Robotica (2013) volume 31, pp. 525–537. © Cambridge University Press 2012
doi:10.1017/S0263574712000550

Dynamic collision risk modeling under uncertainty

F. Belkhouche†* and B. Bendjilali‡

†*Department of Electrical Engineering, CSU Sacramento, CA 95819, USA*
‡*Department of Mathematics, Raritan Valley Community College, Branchburg, NJ 08876, USA*

(Accepted September 4, 2012. First published online: October 3, 2012)

SUMMARY
This paper introduces a probabilistic model for collision risk assessment between moving vehicles. The uncertainties in the states and the geometric variables obtained from the sensory system are characterized by probability density functions. Given the states and their uncertainties, the goal is to determine the probability of collision in a dynamic environment. Two approaches are discussed: (1) The virtual configuration space (VCS), and (2) the rates of change of the visibility angles. The VCS is a transformation of observer that reduces collision detection with a moving object to collision detection with a stationary object. This approach allows to create simple geometric collision cones. Error propagation models are used to solve the problem when going from the VCS to the configuration space. The second approach derives the collision conditions in terms of the rate of change of the limit visibility angles. The probability of collision is then calculated. A comparison between the two methods is carried out. Results are illustrated using simulation, including Monte Carlo simulation.

KEYWORDS: Motion planning; Control of robotic systems; Mobile robots, Navigation, Robot dynamics.

1. Introduction

Conflict risk assessment and resolution are important aspects in intelligent transportation and robotics. Both deterministic and probabilistic approaches are used for conflict detection. Deterministic approaches might handle uncertainty in observed geometry and estimated trajectories. While methods used in computer graphics are mainly deterministic, probabilistic approaches play an important role in robotics and intelligent transportation. Probabilistic methods present an advantage in tackling uncertainties and errors in modeling, sensing, and actuation. Collision detection has two aspects: Static collision detection (overlap for a given configuration), and dynamic collision detection (overlap for all reachable configurations). In this paper, we are concerned with dynamic collision detection in the presence of uncertainty. We formulate the problem as follows: Given the present states and their uncertainties, what are the chances of overlap in the future? This is equivalent to finding the probability of collision given the present states and motion patterns of the moving vehicles/objects. The objective is to reduce complex collision checking scenarios to simple probabilistic rules, from which a decision is made.

Probabilistic collision detection is widely studied in the aerospace community. Detecting and preventing conflicts between satellites in formation and aircrafts is an important priority in air traffic management systems. Due to many internal factors (such as actuation) and external factors (such as wind), the trajectories are not exactly known. Thus, uncertainties and unmodeled dynamics must be taken into account to achieve better reliability. Motivated by increasing the capacity of the air space and improving safety and autonomy, researchers in the aerospace community are looking for automated methods for conflict detection and resolution.

Some models used in the aerospace community assume straight line motion with constant speed. While these assumptions simplify the problem considerably, they are not always realistic. The encounter model is widely used[1–3] to determine the probability of collision. This model uses the relative motion to construct a conflict region characterized by an ellipsoid error centered around the rendezvous point. A two-stage algorithm is described in ref. [4]. In the first stage, maneuvers are derived to avoid collision, and in the second stage, maneuvers are derived to optimize path under wind uncertainty. Certain models take flight trajectories into account,[3] others map uncertainties into the conflict region.[5] Under the assumption of constant velocity, it is possible to construct a probabilistic conflict region, and calculate the probability of collision. The construction of the conflict region is not straightforward when the velocity is not constant or in the presence of unpredicted factors affecting the motion.

Nonlinearities increase the complexity of the collision-detection problem and render analytical solutions more difficult. Computational methods are attractive tools to solve such problems.[6] The Monte Carlo simulation methods are used to determine the probability of collision in refs. [7, 8], and to validate theoretical results in ref. [5]. These methods use a large number of iterations. They are powerful simulation tools that can be used to approximate probability density functions (pdf) and overcome the problems of non-Gaussian noise and nonlinear systems, and thus deduce the probability of collision for the most difficult modes. However, an online response is very important for collision avoidance, especially in highly dynamic environments.

In the robotics and intelligent transportation, the collision-detection problem is usually combined with path planning.[9–13] Sensors and actuators are important sources of

* Corresponding author. E-mail: belkhouf@ecs.csus.edu

https://doi.org/10.1017/S0263574712000550 Published online by Cambridge University Press

uncertainties. In ref. [10], the probabilistic velocity obstacle is combined with the occupancy grid. The probability of collision is determined based on the shape, the velocity, and the position. The probabilistic grid approach is also used in ref. [9], where the Bayesian estimation is used to update the map. Another probabilistic map of the environment is discussed in ref. [14]. In ref. [15], the probability of collision is calculated based on the product of two Gaussians. Speed control is used to avoid collision. Optimal path is discussed in refs. [11, 12], where a stochastic model for the kinematics is derived in ref. [12], and a linear programming approach formulated as a stochastic problem is used in ref. [11]. Linear models are widely used.[11,12] These models allow to bring important simplifications. However, these are only an approximation. Methods in intelligent transportation focus on dynamic models.[8,16–18] Time to collision and predicted decreasing range are simple approaches used for collision risk assessment.[19] The time derivative of the probability of conflict is used in ref. [18] to calculate the conflict zone. Numerical integration is then performed. Reachability analysis is used in refs. [20, 21]. This is a more sophisticated approach that can be integrated with stochastic information. A reachable state is a state to which the system can evolve. Vehicle and road parameters play an important role in collision avoidance. Different models for the vehicle and its subsystems (break, steering, etc.) are used[16] for this purpose.

This paper studies situations where a vehicle moves in a shared dynamic workspace. The motion of the obstacles is not *a priori* known but is measurable. The collision models discussed here take uncertainties in sensing, actuation, and geometry into account. The objective is to predict future collision based on the motion patterns of the vehicles/objects and the geometry of the workspace. The uncertainties are represented by probability density functions. Numerical integration is used when closed form solution is not possible or difficult to obtain. Analytical solutions may be difficult due to the complexity arising from the dimensional nature of the system. Computationally feasible algorithms are highly desired for the real-time response needed in practice. Variables obtained from the sensory system are assumed to follow Gaussian distribution with given mean values and covariance matrices. The Gaussian distribution assumption for sensing measurement is a reasonable one and has been used by many authors.[22–24] The suggested method uses dynamic models that take uncertainties in motion and geometry into account. Two approaches are discussed: the virtual configuration space (VCS), and the rate of change of visibility angles. The advantages of the method can be summarized as follows:

- The use of VCS simplifies the problem considerably, and allows the use of a simple geometric collision cone similar to static collision detection. The simplicity of the algorithm allows for fast calculations and therefore online response.
- The method based on the rate of change of the visibility angles uses the rate of change of the relative range but does not necessarily use the speeds and orientations of the vehicles. Therefore, this approach allows to detect dynamic collision even with incomplete information.
- Both approaches are adequate for continuous motion with no assumption on the speed and orientation; thus, speed and orientation could be time-varying.

In our approach, collision risk is assessed based on the probability of collision (P_{col}). A collision alert is generated when P_{col} is greater than a certain user-defined threshold P_{th}. Monte Carlo simulation is used to validate the results. Monte Carlo simulation is applied to the nonlinear deterministic models which are part of the collision checking algorithm. The input of the Monte Carlo simulator is the distributions of the independent variables (mainly geometric and kinematic variables). The output is the distribution of the dependent variable of the deterministic model. Interested readers are referred to ref [25] for more information about Monte Carlo methods.

This paper is organized as follows: In the next section, we discuss the modeling problem and introduce geometric variables. Section 3 introduces the deterministic models for collision detection, where VCS and collision checking based on the visibility angles are discussed. Section 4 discusses the probabilistic models and derive algorithms to calculate the probability of collision. Finally, simulation is carried out in Section 5.

