Robotica (2013) volume 31, pp. 993–1001. © Cambridge University Press 2013
doi:10.1017/S0263574713000313

A simple biologically inspired algorithm for collision-free navigation of a unicycle-like robot in dynamic environments with moving obstacles

Andrey V. Savkin and Chao Wang*

School of Electrical Engineering and Telecommunications, The University of New South Wales, Sydney 2052, Australia

(Accepted March 28, 2013. First published online: May 1, 2013)

SUMMARY
We present a simple biologically inspired strategy for the navigation of a unicycle-like robot towards a target while avoiding collisions with moving obstacles. A mathematically rigorous analysis of the proposed approach is provided. The performance of the algorithm is demonstrated via experiments with a real robot and extensive computer simulations.

KEYWORDS: Moving obstacles; Biologically inspired guidance; Collision avoidance; Obstacle avoidance; Wheeled mobile robots; Biomimetic navigation; Navigation in unknown dynamic environments.

1. Introduction

Avoidance of collisions with obstacles is a key component of safe navigation. Therefore, a 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 towards the target. A typical objective is to reach a target through the obstacle-free part of the environment.[1] This may involve bypassing a nearby obstacle with a safety margin.[2] This problem has been the subject of extensive research in robotics.

Existing navigation approaches can be generally classified as global or local (reactive).[3] 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.[4–9] 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.[3,10–12]

However, one problem is that these strategies are heuristic and they are not based on mathematical models such as kinematic equations of the vehicles and their non-holonomic constraints. Many of the proposed obstacle avoidance techniques, such as the dynamic window,[13,14] the curvature velocity,[15] and the lane curvature[16] approaches consider the case of stationary obstacles. The problem of avoiding collisions with moving obstacles is much harder. For this problem, a number of approaches like velocity obstacles,[17,18] collision cones,[19] or inevitable collision states[20,21] have been proposed. These approaches assume a deterministic knowledge about the obstacle velocity and a moderate rate of its change. Furthermore, these methods tend to be computationally expensive. Moreover, these and many other papers on this topic (see e.g. refs. [4–7]) do not take into account non-holonomic constraints on robot's motion which is a severe limitation in practice.

Examples concerning non-holonomic robots include artificial potential approach, combined with sliding-mode control for gradient climbing,[22,23] and kinematic control based on polar coordinates and Lyapunov-like analysis.[24] To date, fully actuated robots were mostly studied in this area, the obstacles were interpreted as rigid bodies of the simplest shapes (e.g. discs[23,24] or polygons[22,25]), the sensory data were assumed to be enough to determine the location of obstacle characteristic points concerned with its global geometry (e.g. the disc centre[23,24] or angularly most distant polygon vertex[25]) and to provide access to its full velocity.[23–25] Furthermore, a mathematically rigorous performance justification of the proposed algorithms is rarely given.

A local range-only-based navigation algorithm for non-holonomic robots was proposed in ref. [26]. In ref. [26], the obstacles are not assumed to be rigid or even solid. They have arbitrary and time-varying shapes that may rotate, twist, wring, skew, wriggle, or be deformed in any other way. However, the main results of ref. [26] require quite restrictive assumptions on velocities of the moving obstacles. New algorithms for navigation in dynamic environments were proposed in refs. [27, 28]. However, the algorithms of refs. [27, 28] do not take into account non-holonomic constraints. Moreover, the papers [27, 28] do not give any mathematically rigorous analysis of the proposed navigation algorithms.

In this paper, we propose a novel biologically inspired reactive algorithm for navigation towards a target avoiding collisions with moving obstacles. We assume that the mobile robot is a unicycle-like vehicle described by the standard non-holonomic model with a hard constraint on its 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 e.g. refs. [11, 12, 29, 30] and references therein. Unlike many other papers

* Corresponding author. E-mail: z3184703@zmail.unsw.edu.au

https://doi.org/10.1017/S0263574713000313 Published online by Cambridge University Press

on this area of robotics which present heuristic navigation strategies, we give a mathematically rigorous analysis of our navigation algorithm with a complete proof of the stated theorem. Moreover, the proposed algorithm is quite simple and computationally efficient. Our navigation strategy is based on switching between moving to the target along straight lines, when possible, and a sliding-mode obstacle avoidance navigation law.

Researchers in the area of robot navigation are finding much inspiration from biology, where the problem of controlled animal motion is a central one. Animals, such as insects, birds, or mammals, are believed to use simple, local motion control rules that result in remarkable and complex intelligent behaviours. The algorithm proposed in this paper also belongs to the class of biologically inspired or biomimetic navigation algorithms. In biology, a similar obstacle avoidance strategy is called negotiating obstacles with constant curvatures (see e.g. ref. [31]). An example of such a movement is a squirrel running around a tree. A similar strategy for avoiding stationary obstacles was proposed and studied in ref. [29]. In the case of stationary obstacles, the strategy proposed in this paper is reduced to the algorithm of ref. [29]. We confirm the performance of our real-time navigation strategy with extensive computer simulations and experiments with a Pioneer P3-DX mobile wheeled robot. In particular, the performance of the proposed method has been compared with that of the velocity obstacle approach.[17,18]

The remainder of the paper is organized as follows. Section 2 presents the system description and the assumptions. In Section 3, our navigation algorithm is introduced. A mathematical analysis of the algorithm is given 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. Finally, Section 7 presents brief conclusions.

