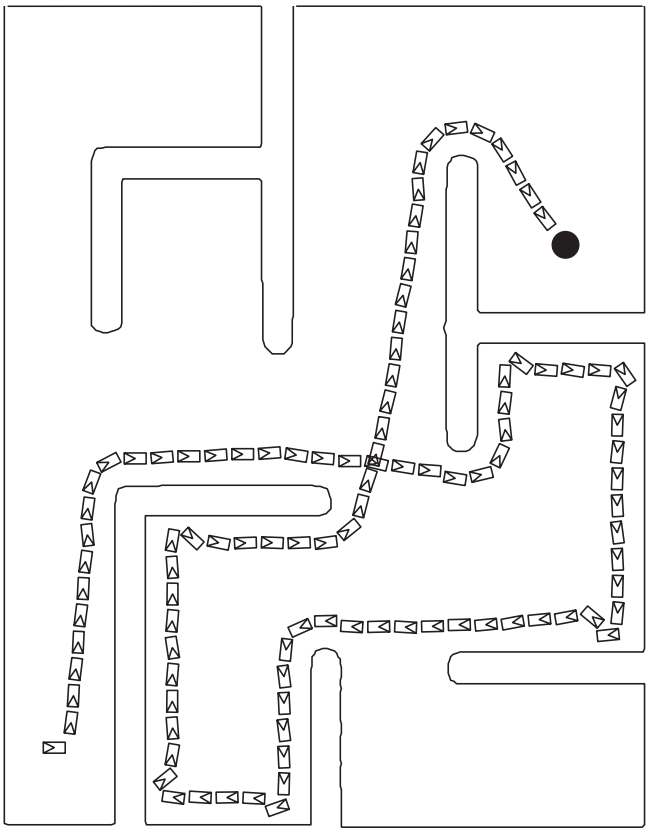

Fig. 10. Simulation with a more challenging environment.

Table II. Control parameters used for experiments with a real robot.

v	0.25 ms^{-1}
d_{tar}	1 m
d_{trig}	1.5 m
l	0.1
k	1.0 m
p	0.7
$R_{\max}$	6 m
d_{thresh}	1 m

Fig. 11. The pioneer P3-DX mobile robot used for experiments.

controller on the robot was modified to filter high-frequency control inputs. The control parameters used for experiments may be found in Table II. Note the turning rate requested was much larger than what was actually obtained.

The vehicle successfully navigates around the obstacles without collision, as indicated in Fig. 12. A video of the navigated robot in action is available at http://www2.ee.unsw.edu.au/temp/tangent_follow.wmv.

Remark 6.1. *Note that in our computer simulations and experiments with a real robot, we implemented the proposed reactive navigation algorithm* **A1–A4**. *Theorem 4.1 implies that this navigation law with probability* 1 *results in a target-reaching path with obstacle avoidance. However, this path is not the shortest (minimal in length) path.*

7. Conclusions

We have described the shortest (minimal in length) path on a unicycle-like mobile robot in a known environment with smooth (possibly non-convex) obstacles with a constraint on curvature. Furthermore, we have proposed a reactive randomized algorithm of robot navigation in unknown environment and proved that the robot will avoid collisions and reach a steady target with probability 1. The performance of our algorithm has been confirmed by computer simulations and outdoor experiments with a Pioneer P3-DX mobile wheeled robot.