2. Modeling and Geometry

The workspace is attached to a global reference frame of coordinates $\mathcal{W}_{\mathcal{O}}$ and is shared by many moving vehicles/objects with independent motion plans. The moving objects or vehicles have a circular shape. With reference to Fig. 1, the vehicle of interest A_o shares the workspace with other vehicles $A_i, i = 1, \ldots, N$. Note that for N vehicles, there are $N(N-1)/2$ pairwise interactions. A vehicle A_i within a given distance from A_o is seen as a potential obstacle with respect to A_o. Subscripts i and o are used throughout the paper to indicate quantities belonging to A_i and A_o respectively. Dynamic collision risk assessment plays a crucial role in achieving safe navigation. Vehicles A_i and A_o are moving in $\mathcal{W}_{\mathcal{O}}$ according to the following kinematics model:

$$\begin{aligned} \dot{x}_k &= v_k \cos\theta_k, \\ \dot{y}_k &= v_k \sin\theta_k, \\ \dot{\theta}_k &= \omega_k, \\ \dot{v}_k &= a_k, \end{aligned} \tag{1}$$

where $k = \{i, o\}$, pair (x_k, y_k) represents the coordinates of the vehicle's reference point in $\mathcal{W}_{\mathcal{O}}$, v_k is the speed, and θ_k is the heading angle with $\theta_k \in [0, 2\pi]$. The reference point of A_i is the center of the bounding circle. The speeds and heading angles could be constant or time-varying. Vehicles A_i and A_o are characterized by states $\mathbf{x}_i = [x_i, y_i, \theta_i, v_i]^T$ and $\mathbf{x}_o = [x_o, y_o, \theta_o, v_o]^T$ respectively. In this paper, $f_X(x)$ denotes the probability density function of variable x, σ_x is its standard deviation, and $\hat{x}$ is the mean value. $f_{XY}(x, y)$ denotes the joint probability density function of pair x, y. We assume a normal distribution for $\mathbf{x}_i$ and $\mathbf{x}_o$, with mean vectors $\hat{\mathbf{x}}_i = [\hat{x}_i, \hat{y}_i, \hat{\theta}_i, \hat{v}_i]^T$ and $\hat{\mathbf{x}}_o = [\hat{x}_o, \hat{y}_o, \hat{\theta}_o, \hat{v}_o]^T$, and covariance matrices C_i and C_o respectively.

https://doi.org/10.1017/S0263574712000550 Published online by Cambridge University Press

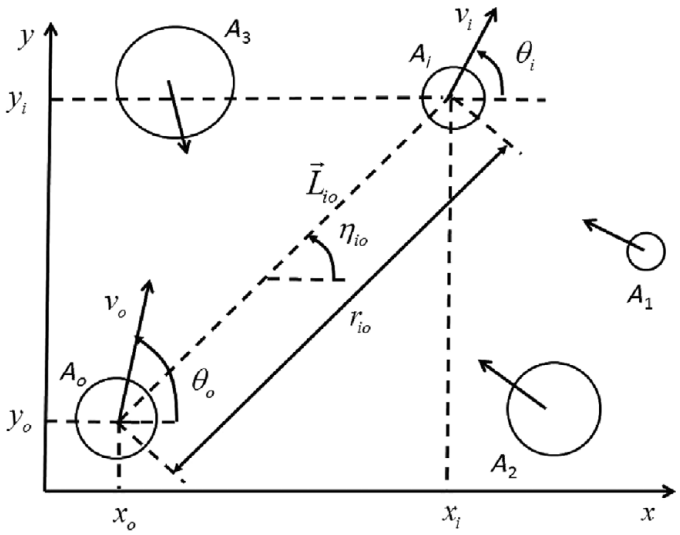


Fig. 1. Illustration of the geometry where the vehicle of interest A_o shares the workspace with other moving vehicles/objects $A_1, A_2, \ldots, A_i, \ldots$ Collision checking is done in pairs. Only vehicles within a given distance are considered.

The geometric variables shown in Fig. 1 are summarized as follows:

- $\vec{L}_{io}$ is the visibility line between A_i and A_o. This is an imaginary line that gives the direction of A_i with respect to A_o.
- $r_{io} = \sqrt{(x_i - x_o)^2 + (y_i - y_o)^2}$ is the relative range between the reference points of A_i and A_o.
- η_{io} is the visibility line angle with $\tan \eta_{io} = \frac{y_i - y_o}{x_i - x_o}$. $\eta_{io} \in [0, 2\pi]$ is measured from the positive x-axis.
- ξ is the collision cone angle. Let d_i be the radius of the augmented vehicle, the collision cone angle is given by $\sin \xi = d_i / r_{io}$.

The most widely used sensors in navigation and collision avoidance provide range and bearing measurements. Let A_i be a vehicle within a given distance from A_o, it is assumed that the onboard sensory system of A_o provides the following information:

- The minimum distance r_m between A_i and A_o.
- The visibility angle η_{io} and the collision cone angle ξ. We define vector $\nu = [\eta_{io}, \xi]^T$. While the size of A_o is known, the size of A_i is deduced from the collision cone angle.
- The orientation and the speed of A_i.

The interoceptive sensors of A_o provide its orientation and speed. The relative velocity in the Cartesian coordinate system is given by $\dot{x}_{io} = \dot{x}_i - \dot{x}_o$ and $\dot{y}_{io} = \dot{y}_i - \dot{y}_o$. The relative kinematics equations in polar form between the reference points of A_i and A_o are given by[26]

$$\begin{aligned} \dot{r}_{io} &= v_i \cos(\theta_i - \eta_{io}) - v_o \cos(\theta_o - \eta_{io}), \\ r_{io}\dot{\eta}_{io} &= v_i \sin(\theta_i - \eta_{io}) - v_o \sin(\theta_o - \eta_{io}). \end{aligned} \tag{2}$$

This system gives the relative motion of A_i as observed by A_o. It can be used to deduce the necessary and sufficient conditions for collision. A negative value for the rate of change of the range implies that A_i and A_o are approaching each other. However, this is not a sufficient condition for collision and it may lead to false alarm.[26] On the other side, an increasing range implies no collision risk. The algorithms discussed in this paper do not use range measurements only. For simplicity and without loss of generality, A_o is reduced to a single point, while A_i is augmented in the size by the radius of A_o. In the rest of the paper, A_o stands for the point-object and A_i for the augmented object, unless mentioned otherwise. This transformation is shown in Fig. 2. Let p be a point on A_i and $\vec{n}$ its normal. $\vec{L}$ is the visibility line to p. p is visible to A_o when the following condition is satisfied,[27]

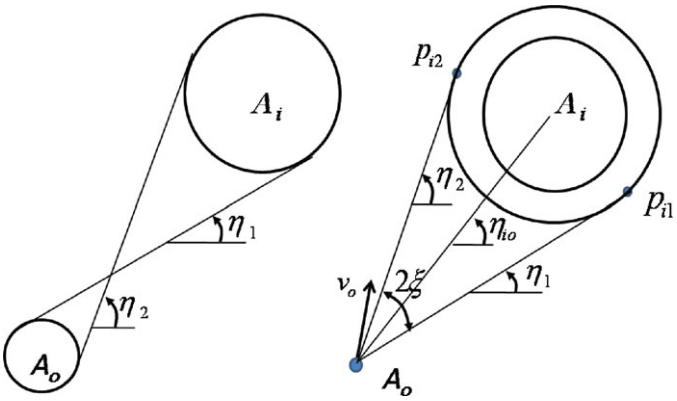


Fig. 2. (Colour online) Illustration of the geometric collision cone.

$$\vec{n} \times \vec{L} \leq 0. \tag{3}$$

The visible surface in A_i from A_o is an arc with end points (p_{i1}, p_{i2}). These points satisfy

$$\vec{n} \times \vec{L} = 0. \tag{4}$$

That is, the angle between their line of sight and the normal is $\pi/2$, which is the maximum viewing angle. The range of (p_{i1}, p_{i2}) is denoted by r_e and their visibility angles are denoted by η_1 and η_2. These angles are invariant under the augmentation-of-size transformation as shown in Fig. 2. The rates of change of these angles denoted by $\dot{\eta}_1$ and $\dot{\eta}_2$ play an important role in our formulation. We define vector $\eta = [\eta_1, \eta_2]^T$. ν and η are related by

$$\eta = T_1 \nu = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \nu. \tag{5}$$

The geometric collision cone is defined as

$$\mathcal{CC} = [\eta_{io} - \xi, \eta_{io} + \xi]. \tag{6}$$

Recall that ξ is the collision cone angle. The total width of the collision cone is 2ξ. The conflict region is defined as the region in the workspace where collision may take place. Figure 3 shows an illustration of the conflict region when orientations are constant, and the limit values for the geometric collision cone angle.

https://doi.org/10.1017/S0263574712000550 Published online by Cambridge University Press

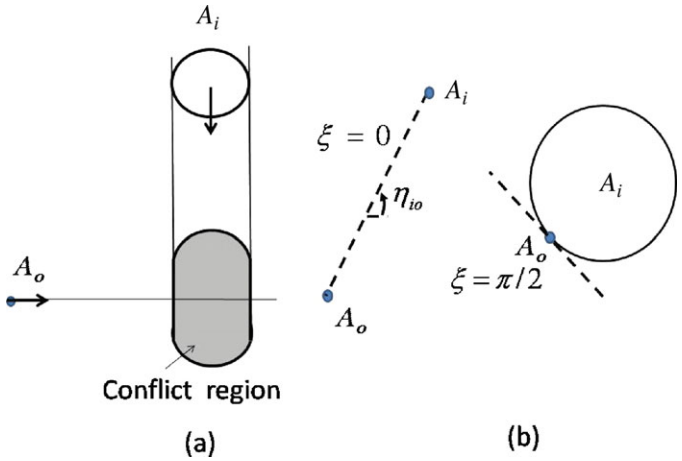


Fig. 3. (Colour online) (a) An illustration of the conflict region when orientations are constant; (b) limit values for the geometric collision cone angle. $\xi = 0$ when A_i is a geometric point, and $\xi = \pi/2$ when A_o is tangent to A_i.