2. Problem Description

We consider a planar vehicle or wheeled mobile robot modelled as a unicycle. The control inputs are angular velocity and speed, both limited by given constants. The mathematical model of the vehicle is as follows:

$$\begin{aligned} \dot{x} &= V(t)\cos\theta, \quad & x(0) &= x_0, \\ \dot{y} &= V(t)\sin\theta, \quad & y(0) &= y_0, \\ \dot{\theta} &= u, & \theta(0) &= \theta_0, \end{aligned} \tag{1}$$

where

$$u \in [-U_{\max}, U_{\max}], \quad V(t) \in [0, V_{\max}]. \tag{2}$$

Here, $[x(t), y(t)]$ is the vector of the vehicle's Cartesian coordinates, $\theta(t)$ gives its orientation, $v(t)$ and $u(t)$ are the speed and angular velocity, respectively. The angle $\theta(t)$ is measured in the counterclockwise direction. The maximum angular velocity $U_{\max}$ and the maximum speed $V_{\max}$ are given. Also, let

$$c_R(t) := [x(t), y(t)] \tag{3}$$

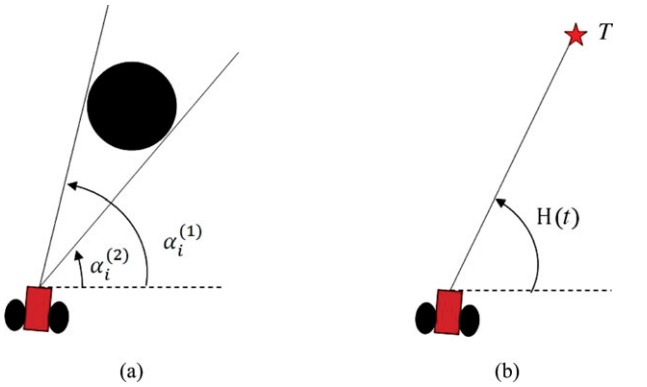


Fig. 1. (Colour online) (a) Vision cone to the obstacle i; (b) heading $H(t)$ to the target.

be the vector of the robot's coordinates, and

$$v_R : (t) = \dot{c}_R(t) \tag{4}$$

be the robot's velocity vector.

We consider a quite general problem of vehicle navigation with collision avoidance. We consider a stationary point-wise target **T** and several disjoint moving obstacles $D_1(t), \ldots, D_k(t)$ in the plane. The objective is to drive the vehicle to the target while avoiding collisions with the moving obstacles $D_1(t), \ldots, D_k(t)$. We assume that any obstacle i is a moving closed bounded planar set. The vehicle has access to the current distance $d_i(t)$ to the border of the obstacle i defined as

$$d_i(t) := \min_{r \in D_i(t)} \|r - c_R(t)\|.$$

Here, $\|\cdot\|$ denotes the standard Euclidean vector norm. Note that min is achieved since $D_i(t)$ is closed. Furthermore, the vehicle has access to $\dot{d}_i(t)$. Moreover, the mass centre of the obstacle i is moving with the velocity $v_i(t)$ which is available to the robot. Furthermore, velocities v_i satisfy the following constraint:

$$\|v_i(t)\| \leq V < V_{\max} \tag{5}$$

for some $V > 0$ and all i, t. It is obvious that if (5) does not hold, the collision avoidance may be impossible. Also, the robot measures the angles $\alpha_i^{(1)}(t)$ and $\alpha_i^{(2)}(t)$ which are the boundaries of the vision cone of the obstacle i from the robot; see Fig. 1(a). Finally, the robot knows the heading $H(t)$ to the target **T**; see Fig. 1(b).

Definition 2.1. Let $d_{safe} > 0$ be a given constant. A robot navigation strategy is said to be *a target reaching strategy with collision avoidance* if there exists a time $t_f > 0$ such that $c_R(t_f) = \boldsymbol{T}$ and $d_i(t) \geq d_{\text{safe}}$ $\forall i = 1, 2, \ldots, k, \quad \forall t \in [0, t_f]$.

3. Navigation Algorithm

To avoid a collision of the robot with the moving obstacle i, we will use the following manoeuvre. Let α_0 be a given

https://doi.org/10.1017/S0263574713000313 Published online by Cambridge University Press

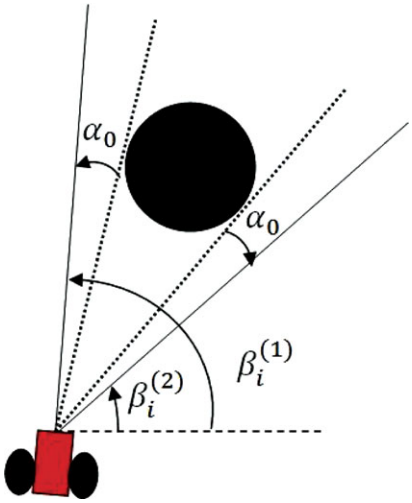


Fig. 2. (Colour online) Enlarged vision cone.

angle, $0 < \alpha_0 < \frac{\pi}{2}$. Let

$$\beta_i^{(1)}(t) := \alpha_i^{(1)}(t) - \alpha_0, \quad \beta_i^{(2)}(t) := \alpha_i^{(2)}(t) + \alpha_0; \quad (6)$$

see Fig. 2. Furthermore, introduce the vectors $l_i^{(1)}$ and $l_i^{(2)}$ as

$$l_i^{(j)}(t) := (V_{\max} - V)\left[\cos(\beta_i^{(j)}t)), \sin(\beta_i^{(j)}(t))\right] \quad \forall j = 1, 2, \quad (7)$$

where $V_{\max}$ and V are the constants from (2) and (5). A geometric interpretation of (7) is straightforward. Equations (7) define the two lines which are the boundaries of an enlarged vision cone of the obstacle. These lines are also known as occlusion lines.

Let l_1 and l_2 be non-zero two-dimensional vectors, and let $\alpha(l_1, l_2)$ denote the angle between the vectors l_1 and l_2 measured from l_1 in the counterclockwise direction, $\alpha(l_1, l_2) \in (-\pi, \pi]$. Now, introduce the following function:

$$f(l_1, l_2) := \begin{cases} 0 & \alpha(l_1, l_2) = 0 \\ 1 & 0 < \alpha(l_1, l_2) \le \pi \\ -1 & -\pi < \alpha(l_1, l_2) < 0. \end{cases} \quad (8)$$

We start our manoeuvre to avoid the moving obstacle i at the time t_0. Let $h \in \{1, 2\}$ be the index for which

$$\min_{j=1,2} |\alpha(v_i(t_0) + l_i^{(j)}(t_0), v_R(t_0))| \quad (9)$$

is achieved. In other words, among the vectors $v_i(t_0) + l_i^{(j)}(t_0)$, $j = 1, 2$ we choose the one which makes the smallest angle between itself and the robot's velocity vector $v_R(t_0)$, and h is the index of such a vector. Now, we propose the following navigation law:

$$u(t) = -U_{\max} f((v_i(t) + l_i^{(h)}(t), v_R(t)),$$
$$V(t) = \|v_i(t) + l_i^{(h)}(t)\|. \quad (10)$$

Remark 3.1. It is clear that the control inputs defined by (10) satisfy the constraints (2). Indeed, (8) implies that $|u(t)| \le U_{max}$. Furthermore, (10), (5), and (7) imply that

$$V(t) = \|v_i(t) + l_i^{(h)}(t)\| \le \|v_i(t)\| + \|l_i^{(h)}(t)\|$$
$$\le V + (V_{max} - V) = V_{max}.$$

Therefore, the second constraint (2) also holds.