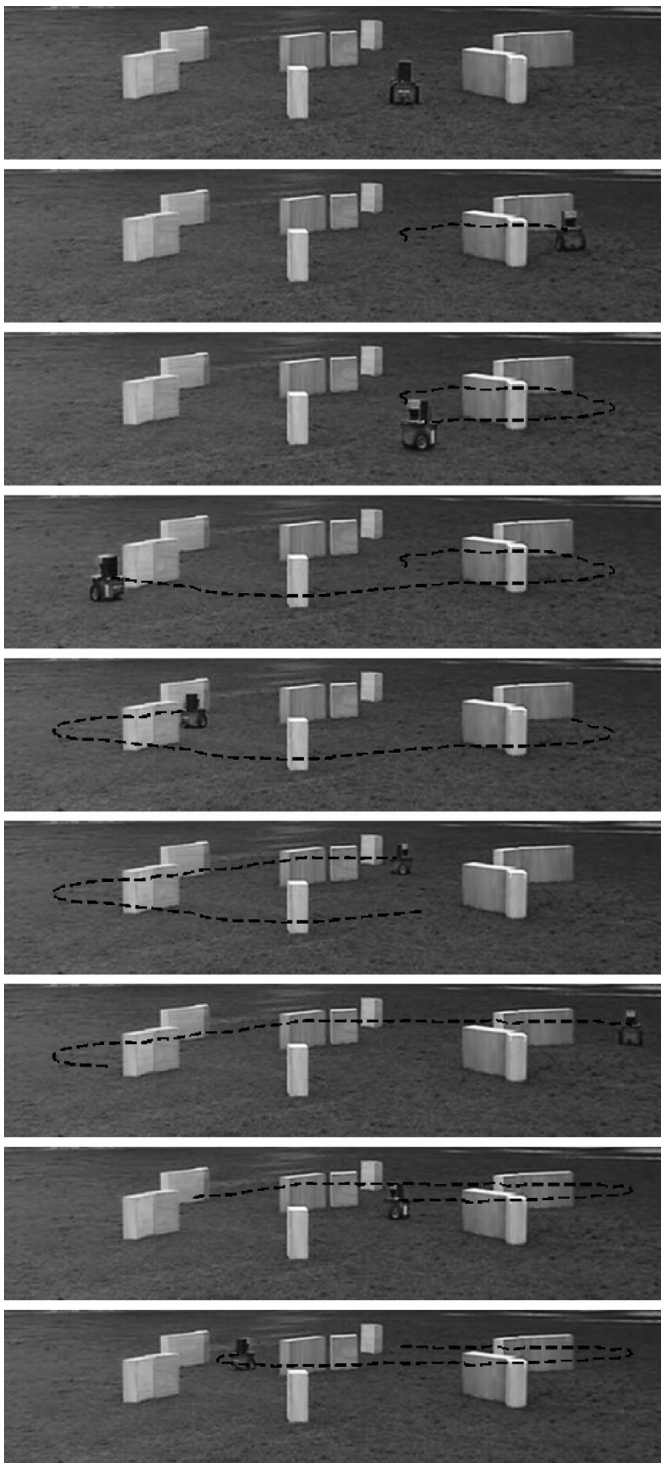

Fig. 12. Sequence of images showing the tangent following controller in action. This was done without a target so that the robot patrols the area indefinitely.

References

1. S. Belkhous, A. Azzouz, M. Saad, V. Nerguizian and C. Nerguizian, "A novel approach for mobile robot navigation with dynamic obstacle avoidance," *J. Intell. Robot. Syst.* **44**(3), 187–201 (2005).
2. M. Deng, A. Inoue, Y. Shibata, K. Sekiguchi and N. Ueki, "An Obstacle Avoidance Method for Two Wheeled Mobile Robot," **In**: *Proceedings of the IEEE International Conference on Networking, Sensing and Control*, London, UK (Apr. 15–17, 2007) pp. 689–692.
3. L. E. Dubins, "On curves of minimal length with a constraint on average curvature and with prescribed initial and terminal positions and tangents," *Am. J. Math.* **79**, 497–516 (1957).
4. I. Kamon, E. Rimon and E. Rivlin, "Tangentbug: A range-sensor-based navigation algorithm," *Int. J. Robot. Res.* **17**(9), 934–953 (1998).
5. I. Kamon and E. Rivlin, "Sensory-based motion planning with global proofs," *IEEE Trans. Robot. Autom.* **13**(6), 817–822 (1997).
6. A. N. Kolmogorov and S. V. Fomin, *Elements of the Theory of Functions and Functional Analysis* (Courier Dover, N. Chemsford, MA, 1999).
7. L. Lapierre, R. Zapata and P. Lepinay, "Combined path following and obstacle avoidance control of a wheeled robot," *Int. J. Robot. Res.* **26**(4), 361–375 (2007).
8. Y. H. Liu and S. Arimoto, "Path planning using a tangent graph for mobile robots among polygonal and curved obtacles," *Int. J. Robot. Res.* **11**(4), 376–382 (1992).
9. I. R. Manchester and A. V. Savkin, "Circular navigation missile guidance with incomplete information and uncertain autopilot model," *J. Guid. Control Dyn.* **27**(6), 1078–1083 (2004).
10. I. R. Manchester and A. V. Savkin, "Circular navigation guidance law for precision missile target engagement," *J. Guid. Control Dyn.* **29**(2), 314–320 (2006).
11. A. S. Matveev, H. Teimoori and A. V. Savkin, "A method for guidance and control of an autonomous vehicle in problems of border patrolling and obstacle avoidance," *Automatica* **47**(3), 515–524 (2011).
12. A. S. Matveev, H. Teimoori and A. V. Savkin, "Navigation of a unicycle-like mobile robot for environmental extremum seeking," *Automatica* **47**(1), 85–91 (2011).
13. A. S. Matveev, H. Teimoori and A. V. Savkin, "Range-only measurements based target following for wheeled mobile robots," *Automatica* **47**(1), 177–184 (2011).
14. A. V. Savkin and H. Teimoori, "Bearings-only guidance of a unicycle-like vehicle following a moving target with a smaller minimum turning radius," *IEEE Trans. Autom. Control* **55**(10), 2390–2395 (2010).
15. Z. Shiller, "Online suboptimal obstacle avoidance," *Int. J. Robot. Res.* **19**(5), 480–497 (2000).
16. D. J. Struik, *Lectures on Classical Differential Geometry* (Courier Dover, N. Chemsford, MA, 1988).
17. H. Teimoori and A. V. Savkin, "A biologically inspired method for robot navigation in a cluttered environment," *Robotica* **28**(5), 637–648 (2010).
18. H. Teimoori and A. V. Savkin, "Equiangular navigation and guidance of a wheeled mobile robot based on range-only measurements," *Robot. Auton. Syst.* **58**(2), 203–215 (2010).
19. N. Vlassis, N. Sgouros, G. Efthivolidis and G. Papakonstantinou, "Global Path Planning for Autonomous Navigation," **In**: *Proceedings of the 8th IEEE International Conference on Tools with AI*, Toulouse, France (Nov. 16–19, 1996) pp. 354–359.