3. Dynamic Collision Checking: The Deterministic Model

This paper describes two approaches for dynamic collision checking. The first approach is based on the transformation of coordinates that we call the VCS. The second approach uses the evolution of the visibility line angles with the outmost points of the obstacle, as well as the rate of change of the relative range. We first discuss the deterministic models.

3.1. The virtual configuration space (VCS)

The VCS is a transformation of observer introduced recently for path planning in dynamic environments.[28] The VCS associates to every pair (A_o, A_i) of vehicles a new pair (A_o^*, A_i^*). The velocity vectors of the new pair are given by

$$\begin{aligned} \vec{v}_o^* &= \vec{v}_o - \vec{v}_i, \\ \vec{v}_i^* &= 0. \end{aligned} \tag{7}$$

A_i^* is simply a representation of the initial position of A_i. Under this transformation, collision between two moving vehicles (A_o, A_i) is reduced to collision between a new moving vehicle A_o^* and a stationary object $A_i^* = A_i(t_0)$. The speed of A_o^* is given by

$$v_o^* = \sqrt{\dot{x}_{io}^2 + \dot{y}_{io}^2} \tag{8}$$

and its orientation by

$$\tan\theta_o^* = \dot{y}_{io}/\dot{x}_{io}. \tag{9}$$

The relative kinematics model between A_o^* and $A(t_0)$ is given by

$$\begin{aligned} \dot{r}_{io} &= -v_o^* \cos(\theta_o^* - \eta_{io}), \\ r_{io}\dot{\eta}_{io} &= -v_o^* \sin(\theta_o^* - \eta_{io}). \end{aligned} \tag{10}$$

This model can be obtained from Eq. (2) when $v_i = 0$. The VCS has several interesting properties, which are as follows:

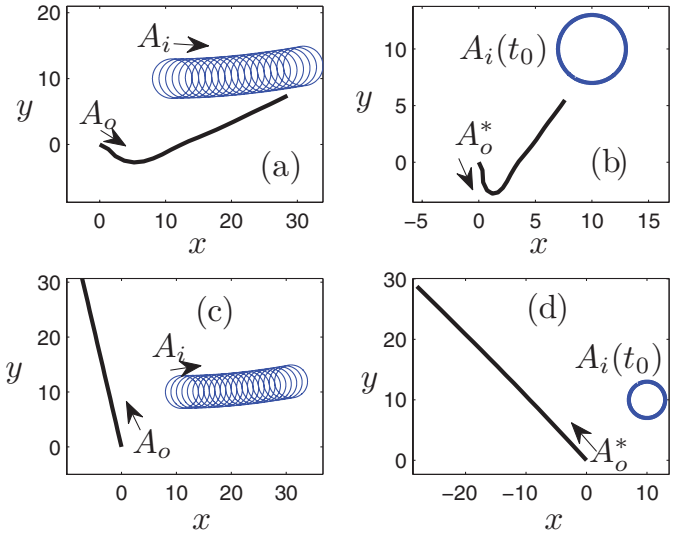


Fig. 4. (Colour online) Illustration of the VCS. Left side figures represent the scenarios in the real plane. Right side figures represent the corresponding VCS.

- r_{io} and η_{io} are invariant under the VCS, and so does the geometric collision cone. This implies that η_1 and η_2 and their rates are also invariant under the transformation.
- VCS is invertible (one-to-one and onto). Therefore, it is possible to obtain the configuration space from the VCS.
- The conflict region in the VCS is centered around $A_i(t_0)$. This conflict region is independent from the speeds and orientations.
- θ_o^* is constant when the motion of A_i is linear with respect to A_o.
- It is possible to associate to pair (A_o, A_i) a new pair $(A_o(t_0), A_i^*)$ (instead of $(A_o^*, A_i(t_0))$). In this case A_o is transformed to a stationary object. This property is used in Example 5 in the Simulation section (Section 5).

Collision between $A_i(t_0)$ and A_o^* does not depend on the speed of A_o^*, it depends on its orientation only. Therefore, VCS allows to use a formulation similar to the geometric collision cone, in this case, between $A_i(t_0)$ and A_o^*. The inverse transformation is used to map the result to the configuration space. A_i and A_o are in a collision course when $\hat{\theta}_o^* \in \mathcal{CC}$.

The following example is an illustration of the VCS.

Example 1. Figure 4 shows two scenarios illustrating the VCS. The left side in Fig. 4 represents the scenarios in the real plane. Their corresponding VCS scenarios are on the right side. The initial positions for A_o and A_i are $(0, 0)$ and $(10, 10)$ respectively. Collision takes place in the top scenario as it clearly appears in the VCS. There is no collision in the bottom scenario, which is also reflected in VCS. Clearly studying collision is much easier in the VCS, where the rendezvous area is centered around $A_i(t_0)$.

3.2. Collision check based on the visibility angles

The geometric collision cone is a simple tool to detect future possible collisions with a stationary object. The VCS allows

https://doi.org/10.1017/S0263574712000550 Published online by Cambridge University Press

to create similar collision cone with a moving object. Another way to characterize collision is through the visibility angles with the outmost points p_{i1} and p_{i2} of the obstacle. Recall that the visibility angles of these points with respect to A_o are η_1 and η_2 respectively. Their range is r_e. The kinematics equations for the triplet (r_e, η_1, η_2) are given by

$$\begin{aligned} \dot{r}_e &= v_i \cos(\theta_i - \eta_1) - v_o \cos(\theta_o - \eta_1), \\ r_e \dot{\eta}_1 &= v_i \sin(\theta_i - \eta_1) - v_o \sin(\theta_o - \eta_1), \\ r_e \dot{\eta}_2 &= v_i \sin(\theta_i - \eta_2) - v_o \sin(\theta_o - \eta_2). \end{aligned} \tag{11}$$

Since r_e, η_1, and η_2 are invariant under the VCS, system (11) can be written as

$$\begin{aligned} \dot{r}_e &= -v_o^* \cos(\theta_o^* - \eta_1), \\ r_e \dot{\eta}_1 &= -v_o^* \sin(\theta_o^* - \eta_1), \\ r_e \dot{\eta}_2 &= -v_o^* \sin(\theta_o^* - \eta_2). \end{aligned} \tag{12}$$

In order to explain the approach, consider the case when A_i is stationary. The principle is valid when A_i is moving as well. When A_i is stationary, the relative kinematics system in Eq. (2) becomes

$$\begin{aligned} \dot{r}_{io} &= -v_o \cos(\theta_o - \eta_{io}), \\ r_{io} \dot{\eta}_{io} &= -v_o \sin(\theta_o - \eta_{io}). \end{aligned} \tag{13}$$

Note that $\theta_o = \eta_{io}$ when A_o is directly heading toward the reference point of A_i. As a result, system (13) satisfies

$$\begin{aligned} \dot{r}_{io} &= -v_o < 0, \\ \dot{\eta}_{io} &= 0. \end{aligned} \tag{14}$$

These two conditions characterize collision course between a moving point-object A_o and another moving or stationary point-object A_i.[26,29] Conditions in system (14) can be generalized to any point on the visible arc in A_i. The following result is a generalization to any visible point between p_{i1} and p_{i2}.

Proposition 1. A_i and A_o are in a collision course when

$$\begin{aligned} \dot{r}_{io} &< 0 \ \texttt{and} \\ \dot{\eta}_1 \cdot \dot{\eta}_2 &\leq 0. \end{aligned} \tag{15}$$

Proof. The goal is to prove that there is a visible point between p_{i1} and p_{i2} with a visibility angle between η_1 and η_2 for which system (14) is satisfied (these conditions are necessary and sufficient for collision). According to the mean value theorem, a negative value for the $\dot{\eta}_1 \times \dot{\eta}_2$ implies that there is a point for which $\dot{\eta} = 0$, with $\eta_1 < \eta < \eta_2$. The first condition implies a decreasing range. The case when $\dot{\eta}_1 \times \dot{\eta}_2 = 0$ implies that A_o is in a collision course with either p_{i1} or p_{i2}. □

It is possible to derive an expression for the visibility angles rates. By replacing $\dot{\eta}_1$ and $\dot{\eta}_2$ by their values, we get

$$\dot{\eta}_1 \times \dot{\eta}_2 = 0.5 \left(\frac{v_o^*}{r_e} \right)^2 [\cos(2\xi) - \cos(2(\theta_o^* - \eta_{io}))]. \tag{16}$$

This equation provides another way to describe the collision cone as defined in Eq. (6). The result from Proposition 1 can be formulated based on the visibility line angles and collision cone angle as follows.