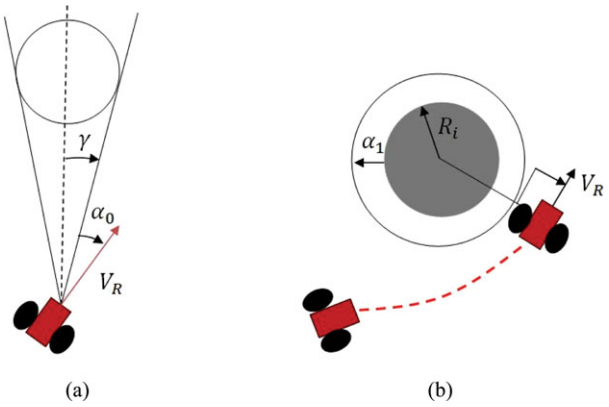


Fig. 3. (Colour online) (a) Constant avoiding angle α_0; (b) the robot moves along the circle of radius $\frac{R_i}{\cos\alpha_0}$.

Remark 3.2. The intuition behind the obstacle avoidance strategy (10) can be explained as follows. The law (10) belongs to the class of sliding mode control laws; see e.g. ref. [32] . The closed-loop system (1), (10) is a system of ordinary differential equations with a discontinuous right-hand side. The control law (10) steers the velocity vector $v_R(t)$ of the robot to a switching surface on which the robot's velocity becomes equal to the vector $v_i(t) + l_i^{(h)}(t)$. Now, consider a coordinates system moving with the velocity $v_i(t)$. In these coordinates, the obstacle i becomes stationary, and the robot is moving with the velocity $l_i^{(h)}(t)$. Furthermore, the motion with the velocity $l_i^{(h)}(t)$ is an obstacle avoidance strategy which keeps a constant avoiding angle α_0 between the instantaneous moving direction of the robot and one of the two boundary rays of the vision cone of the obstacle i from the robot; see Fig. 3(a). In other words, the control law consists of steering towards the closer of the two edges of the enlarged obstacle vision cone defined by (7). The idea of this obstacle avoidance strategy originates from biology where it was called negotiating obstacles with constant curvatures (see e.g. ref. [31]). An example of such a movement is a squirrel running around a tree. A similar strategy for avoiding stationary obstacles was proposed and studied in ref. [29].

Our proposed strategy consists of switching between the obstacle avoidance law (10) and the straight motion to the target at the maximum speed:

$$u(t) = 0; \quad V(t) = V_{\max}. \quad (11)$$

Let $C > 0$ be a given number. Furthermore, introduce the constants $a_i > 0$ as

$$a_i := \frac{R_i}{\cos\alpha_0} - R_i \quad \forall i = 1, 2, \ldots, k. \quad (12)$$

https://doi.org/10.1017/S0263574713000313 Published online by Cambridge University Press

Specifically, the rule for switching between (11) and (10) is as follows.

R1. Switching from (11) to (10) occurs at any time t_0 when the distance from the robot to the obstacle i reduces to the value C, i.e., $d_i(t_0) = C$ and $\dot{d}_i(t_0) < 0$;

R2. Switching from (10) to (11) occurs at any time t_* when $d_i(t_*) \leq 1.1a_i$ and the vehicle is oriented towards the target, i.e., $\theta(t^*) = H(t^*)$.

4. Mathematical Analysis of the Navigation Strategy

In this section, we prove that under a number of assumptions, the proposed navigation strategy (R1 and R2) is a target reaching strategy with collision avoidance (see Definition 2.1).

Our main theoretical result requires a number of assumptions.

Assumption 4.1. Every obstacle i is a disc of radius $R_i > 0$ moving with the constant velocity $v_i \neq 0$. Furthermore, $V_i < V_{\max} \quad \forall i = 1, 2, \ldots, k$ where $V_i := \|v_i\|$, and the vectors v_i and v_j are not collinear for all $i \neq j$.

Introduce the constants F_i as

$$F_i := \frac{(V_i + V_{\max})R_i}{(R_i + d_{\text{safe}})^2\sqrt{1 - \frac{R_i^2}{(R_i + d_{\text{safe}})^2}}} \quad \forall i = 1, 2, \ldots, k. \quad (13)$$

Assumption 4.2. The inequality $F_i < U_{max}$ holds for $i = 1, 2, \ldots, k$ where F_i are defined by (13).

Assumption 4.3. The parameter C in the switching rule R1 satisfies

$$C \geq \frac{\pi V_{max}}{U_{max} - F_i} + 1.1a_i \quad \forall i = 1, 2, \ldots, k.$$

Let $S_{ij}(t)$ denote the distance between the obstacles i and j at time t, and let $S_i(t)$ be the distance between the obstacle i and the target $\mathbf{T}$ at time t:

$$S_{ij}(t) := \min_{r \in D_i(t), q \in D_j(t)} \|r - q\|, \quad S_i(t) := \min_{r \in D_i(t)} \|r - \mathbf{T}\|.$$

Assumption 4.4. We assume that

$$S_{ij}(t) \geq 2C + \frac{\pi V_{max}}{U_{max} - F_i}$$
$$S_i(t) > 1.1a_i$$

for all $i \neq j, t$.

Assumption 4.5. The parameter α_0 in (6) and the safety margin d_{safe} satisfy

$$\alpha_0 \geq \arccos\left(\frac{R_i}{R_i + d_{safe}}\right) \quad \forall i = 1, 2, \ldots, k.$$

Remark 4.1. It obviously follows from Assumption 4.5 and (12) that $a_i \geq d_{safe}$ for all i.

Assumption 4.6. We also assume that at initial time $t = 0$, the robot is moving towards the target according to (11), and $d_i(0) > C$ for all i.