Appendix

The proof of Theorem 3.1. First, we prove that there exists the shortest target-reaching path. As in ref. [3], it easily follows from Ascoli's Theorem (see e.g. ref. [6]).

Now let P be the shortest (minimum length) target-reaching path. We now prove that the path P does not go inside any of the two initial circles. Indeed, consider the circle C of radius $6R_{\min}$ centered at $p(0)$. It follows from Assumption 2.5 that the target and all the obstacles are outside this circle. Let p_1 be the point of the circle at which the path P leaves the circle for last time, and θ_1 be the trajectory heading at p_1. It means that P consists of two segments $(p(0), p_1)$ and $(p_1, \mathcal{T})$ where $(p_1, \mathcal{T})$ is outside the circle C. Now let P_1 be a path of minimum length connecting the points

$p(0)$ and p_1 such that it has headings $\theta(0)$ and θ_1 at these points, respectively, and the average curvature condition (4) is satisfied. It follows from the main result of ref. [3] that such a path exists and P_1 belongs to the disk of radius $8R_{\min}$ centered at $p(0)$. Therefore, Assumption 2.5 implies that P_1 does not intersect any $\partial D_i(d_0)$. Since P_1 is a minimum length path, its length is less or equal to the length of $(p(0), p_1)$. On the other hand, the length of $(p(0), p_1)$ cannot be greater than the length of P_1 (if it is not true, then the path consisting of P_1 and $(p_1, \mathcal{T})$ would be a shorter target-reaching path than P, which contradicts our assumption that P is the shortest path). Hence, $(p(0), p_1)$ is the shortest path satisfying condition (4). Therefore, it follows from the main result of ref. [3] that $(p(0), p_1)$ consists of segments of two or less minimum radius circles and a straight line segment, and $(p(0), p_1)$ does not cross the interiors of both initial circles. Hence, we have proved that the path P does not go inside any of two initial circles.

Now let P_2 be the shortest path connecting the points $p(0)$ and $\mathcal{T}$ which is not crossing the interiors of the sets $\mathcal{N}[D_i, d_0]$ and two initial circles. Now we do not assume that P_2 satisfies the average curvature constraint (4). Since we have proved that the path P does not go inside any of two initial circles, the length of P_2 is less or equal to the length of P. Now we prove that P_2 consists of segments of the boundaries of the sets $\mathcal{N}[D_i, d_0]$ and two initial circles, and straight line segments connecting two different points on these boundaries. Indeed, we prove it by contradiction. If this statement does not hold, then there exists a point $p_2 \in P_2$, which is outside the sets $\mathcal{N}[D_i, d_0]$ and two initial circles, and p_2 is not an interior point of a straight line segment of P_2. Therefore, in some small neighborhood of p_2 we can replace the segment of path P_2 containing p_2 by a straight line segment which does not intersect the sets $\mathcal{N}[D_i, d_0]$ and two initial circles (see Fig. 13). Hence, we can construct a path with shorter length than P_2 satisfying all the requirements, as demonstrated in Fig. 13. Furthermore, it follows from Theorem 5 of ref. [8] that any straight line segment of P_2 connects two tangent points, and it also follows from Theorem 4 of ref. [8] that any segment of $\partial D_i(d_0)$ belonging to P_2 has non-negative curvature. Therefore, the path P_2 may consist of segments of two initial circles, segments of $\partial D_i(d_0)$ with non-negative curvature and segments of straight lines connecting tangent points or a tangent point and the target. Also, Assumption 2.6 implies that the first segment of P_2 is a segment of one of the initial circles, not a straight line segment. Furthermore, since all boundary segments of P_2 have curvature magnitude less or equal to $\frac{1}{R_{\min}}$, and all straight line segments of P_2 connect tangent points (or a tangent point and the target), the path P_2 satisfies (4). This completes the proof of Theorem 3.1. □

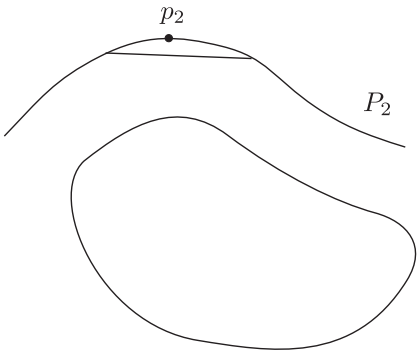

Fig. 13. If a segment of a path outside obstacles and initial circles is not a straight line segment, then there exists a shorter path.

The proof of Theorem 4.1. The algorithm **A1–A4** defines the absorbing Markov chain in which the states are the vertices of the tangent graph. It contains one absorbing state (that is impossible to leave), which is the vertex corresponding to the target $\mathcal{T}$. Moreover, it is obvious that this absorbing state can be reached with non-zero probability from any other state of the chain. This implies that with probability 1, the absorbing state will be reached. This completes the proof of Theorem 4.1.