Proposition 2. A_i and A_o are in a collision course when

$$\begin{aligned} \dot{r}_{io} &< 0 \ \texttt{and} \\ |\dot{\eta}_{io}| &\leq |\dot{\xi}|. \end{aligned} \tag{17}$$

Proof. Proposition 1 states that collision takes place when the range is decreasing and $\dot{\eta}_1 \times \dot{\eta}_2 \leq 0$. By replacing η_1 and η_2, by their values, we get $\dot{\eta}_{io}^2 \leq \dot{\xi}^2$, which yields $|\dot{\eta}_{io}| \leq |\dot{\xi}|$. □

This proposition interprets the collision risk in terms of the collision cone rate. The rate of the collision cone angle is given by

$$r_e \dot{\xi} = v_o^* \cos(\theta_o^* - \eta_{io}) \sin(\xi). \tag{18}$$

The following result provides an insight about the collision risk.

Proposition 3. There is no collision risk between A_o and $A_i(t_0)$ when

$$|\theta_o - \eta_{io}| \geq \pi/2. \tag{19}$$

Proof. As it can be seen from Eq. (13), the relative range is increasing or constant ($\dot{r}_{io} \geq 0$) when $|\theta_o - \eta_{io}| \geq \pi/2$. Clearly, there is no collision risk when the range is increasing or constant. □

This proposition allows to determine the sweeping volume intersection condition characterized by the domain of $\theta_o - \eta_{io}$ for which collision may take place. The following example illustrates Proposition 1.

Example 2. Consider the scenarios of Example 1. The time evolution of the rates of change of η_1 and η_2 as well as θ_o^* and the VCS collision cone are shown in Fig. 5. Figures 5(a) and (b) correspond to the top scenario in Fig. 4. Figures 5(c) and (d) correspond to the bottom scenario in Fig. 4. There is no collision in the bottom scenario as $\dot{\eta}_1 \times \dot{\eta}_2 > 0$ and $\theta_o^* \notin \mathcal{CC}$. In the top scenario, A_o and A_i are in collision course for $t \geq 5$ s when $\dot{\eta}_1 \times \dot{\eta}_2$ becomes negative as seen in Fig. 5(a). This can be confirmed in Fig. 5(b) where $\theta_o^* \in \mathcal{CC}$ for time $t \geq 5$ s.

4. Collision Detection: Probabilistic Model

This section discusses collision checking when A_i is stationary. In this case, the obstacle parameters vector is $m_i = (x_i, y_i, d_i)$, where d_i is the radius. The states and geometric variables are not exactly known. An extended Kalman filter is used to predict and update the relative state. Figure 6 shows the uncertainty in the location and the size of the obstacle, and the uncertainty in the orientation of the vehicle with $\hat{\theta}_o = \pi/3, \hat{\xi} = \pi/30, \hat{r}_{io} = 20, \hat{\eta}_{io} = \pi/4, (\hat{x}_o, \hat{y}_o) = (0, 0)$. We first study the probability of collision with a stationary object. Results are later

https://doi.org/10.1017/S0263574712000550 Published online by Cambridge University Press

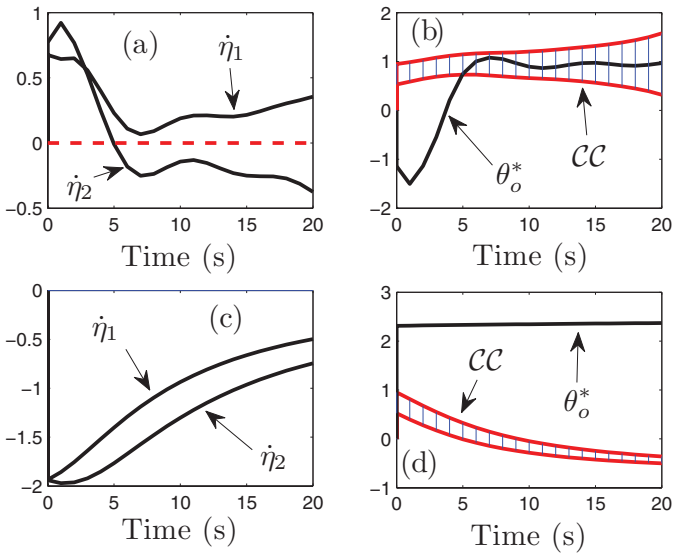


Fig. 5. (Colour online) Illustration of the rates of change of η_1 and η_2 and evolution of θ_o^* for the scenarios of Fig. 4. (a) Time evolution of $\dot{\eta}_1$ and $\dot{\eta}_2$ for the top scenario of Fig. 4, (b) time evolution of θ_o^* for the top scenario of Fig. 4, (c) time evolution of $\dot{\eta}_1$ and $\dot{\eta}_2$ for the bottom scenario of Fig. 4, (d) time evolution of θ_o^* for the bottom scenario of Fig. 4.

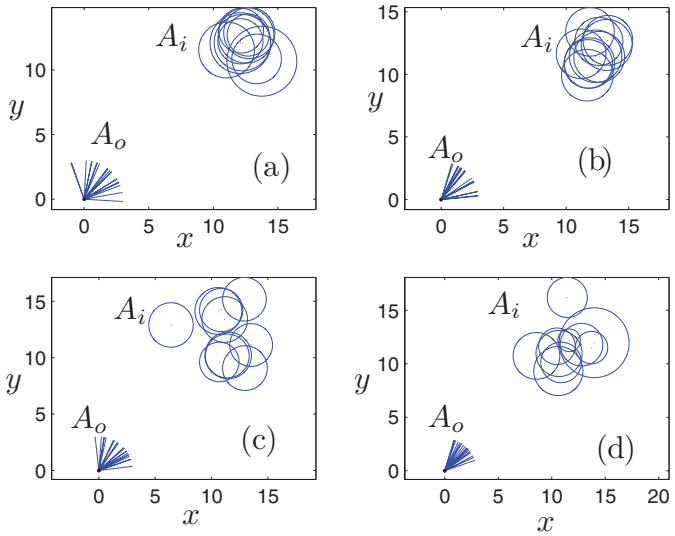


Fig. 6. (Colour online) A representation of uncertainty in the location and size of the obstacle and the orientation of the vehicle. The lines at A_o represent uncertainty in the orientation. The obstacle is represented by circles centered at its reference point. $\hat{\theta}_o = \pi/4$, $\hat{\xi} = 0.118$, $\hat{r}_{io} = 17$, $\hat{\eta}_{io} = \pi/4$. (a) $\sigma_{\theta_o} = \pi/6$, $\sigma_\xi = 0.03$, $\sigma_{r_{io}} = 0.8$, $\sigma_{\eta_{io}} = 0.047$; (b) $\sigma_{\theta_o} = \pi/6$, $\sigma_\xi = 0.008$, $\sigma_{r_{io}} = 0.8$, $\sigma_{\eta_{io}} = 0.047$; (c) $\sigma_{\theta_o} = \pi/10$, $\sigma_\xi = 0.015$, $\sigma_{r_{io}} = 1.8$, $\sigma_{\eta_{io}} = 0.1$; (d) $\sigma_{\theta_o} = \pi/10$, $\sigma_\xi = 0.03$, $\sigma_{r_{io}} = 1.8$, $\sigma_{\eta_{io}} = 0.11$.

generalized to moving objects. Accurate vehicle localization is important for safe navigation. Therefore, in addition to the parameters vector of the obstacle (location and size), it is important to obtain an accurate estimate of the state of the vehicle. A simple extended Kalman filter is used for this purpose.

4.1. Process model

In order to use Kalman filter, the kinematics model of motion are first discretized. The discrete process model has the general form

$$\begin{aligned} x(k+1) &= f(x(k), u(k)) + v(k), \\ v(k) &\sim N(0, Q(k)), \end{aligned} \tag{20}$$

where $f(x(k), u(k))$ is the state transition function. $v(k)$ is the process noise, assumed to be Gaussian with zero mean and covariance matrix $Q(k)$. The state transition function can be written in discrete form as

$$f(x(k), u(k)) = \begin{pmatrix} x(k) + T(k)\cos(\theta(k)) \\ y(k) + T(k)\sin(\theta(k)) \\ \theta(k) + \Delta\theta(k) \\ v(k) + \Delta v(k) \end{pmatrix}, \tag{21}$$

where $T(k)$ represents the translational displacement and $\Delta\theta(k)$ is the rotational displacement. In order to implement the extended Kalman filter, the state transition function is linearized about the mean values.

4.2. Obstacle and measurement models

The obstacle is a reflecting surface that is a part of the environment observed by the sensors such as sonar sensors. In our case, the observed surface is an arc between p_{i1} and p_{i2}. The measurement model has the following general form:

$$\begin{aligned} z(k) &= h(x(k), m_i(k)) + w(k), \\ w(k) &\sim N(0, R(k)). \end{aligned} \tag{22}$$

This measurement model relates the sensors' readings to the state of the vehicle and the parameter vector of the obstacle. The measurement model along with the process model are incorporated in an extended Kalman filter to estimate and update the state of the vehicle. Details about the extended Kalman filter can be found in the literature (e.g., in ref. [30]) and are beyond the scope of this paper.