Theorem 4.1. *Let $d_{\text{safe}} > 0$, $0 < \alpha_0 < \frac{\pi}{2}$ and $C > 0$ be given and suppose that Assumptions* 4.1–4.6 *hold. Then, the control rule (R1 and R2) provides a target reaching strategy with collision avoidance.*

Proof of Theorem 4.1. First, we prove the following.

Claim. If the event R1 switching from (11) to (10) occurs at some time t_0, then there exists a time $t_* > t_0$ such that the event R2 switching from (10) to (11) occurs.

Indeed, the navigation law (10) steers the vector $v_i(t) + l_i^{(h)}(t)$ towards the robot velocity vector $v_R(t)$. It follows from a simple geometry that if $d_i(t) \geq d_{\text{safe}}$, then the angular velocity of the vector $v_i(t) + l_i^{(h)}(t)$ is less than F_i, where F_i is defined by (13). Therefore, Assumption 4.2 implies that the navigation law (10) guarantees that $v_i(t) + l_i^{(h)}(t) = v_R(t)$ for all $t > t_0 + \frac{\pi}{U_{\max} - F_i}$ until the event R2 switching from (10) to (11) occurs. Note that Assumption 4.4 implies that during the time interval when the robot is governed by the control law (10), $d_j(t) > C$ for all $j \neq i$. Hence, when the robot is avoiding the obstacle i, switching to avoiding any other obstacle cannot occur. Furthermore, if $v_i(t) + l_i^{(h)}(t) = v_R(t)$, we can consider a coordinates system moving with the velocity $v_i(t)$. In these coordinates, the obstacle i becomes stationary, and the robot is moving with the velocity $l_i^{(h)}(t)$. Furthermore, the motion with the velocity $l_i^{(h)}(t)$ is an obstacle avoidance strategy which keeps a constant avoiding angle α_0 between the instantaneous moving direction of the robot and one of the two boundary rays of the vision cone of the obstacle i from the robot; see Fig. 3(a). For the motion with the velocity $l_i^{(h)}(t)$, we have that

$$\dot{d}_i(t) = -(V_{\max} - V)\cos(\gamma(t) + \alpha_0), \quad (14)$$

where $\gamma(t) + \alpha_0$ is the angle between the line connecting the robot with the centre of the obstacle i and the robot's velocity vector; see Fig. 3(a). It follows from (14) that the distance $d_i(t)$ is decreasing if $\gamma(t) + \alpha_0 < \frac{\pi}{2}$. Therefore, the distance $d_i(t)$ is decreasing when the robot is outside the circle of radius $\frac{R_i}{\cos\alpha_0}$ positioned at the obstacle's centre, and $\dot{d}_i(t) = 0$ when the robot's coordinates belong to this circle. Hence, the robot's trajectory is converging to the circle of radius $\frac{R_i}{\cos\alpha_0}$ positioned at the obstacle's centre. On this circle, $v_R(t)$ becomes orthogonal to the line connecting the robot with the centre of the obstacle i (see Fig. 3b). It follows from Assumption 4.5 that the safety constraint $d_i(t) \geq d_{\text{safe}}$ will be satisfied. Moreover, it follows from (12) that the circle's radius $\frac{R_i}{\cos\alpha_0}$ equals $R_i + a_i$; hence, $d_i(t) \to R_i + a_i$. This implies that $d_i(t) \leq 1.1a_i$ for large enough t.

Furthermore, it is obvious that if the robot's trajectory is converging to a circle positioned at the obstacle's centre and the stationary target is outside of this circle (see Assumption 4.4), there exists a time t_* such that the vehicle is oriented

https://doi.org/10.1017/S0263574713000313 Published online by Cambridge University Press

towards the stationary target. Therefore, switching R2 from (10) to (11) occurs at some t_*. We have proved that every time the robot moves according to the obstacle avoiding law (10), it will start moving towards the stationary target (11).

Furthermore, Assumption 4.1 states that all obstacles move in different directions. Therefore, there exists a time $t_{**} > 0$ such that at time t_{**} the distance between any point on the straight line interval between the target and any point on the circle of radius $R_i + 1.1a_i$ positioned at the centre of the obstacle i is greater than C. Hence, after the time t_{**}, switching R1 to the obstacle avoiding control law (10) cannot occur. This and the Claim above immediately imply that the target will be reached. This completes the proof of Theorem 4.1.

5. Computer Simulations

In this section, we present computer simulation results for a wheeled mobile robot navigating in a dynamic environment with moving obstacles. The motion of the robot is governed by our biomimetic algorithm R1 and R2. The simulations are performed in Matlab.

The biomimetic algorithm R1 and R2 is applied in various dynamic environments, and the results are shown. First, we examine the simplest case: the environment with one moving obstacle as shown in Fig. 4. Figure 4(a) shows the initial position of the robot, the obstacle and the target, as well as the moving direction of the obstacle and the initial orientation of the robot. The paths of the robot and the obstacles are depicted in blue and green in Fig. 4(b), respectively. The speed and the angular velocity of the robot in simulation are shown in Figs. 4(c) and (d).

Figure 5 shows a more challenging case where the robot is in a dynamic environment with six moving obstacles. Figures 5(a), (c) and (e) display the crucial moments of the robot's manoeuvres to avoid each of the obstacles. Figures 5(b), (d) and (f) show the evaluation of distance between the robot and the closest obstacle. Note that the red circles Figs. 5(a), (c) and (e) indicate the moments when the distances between the robot and the obstacles are minimal, and the corresponding minimal distance values are similarly marked in Figs. 5(b), (d) and (f). In this experiment, the robot successfully reaches the target position and avoids every moving obstacles along its path.

The performance of our biologically inspired algorithm was compared with that of the velocity obstacle approach (VOA).[17,18] One scenario is illustrated in Fig. 6. In Figs. 6(a) and (b), a long obstacle is moving perpendicularly to the path between the robot and the target. Figures 6(c) and (d) show the moment when the robot is about to pass the lower end of the obstacle. As can be observed, the biomimetic algorithm avoids the obstacle by keeping a constant avoiding angle between the robot's moving direction and the lower end of the obstacle. On the other hand, the VOA drives the robot towards the lower end of the obstacle in a straight line and then move towards the target position. The complete paths for the robot to reach the target are shown in Figs. 6(e) and (f). The overall manoeuvre time for the biomimetic algorithm is 17.3 s and for the VOA it is 18.7 s. Hence, the biomimetic algorithm outperforms the VOA in this scenario in terms of manoeuvre time.

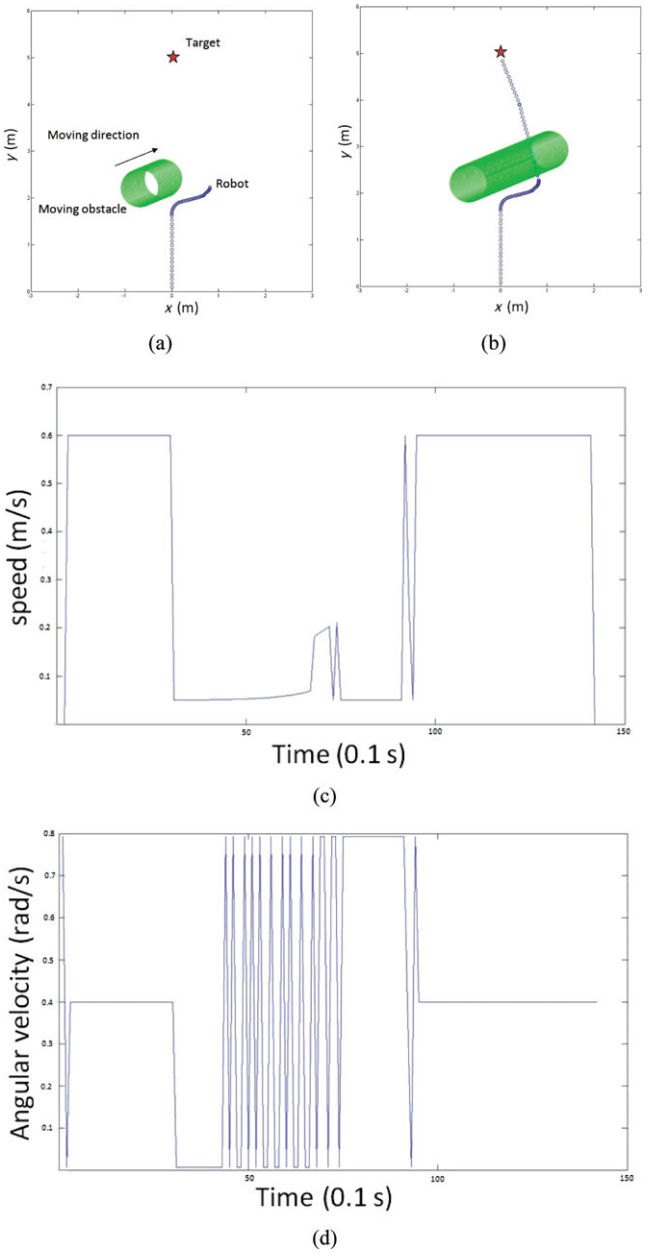


Fig. 4. (Colour online) Simple scenario with one moving obstacle.

6. Experiments with a Real Robot

The testing of the proposed algorithm with a real robot was conducted with an ActivMedia Pioneer 3-DX robot. The Pioneer 3-DX is a differential drive robot, and it has an on-board computer which can be controlled by a laptop via serial connection or WIFI connection. It is also equipped with a Sick laser range finder and 16 sonar sensors which provide distance and angle information to the obstacle if it enters the predefined scan range. To implement the proposed biomimetic algorithm on this robot, we use the C++ programming language and active media robotics application, ARIA, which is an object-oriented C++ library for controlling ActivMedia mobile robots, running in the Linux operating system.

Note that the odometry system has been used to estimate the position of the robot, the obstacle, and the target. Knowing the robot's position at each time, the robot's heading $H(t)$

https://doi.org/10.1017/S0263574713000313 Published online by Cambridge University Press

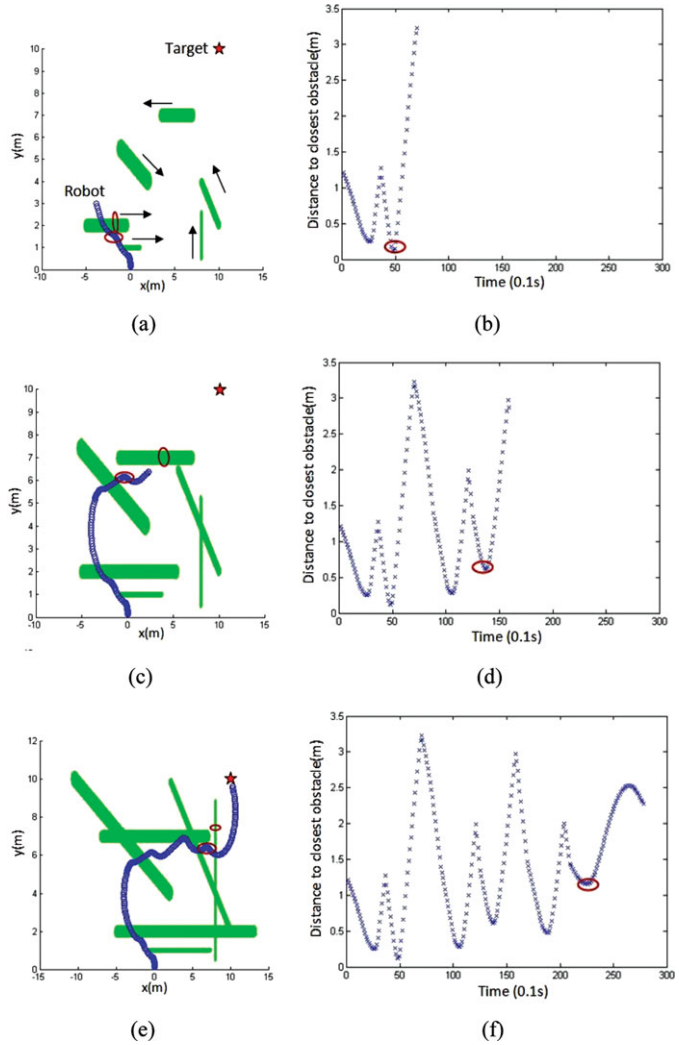


Fig. 5. (Colour online) Robot navigation in cluttered dynamic environment.