The geometric quantities returned by the robot's sensory system are used to determine the obstacle's position and size (size that is visible to the sensory system). These measurements are used to obtain the relative position given by

$$\begin{aligned} x_i - x_o &= x_{io} = r_{io}\cos\eta_{io}, \\ y_i - y_o &= y_{io} = r_{io}\sin\eta_{io}. \end{aligned} \tag{23}$$

The relative coordinates x_{io}, y_{io} are characterized by covariance matrix with the following elements:[31]

$$\begin{aligned} \sigma_{x_{io}}^2 &= \sigma_{r_{io}}^2\cos^2(\hat{\eta}_{io}) + \hat{r}_{io}^2\sigma_{\eta_{io}}^2\sin^2(\hat{\eta}_{io}), \\ \sigma_{y_{io}}^2 &= \sigma_{r_{io}}^2\sin^2(\hat{\eta}_{io}) + \hat{r}_{io}^2\sigma_{\eta_{io}}^2\cos^2(\hat{\eta}_{io}), \\ \sigma_{(xy)_{io}} &= \left[\sigma_{r_{io}}^2 - \hat{r}_{io}^2\sigma_{\eta_{io}}^2\right]\sin(\hat{\eta}_{io})\cos(\hat{\eta}_{io}). \end{aligned} \tag{24}$$

The coordinates of the outmost points p_{i1} and p_{i2} are deduced in a similar way. Their covariance matrices are characterized by systems similar to system (24). The uncertainties in these positions allow to deduce the conflict region.

https://doi.org/10.1017/S0263574712000550 Published online by Cambridge University Press

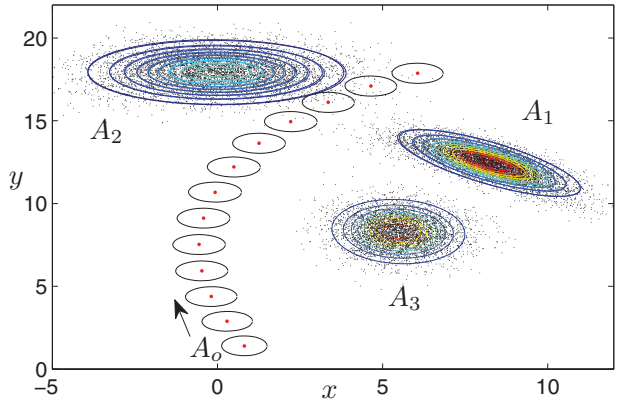


Fig. 7. (Colour online) Representation of the uncertainty in the position of the vehicle and the obstacles.

A representation of the uncertainty in the position of the obstacles and the vehicle is shown in Fig. 7 where an extended Kalman filter is used to predict and update the state of the vehicle.

4.3. Probabilistic geometric collision cone

It is possible to construct a geometric collision cone when A_i is stationary as shown in Fig. 2. The mean value of ν is $\hat{\nu} = [\hat{\eta}_{io}, \hat{\xi}]^T$. Its covariance matrix is denoted by C_ν. We have

$$\begin{aligned} \hat{\eta} &= T_1 \hat{\nu}, \\ C_\eta &= T_1 C_\nu T_1^T, \end{aligned} \tag{25}$$

where C_η is the covariance matrix of vector η. Vector η also has a Gaussian distribution. Vectors ν and η constitute important characteristics of the obstacle.

The geometric collision cone $\mathcal{CC}$ has width 2ξ and is centered at the line of sight $\vec{L}_{io}$. This is a probabilistic collision cone since it depends on the sensory readings. Collision takes place when $\theta_o \in \mathcal{CC}$. Recall that the direction of A_i with respect to A_o is characterized by η_{io}. While $\eta_{io} \in [0, 2\pi]$, the values of ξ are restricted to the interval $\xi \in [0, \pi/2]$, with the following limit cases:

- $\xi = 0$ when both A_i and A_o are geometric points.
- $\xi = \pi/2$ when A_o is tangent to A_i. This is the case when A_i has an infinite size and thus $p_{i1}, p_{i2} \longrightarrow \pm\infty$.

The probability of collision is then given by

$$\begin{aligned} P_{\text{col}} = P(\theta_o \in \mathcal{CC}) = P(\theta_o - \eta_{io} + \xi \geq 0, \theta_o \\ - \eta_{io} - \xi \leq 0). \end{aligned} \tag{26}$$

We define δ as $\delta = \theta_o - \eta_{io}$. δ has a Gaussian distribution with $\hat{\delta} = \hat{\theta}_o - \hat{\eta}_{io}$ and $\sigma_\delta^2 = \sigma_{\theta_o}^2 + \sigma_{\eta_{io}}^2$. Now we define vector $\mu = [\delta, \xi]^T$, which has mean value $\hat{\mu} = [\hat{\delta}, \hat{\xi}]^T$ and covariance matrix C_μ. We also define a new vector of variables $\beta = [\beta_1, \beta_2]^T$, with $\beta_1 = \delta + \xi$ and $\beta_2 = \delta - \xi$. The relationship between the two vectors under matrix form can be written as

$$\beta = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} \delta \\ \xi \end{bmatrix} = T_2 \begin{bmatrix} \delta \\ \xi \end{bmatrix}. \tag{27}$$

The mean values and the covariance matrix of vector β are given by

$$\begin{aligned} \hat{\beta} &= T_2 \hat{\mu}, \\ C_\beta &= T_2 C_\mu T_2^T. \end{aligned} \tag{28}$$

The intervals for β_1 and β_2 are given by $\beta_1 \in [-\pi/2, \pi]$ and $\beta_2 \in [\pi, \pi/2]$ respectively. Under the new formulation, the probability of collision is given by

$$P_{\text{col}} = P(\theta_o \in \mathcal{CC}) = P(\beta_1 \geq 0, \beta_2 \leq 0). \tag{29}$$

The joint probability density function of β_1 and β_2 is given by

$$f_{\beta_1,\beta_2}(\beta_1, \beta_2) = \frac{1}{2\pi\sqrt{|C_\beta|}} e^{-\frac{1}{2}(\beta-\hat{\beta})^T C_\beta (\beta-\hat{\beta})}. \tag{30}$$

This probability density function characterizes the probabilistic collision cone. The probability of collision is given by

$$P_{\text{col}} = \int\int_D f_{\beta_1,\beta_2}(\beta_1, \beta_2) d\beta_1 d\beta_2, \tag{31}$$

where D is the domain for β_1 and β_2. Thus, the probability of collision can be written as follows:

$$P_{\text{col}} = \int_{-\pi}^{0} \int_{0}^{\pi} f_{\beta_1,\beta_2}(\beta_1, \beta_2) d\beta_1 d\beta_2. \tag{32}$$

A simple numerical integration is used to estimate this probability. Its value is then transferred to the path planner, which will correct the path whenever $P_{\text{col}} > P_{\text{th}}$.

Now our goal is to generalize the previous result to moving objects. Two approaches are used: (1) The VCS, where an extended version of the geometric collision cone is used, and (2) rates of change of the visibility angles.

4.4. Collision checking using the VCS: the probabilistic approach

Collision between A_o and A_i is reduced to collision between A_o^* and $A_i(t_0)$. Collision takes place when $\theta_o^* \in \mathcal{CC}$. While θ_o (orientation angle of A_o) is obtained from sensory measurement, θ_o^* is obtained using Eq. (9). Thus, the density function of θ_o^* depends on those of the velocities and orientations of A_i and A_o. The assumption of Gaussian distribution for sensory measurement implies that $\dot{x}_{io}$ and $\dot{y}_{io}$ have Gaussian distributions. The calculation of the distribution of θ_o^* is done in the following three steps:

- Find the distributions of $\dot{x}_{io}$ and $\dot{y}_{io}$ from the distributions of $\dot{x}_o$, $\dot{y}_o$, $\dot{x}_i$ and $\dot{y}_i$.
- Find the distribution of the ratio of the relative velocities $q = \dot{y}_{io}/\dot{x}_{io}$.
- Find the distribution of the inverse tangent function.