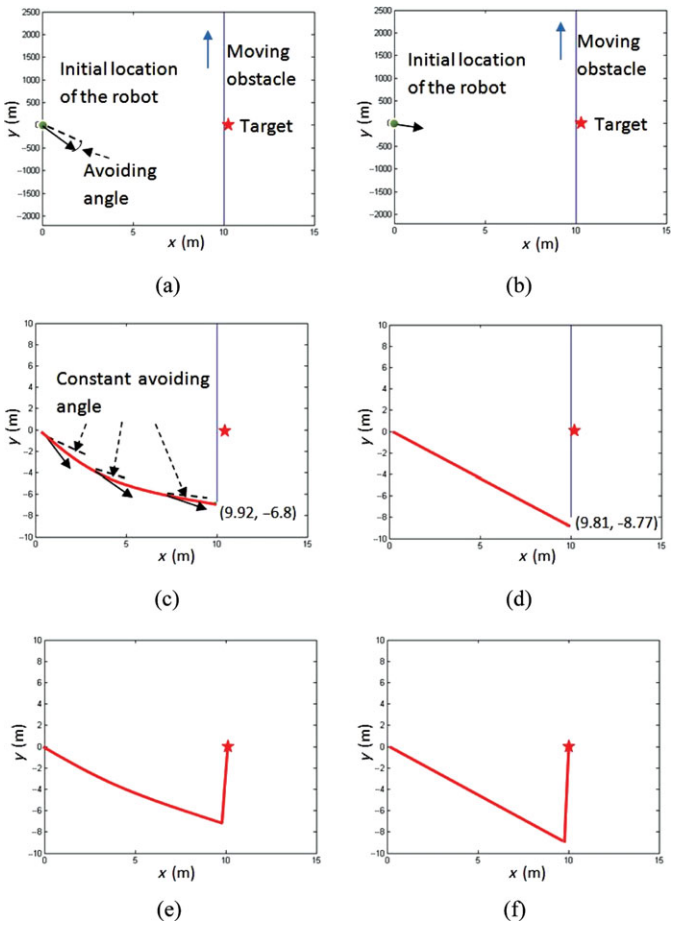


Fig. 6. (Colour online) Biomimetic algorithm (left column) versus VOA (right column).

to the target (**T**) can be computed. Furthermore, the laser and sonar range finders provide the relative distance $d(t)$ to the obstacle and the angle $\theta(t)$ (see Fig. 7). Therefore, the coordinates of the obstacle's centre can be computed by

$$x_{ob} = x_R + d\cos(\theta),\ y_{ob} = y_R + d\sin(\theta)$$

and thus the velocity of the obstacle is estimated by

$$V_{ob} = \frac{\sqrt{(y_{ob}(t) - y_{ob}(t-1))^2 + (x_{ob}(t) - x_{ob}(t-1))^2}}{T_s},$$

$$\theta_{ob} = \operatorname{atan}\left(\frac{y_{ob}(t) - y_{ob}(t-1)}{x_{ob}(t) - x_{ob}(t-1)}\right).$$

We consider an environment with three moving obstacle as shown in Fig. 8. The velocities of the obstacle can be estimated and the navigation of the robot is governed by our biologically inspired or biomimetic algorithm. Figures 8(a–d) show the snapshots of the experiment. The moving direction of the person (moving obstacle) is indicated by an arrow and the trajectory of the robot is indicated by dashed lines. The overall trajectory is depicted in Fig. 8(e).

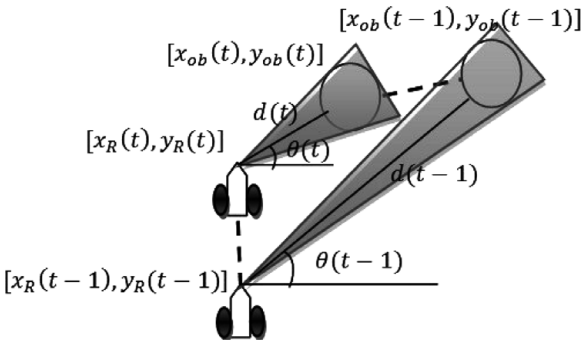
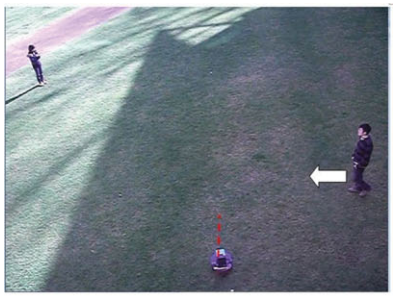


Fig. 7. Estimation of obstacle's velocity.

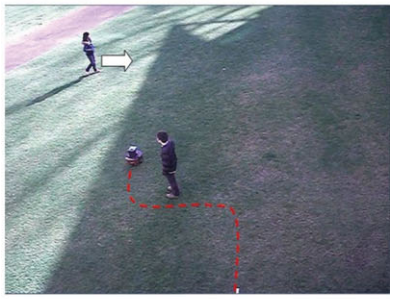
We also test the proposed biomimetic navigation algorithm in a more complicated real-world scenario, robot navigation in a narrow corridor with stationary and moving obstacles, as illustrated in Fig. 9. In this scenario, the robot needs to avoid the obstacles and not to collide with the walls. Figures 9(a–c) show the moments when the robot avoids the stationary and moving obstacles, while keeping a certain distance to the walls. The complete path taken by the robot is drawn in Fig. 9(d).