It is assumed that $(\dot{x}_i, \dot{y}_i)$ and $(\dot{x}_o, \dot{y}_o)$ are independent. The probability density functions for $\dot{x}_{io}$ and $\dot{y}_{io}$ are given by

$$\begin{aligned} f_{\dot{X}_{io}}(\dot{x}_{io}) &= \textstyle\int_{-\infty}^{\infty} f_{\dot{X}_i}(\dot{x}_{io} + \dot{x}_o) f_{\dot{X}_o}(\dot{x}_o) d\dot{x}_o, \\ f_{\dot{X}_{io}}(\dot{x}_{io}) &= f_{\dot{X}_i}(-\dot{x}_{io}) \bigotimes f_{\dot{X}_o}(\dot{x}_o), \end{aligned} \tag{33}$$

https://doi.org/10.1017/S0263574712000550 Published online by Cambridge University Press

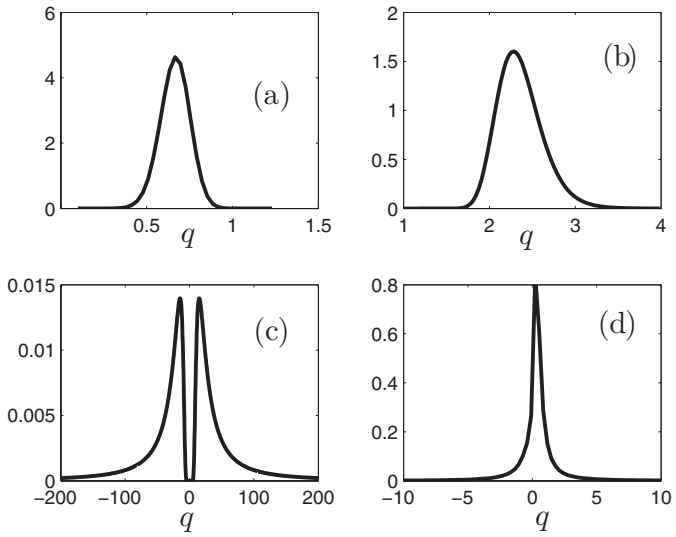


Fig. 8. Illustration of the probability distribution function of the velocity ratio q for different mean values and covariance matrices: (a) $\hat{\dot{y}}_{io} = 2$, $\hat{\dot{x}}_{io} = 3$, $\sigma_{\dot{y}_{io}} = 1/3$, $\sigma_{\dot{y}_{io}} = 1/5$, $\rho = 0.7$; (b) $\hat{\dot{y}}_{io} = 7$, $\hat{\dot{x}}_{io} = 3$, $\sigma_{\dot{y}_{io}} = 5$, $\sigma_{\dot{y}_{io}} = 14$, $\rho = 0.2$; (c) $\hat{\dot{y}}_{io} = 7$, $\hat{\dot{x}}_{io} = 0$, $\sigma_{\dot{y}_{io}} = 1/5$, $\sigma_{\dot{y}_{io}} = 1/3$, $\rho = 0.2$; (d) $\hat{\dot{y}}_{io} = 0$, $\hat{\dot{x}}_{io} = 7$, $\sigma_{\dot{y}_{io}} = 1/5$, $\sigma_{\dot{y}_{io}} = 1/3$, $\rho = 0.2$.

and

$$\begin{aligned} f_{\dot{Y}_{io}}(\dot{y}_{io}) &= \int_{-\infty}^{\infty} f_{\dot{Y}_i}(\dot{y}_{io} + \dot{y}_o) f_{\dot{Y}_o}(\dot{y}_o) d\dot{y}_o, \\ f_{\dot{Y}_{io}}(\dot{y}_{io}) &= f_{\dot{Y}_i}(-\dot{y}_{io}) \bigotimes f_{\dot{Y}_o}(\dot{y}_o), \end{aligned} \tag{34}$$

where $\bigotimes$ denotes convolution. It can be proved that the mean values and variances are given by

$$\begin{aligned} \hat{\dot{x}}_{io} &= \hat{\dot{x}}_i - \hat{\dot{x}}_o, \\ \hat{\dot{y}}_{io} &= \hat{\dot{y}}_i - \hat{\dot{y}}_o, \\ \sigma^2_{\dot{x}_{io}} &= \sigma^2_{\dot{x}_i} + \sigma^2_{\dot{x}_o}, \\ \sigma^2_{\dot{y}_{io}} &= \sigma^2_{\dot{y}_i} + \sigma^2_{\dot{y}_o}. \end{aligned} \tag{35}$$

The distribution of q is given by

$$f_Q(q) = \int_{-\infty}^{\infty} |\dot{x}_{io}| f_{\dot{X}_{io}\dot{Y}_{io}}(\dot{x}_{io}, q\dot{x}_{io}) d\dot{x}_{io}. \tag{36}$$

The distribution of the quotient of two Gaussian variables has been studied by many researchers.[32,33] In general, the closed form solution of the distribution is complex. Interested readers are referred to refs. [32, 33]. When both variables have zero mean values, q has a Cauchy distribution. The distribution of the relative velocities ratio is shown in Fig. 8. It is clear that except in the case of Fig. 8(c), the distribution of q can be approximated by a Gaussian distribution. For example, the case of Fig. 8(a) can be approximated by a Gaussian distribution with $\hat{q} = 0.667$ and $\sigma_q = 0.084$.

The distribution of θ_o^* is deduced from $f_Q(q)$ as follows:

$$f_{\theta_o^*}(\theta_o^*) = f_Q(\tan(\theta_o^*))(1 + \tan(\theta_o^*))^2. \tag{37}$$

An illustration of the distribution of θ_o^* is shown in Fig. 9. Left side is obtained using Eq. (37). Right side is obtained using Monte Carlo simulation. Comparison shows similarity between the result of Eq. (37) and Monte Carlo simulation.

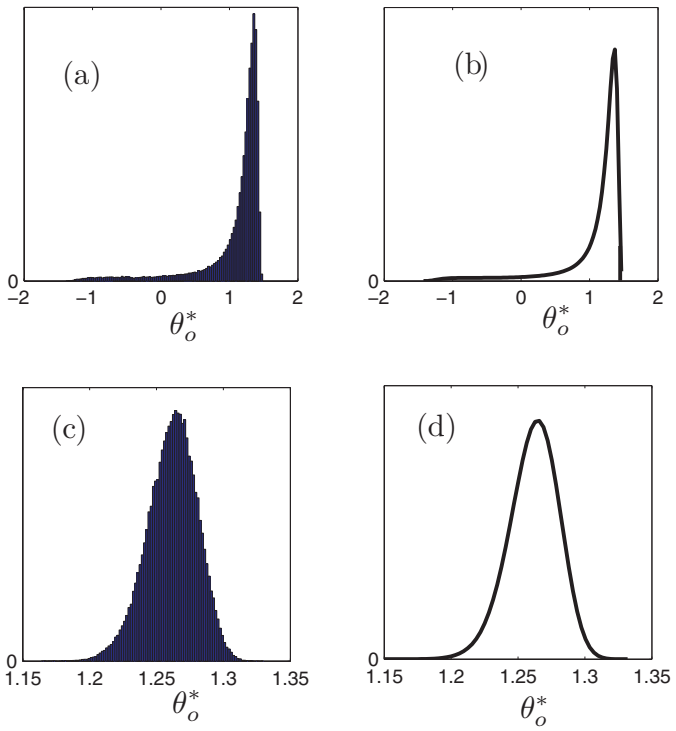


Fig. 9. (Colour online) Illustration of the probability distribution function of θ_o^* using system (37) and Monte Carlo simulation. Standard deviation is $\sigma_q = 2$ for top figures, and $\sigma_q = 0.2$ for bottom figures. For both cases $\hat{q} = \pi$.

We put $\delta^* = \theta_o^* - \eta_{io}$, and we define vectors $\mu^* = [\delta^*, \xi]^T$ and $\beta^* = [\beta_1^*, \beta_2^*]^T$, with $\beta_1^* = \delta^* + \xi$ and $\beta_2^* = \delta^* - \xi$. We write under matrix form

$$\beta^* = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} \delta^* \\ \xi \end{bmatrix}. \tag{38}$$

Under this formulation, the probability of collision is given by

$$P_{\text{col}} = P(\theta_o^* \in \mathcal{CC}) = P(\beta_1^* \geq 0, \beta_2^* \leq 0) \tag{39}$$

System (38) involves subtraction of random variables of different distributions. General formulas can be used to obtain the distribution $f_{\beta^*}(\beta^*)$ of β^*. However, it is difficult to obtain the closed form solution for P_{col}. Numerical integration is used.

4.5. Collision checking using the visibility angles: the probabilistic approach

Under this formulation, the probability of collision is deduced as follows:

$$P_{\text{col}} = P(\dot{\eta}_1 \times \dot{\eta}_2 \leq 0, \dot{r}_{io} < 0) \tag{40}$$

with

$$\begin{aligned} P(\dot{\eta}_1 \times \dot{\eta}_2 \leq 0) = P(\dot{\eta}_1 \geq 0, \dot{\eta}_2 \leq 0) \\ + P(\dot{\eta}_1 \leq 0, \dot{\eta}_2 \geq 0). \end{aligned} \tag{41}$$

https://doi.org/10.1017/S0263574712000550 Published online by Cambridge University Press

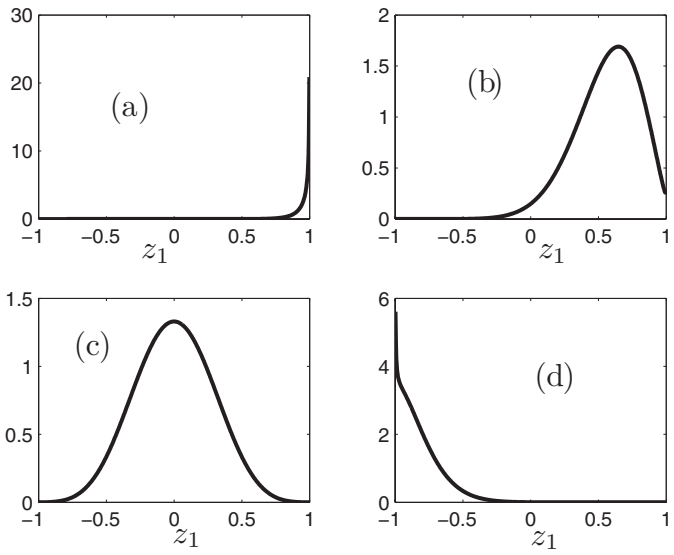


Fig. 10. Illustration of the distribution of z_1 for $\sigma_{\beta_1} = 0.3$ and different values of $\hat{\beta}_1$: (a) $\hat{\beta}_1 = \pi/2$, (b) $\hat{\beta}_1 = \pi/5$, (c) $\hat{\beta}_1 = 0$, (d) $\hat{\beta}_1 = -\pi/3$.

Since $\dot{r}_{io}$ is independent from $\dot{\eta}_1$ and $\dot{\eta}_2$, P_{col} can be written as

$$P_{\text{col}} = P(\dot{\eta}_1 \times \dot{\eta}_2 \leq 0) P(\dot{r}_{io} < 0). \tag{42}$$

Recall that no collision takes place when $P(\dot{r}_{io} < 0) = 0$. The evolution of $\dot{\eta}_1$ and $\dot{\eta}_2$ is given by systems (11) and (12). Since the range and speed are always positive, it is recommended to replace $(\dot{\eta}_1, \dot{\eta}_2)$ by a new vector (z_1, z_2) defined as follows:

$$\begin{aligned} z_1 &= -\frac{r_e \dot{\eta}_1}{v_o^*} = \sin(\beta_1^*), \\ z_2 &= -\frac{r_e \dot{\eta}_2}{v_o^*} = \sin(\beta_2^*). \end{aligned} \tag{43}$$

The marginal distribution for z_1 is given by

$$f_{Z_1}(z_1) = f_{\beta_1}(\sin^{-1}(z_1)) \frac{1}{\sqrt{1 - z_1^2}}. \tag{44}$$

This distribution is shown in Fig. 10 for different values of $\hat{\beta}_1$ with $\sigma_{\beta_1} = 0.3$. $f_{Z_1}(z_1)$ has a Gaussian distribution only when $\hat{\beta}_1 = 0$. z_2 follows a similar marginal distribution. The joint distribution of $z = [z_1, z_2]^T$ is given by

$$f_Z(z) = f_S(g^{-1}(z)) |J(z)|, \tag{45}$$

where $S = [\delta, \xi]^T$, $g = [\sin(\delta - \xi), \sin(\delta + \xi)]^T$, and J is the Jacobian of g. The distribution for z is shown in Fig. 11 for different mean values. The contour is shown in Fig. 12. It is also possible to obtain the distribution of the product $z_1 \times z_2$ and deduce the probability of collision from there. We have

$$z_1 \times z_2 = 0.5(\cos(2\xi) - \cos(2\delta)). \tag{46}$$

The other way to obtain $\dot{\eta}_1$ and $\dot{\eta}_2$ and their distributions is to use η_1 and η_2 as obtained from the sensory system with the definition of the time derivative definition

$$\dot{\eta}_{1,2} = \frac{\eta_{1,2}(k+1) - \eta_{1,2}(k)}{T}, \tag{47}$$

where T is the sampling period and $\eta_{1,2}(k)$ is the state at time kT ($k = 0, 1, 2...$). This approach relies directly on the information from the sensory system and does not need to estimate the orientation of the vehicles.

4.6. Comparison summary

Both the rate of change of the visibility angles and the VCS present simple tests that allow to check for possible collision in the presence of uncertainties. Table 4.5 shows a

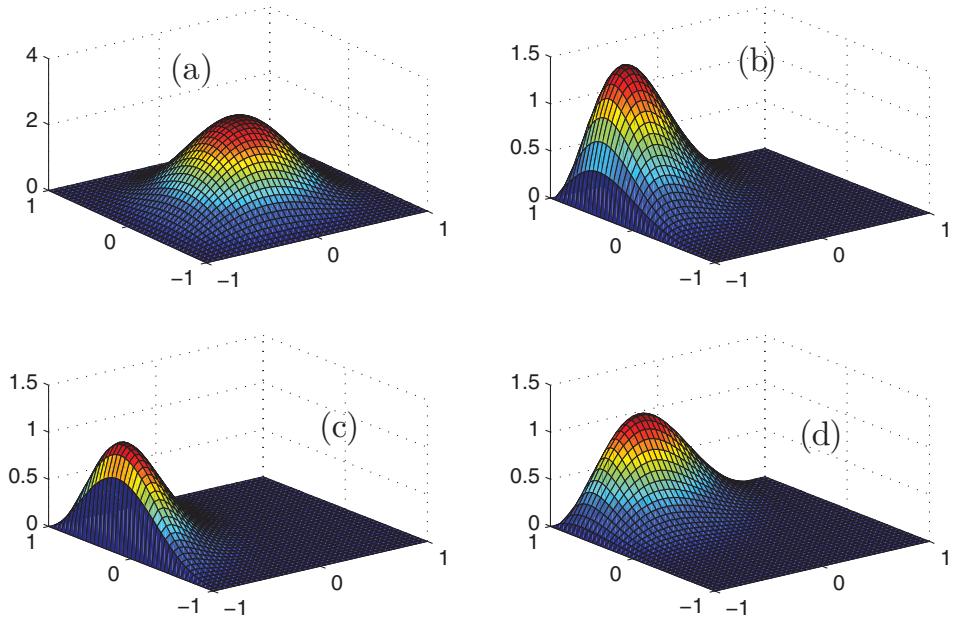


Fig. 11. (Colour online) Illustration of the distribution of z for $\sigma_\delta = 0.42$, $\sigma_\xi = 0.3$, and different mean values: (a) $(\hat{\delta}, \hat{\xi}) = (0, 0)$, (b) $(\hat{\delta}, \hat{\xi}) = (0, \pi/4)$, (c) $(\hat{\delta}, \hat{\xi}) = (-\pi/6, \pi/4)$, (d) $(\hat{\delta}, \hat{\xi}) = (\pi/6, \pi/4)$.

https://doi.org/10.1017/S0263574712000550 Published online by Cambridge University Press

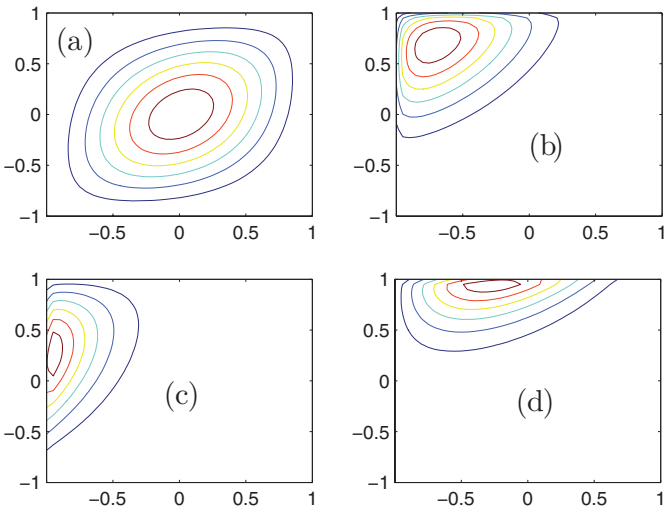


Fig. 12. (Colour online) Contour representing the distribution of z for different mean values: (a) $(\hat{\delta}, \hat{\xi}) = (0, 0)$, (b) $(\hat{\delta}, \hat{\xi}) = (0, \pi/4)$, (c) $(\hat{\delta}, \hat{\xi}) = (-\pi/6, \pi/4)$, (d) $(\hat{\delta}, \hat{\xi}) = (\pi/6, \pi/4)$.

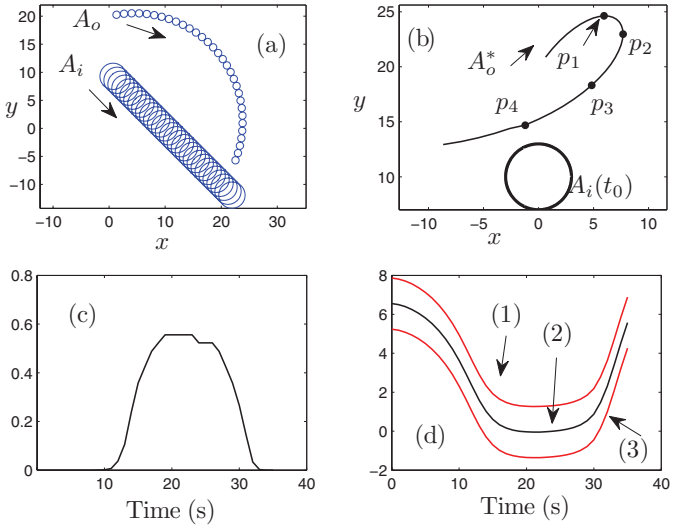


Fig. 13. (Colour online) An illustration of the probability of collision when the vehicle of interest moves with time-varying orientation: (a) Real scenario; (b) VCS scenario, p_1, p_2, p_3, p_4 represent the location of A_o^* at times 10,17,25, 33 s respectively; (c) time evolution probability of collision; (d) the evolution of the product $z_1 \times z_2$.

summary comparison between the two approaches. Perhaps the most important difference between the two methods lies in the calculation of the probability of collision. The VCS approach involves more steps and requires more variables such as the directions and the speeds of the vehicles. Thus, checking for collision using the rate of change of the visibility angles is simpler than the VCS (because of the error propagation and the estimation of different distributions). On the other side, the VCS is more adequate for motion planning in the presence of uncertainties, since it directly uses the orientations and the speeds. Example 2 showed a comparison between the two approaches in the deterministic case. Example 6 in the simulation section shows a numerical comparison.