Figure 10 shows a situation where the robot is in an environment with two moving obstacles, the velocities of the obstacles are unknown to the robot in prior and has to be estimated in real time. Figure 10(a) shows the path of

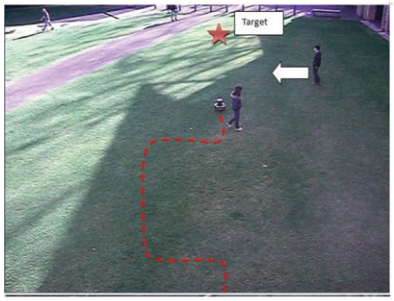
https://doi.org/10.1017/S0263574713000313 Published online by Cambridge University Press



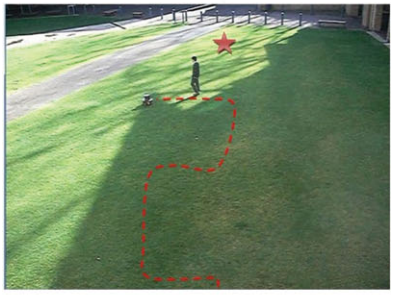
(a)



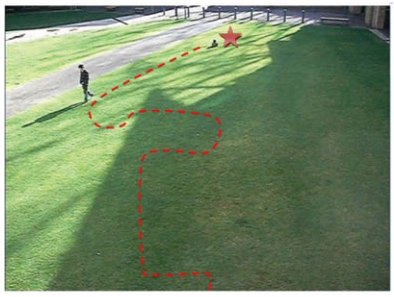
(b)



(c)

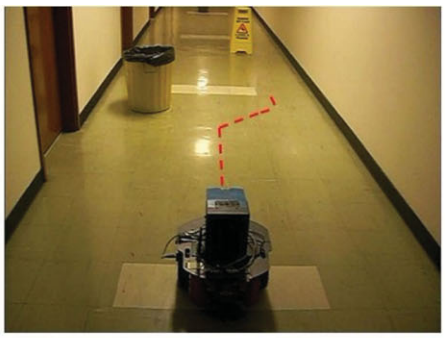


(d)

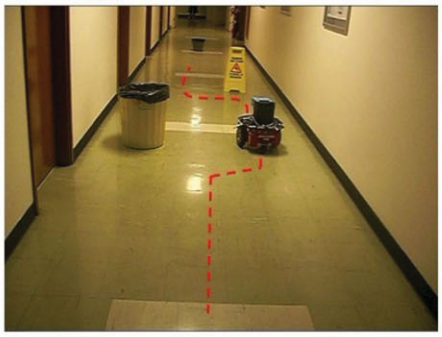


(e)

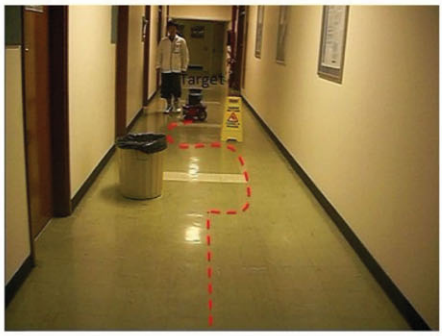
Fig. 8. (Colour online) Robot avoiding obstacles in a narrow corridor.



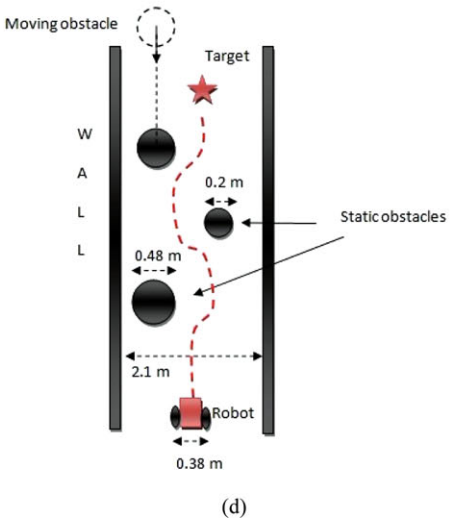
(a)



(b)



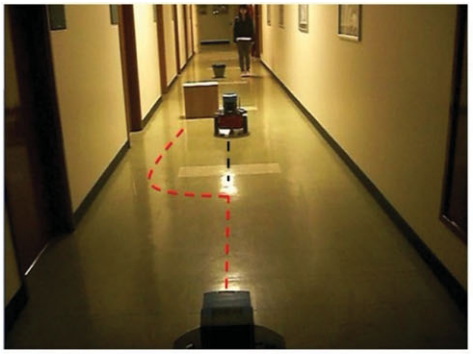
(c)



(d)

Fig. 9. (Colour online) Robot avoiding three moving obstacles.

https://doi.org/10.1017/S0263574713000313 Published online by Cambridge University Press



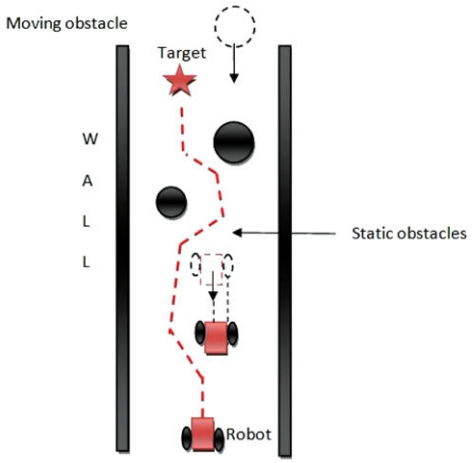
(a)



(b)



(c)



(d)

Fig. 10. (Colour online) Robot navigation in a cluttered dynamic environment.

the robot while avoiding the first moving obstacle, which comes towards the robot along a straight line. Figures 10(b) and (c) depict the moment when the robot is avoiding the stationary obstacle and then immediately avoiding the second moving obstacle. The overall path is shown in Fig. 10(d) along with the distance information from the laser. Note that our theoretical analysis is given for the navigation algorithm that is based on infinite vision cones of obstacles. In our real-life experiments, the obstacle vision cones are finite. In particular, the vision cone of a wall consists of straight lines connecting the robot and points of the wall at the distance less than some value D.

7. Conclusions

We have proposed a novel biologically inspired reactive algorithm to navigate a wheeled mobile robot in dynamic cluttered environments with moving obstacles. A mathematically rigorous analysis of the algorithm has been given. The performance of the proposed real-time navigation strategy has been confirmed by extensive computer simulations and experiments with a Pioneer P3-DX mobile wheeled robot.

Acknowledgement

This work was supported by the Australian Research Council.

References

1. J. Latombe, *Robot Motion Planning* (Kluwer, London, 1991).
2. J. Minguez and L. Montano, "Robot Navigation in Very Complex, Dense and Cluttered Indoor/Outdoor Environments," *Proceedings of the 15th IFAC World Congress*, Barcelona, Spain (Jul. 2002) pp. 218–223.
3. 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).
4. Z. Shiller, "Online suboptimal obstacle avoidance," *Int. J. Robot. Res.* **19**(5), 480–497 (2000).
5. I. Kamon and E. Rivlin, "Sensory-based motion planning with global proofs," *IEEE Trans. Robot. Autom.* **13**(6), 814–822 (1997).
6. I. Kamon, E. Rimon and E. Rivlin, "Tangentbug: A range-sensor-based navigation algorithm," *Int. J. Robot. Res.* **17**(9), 934–953 (1998).
7. Y. H. Liu and S. Arimoto, "Path planning using a tangent graph for mobile robots among polygonal and curved obstacles," *Int. J. Robot. Res.* **11**(4), 376–382 (1992).
8. N. Vlassis, N. Sgouros, G. Efthivolidis and G. Papakonstantinou, "Global Path Planning for Autonomous Qualitative Navigation," *Proceedings of the IEEE Conference on Tools with Artificial Intelligence*, Toulouse, France (Nov. 1996) pp. 354–359.
9. S. Belkhous, A. Azzouz, M. Saad, V. Nerguizian and C. Nerguizian, "A novel approach for mobile robot navigation with dynamic obstacles avoidance," *J. Intell. Robot. Syst.* **44**(3), 187–201 (Nov. 2005).
10. M. Deng, A. Inoue, Y. Shibata, K. Sekiguchi and N. Ueki, "An Obstacle Avoidance Method for Two Wheeled Mobile Robot," *Proceedings of the 2007 IEEE International Conference on Networking, Sensing and Control*, London, UK (Apr. 2007) pp. 689–692.
11. 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 (Feb. 2010).

https://doi.org/10.1017/S0263574713000313 Published online by Cambridge University Press

12. A. Matveev, H. Teimoori and A. Savkin, "A method for guidance and control of an autonomous vehicle in problems of border patrolling and obstacle avoidance," *Automatica* **47**, 515–524 (2011).
13. M. Seder, K. Macek and I. Petrovic, "An Integrated Approach to Realtime Mobile Robot Control in Partially Known Indoor Environments," *Proceedings of the 31st Annual Conference of the IEEE Industrial Electronics Society*, Raleigh, North Carolina, USA (Nov. 2005) pp. 1785–1790.
14. D. Fox, W. Burgard and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robot. Autom. Mag.* **4**, 23–33 (1997).
15. R. Simmons, "The Curvature-Velocity Method for Local Obstacle Avoidance," *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 4, Minesota, USA (Nov. 1996) pp. 3375–3382.
16. Y. Nak and R. Simmons, "The Lane–Curvature Method for Local Obstacle Avoidance," *Proceedings of the IEEE International Conference on Robotics and Automation*, Leuven, Belgium (Nov. 1998) pp. 1615–1621.
17. P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using velocity obstacles," *Int. J. Robot. Res.* **17**(7), 760–772 (Jul. 1998).
18. F. Large, C. Lauger and Z. Shiller, "Navigation among moving obstacles using the NLVO: Principles and applications to intelligent vehicles," *Auton. Robots*, **19**, 159–171 (2005).
19. A. Chakravarthy and D. Ghose, "Obstacle avoidance in a dynamic environment: A collision cone approach," *IEEE Trans. Syst. Man Cybern.* **28**(5), 562–574 (1998).
20. T. Fraichard and H. Asama, "Inevitable Collision States. A Step Towards Safer Robots?," *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, Las Vegas, Nevada, USA (Oct. 2003) pp. 388–393.
21. E. Owen and L. Montano, "A Robocentric Motion Planner for Dynamic Environments using the Velocity Space," *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, Beijing, China (Oct. 2006) pp. 2833–2838.
22. S. Lindemann, I. Hussein and S. M. LaValle, "Real Time Feedback Control for Nonholonomic Mobile Robots with Obstacles," *Proceedings of the 45th IEEE Conference on Decision and Control*, San Diego, USA (Dec. 2006) pp. 2406–2411.
23. A. Ferrara and M. Rubagotti, "Sliding Mode Control of a Mobile Robot for Dynamic Obstacle Avoidance Based on a Time-Varying Harmonic Potential Field," *ICRA 2007 Workshop: Planning, Perception and Navigation for Intelligent Vehicles*, Rome, Italy (Apr. 2007).
24. J. Chunyu, Z. Qu, E. Pollak and M. Falash, "Reactive Target-Tracking Control with Obstacle Avoidance of Unicycle-Type Mobile Robots in a Dynamic Environment," *Proceedings of the American Control Conference*, Baltimore, MD (Jun. 2010) pp. 1190–1196.
25. E. Masehian and Y. Katebi, "Robot motion planning in dynamic environments with moving obstacles and target," *World Acad. Sci. Eng. Technol.* **29**, 107–112 (2007).
26. A. S. Matveev, C. Wang and A. S. Savkin, "Real-time navigation of mobile robots in problems of border patrolling and avoiding collisions with moving and deforming obstacles," *Robot. Auton. Syst.* **60**(6), 769–788 (2012).
27. P. Raja and S. Pugazhenthi, "Path planning for a mobile robot in dynamic environments," *Int. J. Phys. Sci.* **6**, 4721–4731 (2011).
28. Y. Zhu, T. Zhang, J. Song, X. Li and M. Nakamura, "A new method for mobile robots to avoid collision with moving obstacles," *Artif. Life Robot.* **16**, 507–510 (2012).
29. H. Teimoori and A. V. Savkin, "A biologically inspired method for robot navigation in a cluttered environment," *Robotica* **28**(5), 637–648 (2010).
30. A. Matveev, H. Teimoori and A. Savkin, "Range-only measurements based target following for wheeled mobile robots," *Automatica* **47**(6), 177–184 (2011).
31. D. N. Lee, "Guiding movement by coupling taus," *Ecol. Psychol.* **10**(3–4), 221–250 (1998).
32. V. I. Utkin, *Sliding Modes in Control Optimization* (Springer-Verlag, Berlin, 1992).

https://doi.org/10.1017/S0263574713000313 Published online by Cambridge University Press