5. Simulation

The simulation section illustrates various scenarios for different paths and uncertainty levels. We calculate and compare the probability of collision for the vehicle of interest with other vehicles. All units of the variables are SI units unless mentioned otherwise.

Example 3. The first example in the simulation section shows a scenario where the vehicle of interest moves with time-varying orientation angle. The initial values for the state vectors of A_o and A_i are given by $\mathbf{x}_o = [0, 20, 0.2, 1.35]^T$ and $\mathbf{x}_i = [0, 10, -\pi/4, 1]^T$ respectively. The augmented radius of A_i is $d_i = 3$. The scenario is shown in Fig. 13. Figure 13(a) shows the real plane. Figure 13(b) shows the VCS. p_1, p_2, p_3, p_4 represent the location of A_o^* at times 10, 17, 25, 33 s respectively. Figure 13(c) shows the probability of collision, which is obtained using the rate of change of the visibility angles in this example. Initially the probability of collision is zero (in the time interval [0, 10]). This can be easily verified in the VCS, where A_o^* is heading in the opposite direction of $A_i(t_0)$. In the time interval [10, 17] the probability of collision increases as A_o^* begins turning toward $A_i(t_0)$. The probability of collision remains flat in the time interval [17, 25], and then begins decreasing. The probability of collision reaches a zero value at time 33 s. Figure 13(d) shows the time evolution of the mean value of the product $z_1 \times z_2$ (middle curve). The upper curve shows the mean value of the product $(z_1 \times z_2)$, plus $\sigma_{z_1 \times z_2}$ and the lower curve shows mean values of the product $(z_1 \times z_2)$, minus $\sigma_{z_1 \times z_2}$.

Example 4. The second example in this section is shown in Fig. 14. The initial values for the state vectors

Table I. Summary of comparison between the rate of change of the visibility angles and the VCS.

	Variables used	Steps	Probability of collision	Motion planning
Rate of change of the visibility angles	Visibility angles.	Deduce the rates of the visibility angles.	Directly calculated from the distribution of $\dot{\eta}_1$ and $\dot{\eta}_2$.	Cannot be used directly for motion planning.
VCS	Speed and orientation of A_o and A_i and visibility angles to construct the collision cone.	Deduce the orientation θ_o^* and the collision cone $\mathcal{CC}$.	Calculated based on the probabilistic collision cone and the distribution of θ_o^*.	Can be used for motion planning.

https://doi.org/10.1017/S0263574712000550 Published online by Cambridge University Press

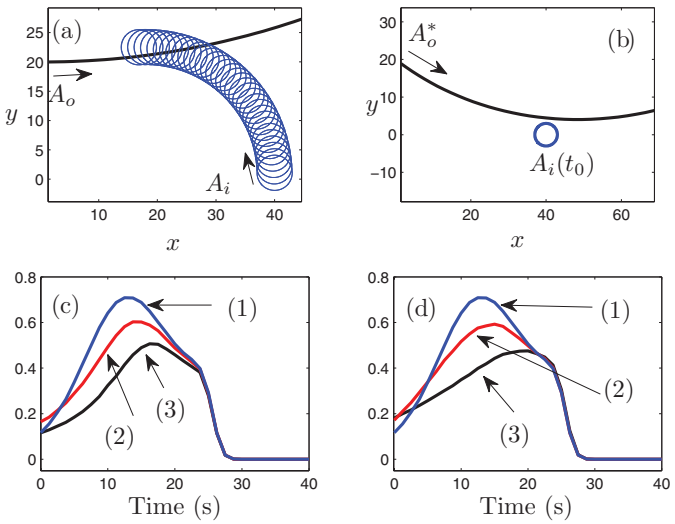


Fig. 14. (Colour online) An illustration of the uncertainty in the obstacle motion: (a) Real scenario; (b) VCS scenario; (c) the probability of collision for $\sigma_{v_i} = 0.1$ and different values of σ_{θ_i}, curve (1): $\sigma_{\theta_i} = 0.0873$; curve (2): $\sigma_{\theta_i} = 0.2618$; curve (3): $\sigma_{\theta_i} = 0.5236$; (d) the probability of collision for $\sigma_{\theta_i} = 0.0873$ and different values of σ_{v_i}, curve (1): $\sigma_{v_i} = 0.1$; curve (2): $\sigma_{v_i} = 0.3$; curve (3): $\sigma_{v_i} = 0.6^{\circ}$.

of A_o and A_i are given by $\mathbf{x}_o = [0, 20, 0, 1.1]^T$ and $\mathbf{x}_i = [40, 0, \pi/2, 0.88]^T$ respectively. The probability of collision is studied for the same scenario but for different uncertainties in the motion of A_i. The probability is calculated using the VCS in this example. The scenario is shown in Fig. 14(a). The corresponding VCS is shown in Fig. 14(b). The probability of collision is shown in Figs. 14(c) and (d). Figure 14(c) shows the probability of collision for $\sigma_{v_i} = 0.1$ and different values of σ_{θ_i}, curve (1): $\sigma_{\theta_i} = 0.0873$; curve (2): $\sigma_{\theta_i} = 0.2618$; curve (3): $\sigma_{\theta_i} = 0.5236$. Figure 14(d) shows the probability of collision for $\sigma_{\theta_i} = 0.0873$ and different values of σ_{v_i}, curve (1): $\sigma_{v_i} = 0.1$; curve (2): $\sigma_{v_i} = 0.3$; curve (3): $\sigma_{v_i} = 0.6$. Figures 14(c) and (d) show that the uncertainty level in the speed or orientation affects the probability of collision, that is the probability of collision varies with the uncertainty level in the motion of A_i. Also, all probability curves have the same pattern. For all curves, the probability of collision reaches zero at time 29 s approximately.

Example 5. This scenario shows the case where the vehicle of interest has a collision risk with multiple vehicles. The probability of collision is calculated using the VCS in this example. The initial values for the state vectors of A_o and $A_i, i = 1, \ldots, 4$ are given by $\mathbf{x}_o = [40, 40, -\pi/2, 1.35]^T$, and $\mathbf{x}_1 = [20, 0, \pi/2, 1]^T$, $\mathbf{x}_2 = [0, 20, 0, 1]^T$, $\mathbf{x}_3 = [40, -20, \pi/2, 1]^T$, and $\mathbf{x}_4 = [60, 10, -0.2, 1]^T$. The vehicle of interest calculates the probability of collision with each vehicle. In this simulation, there are four vehicles in addition to the vehicle of interest. The real scenario is shown in Fig. 15. The VCS scenario is shown in Fig. 16. The VCS formulation is slightly different from the previous one. Here A_o is reduced to a stationary point $A_o(t_0)$ instead of A_i, and A_o is augmented in

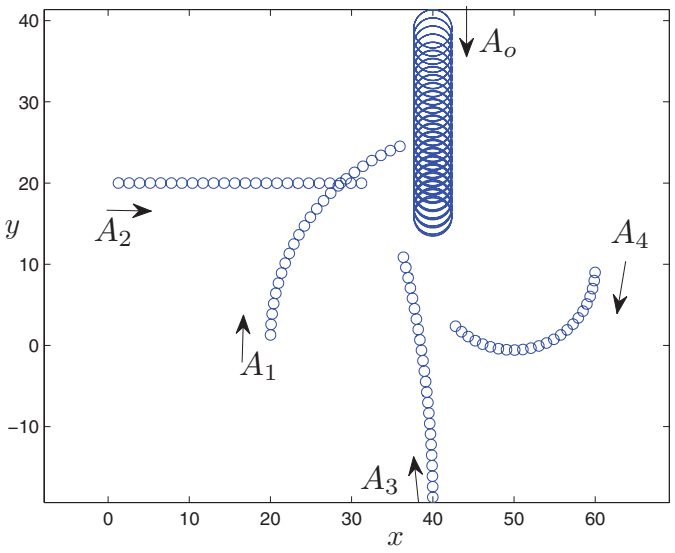


Fig. 15. (Colour online) Checking collision with multiple vehicles: real scenario.

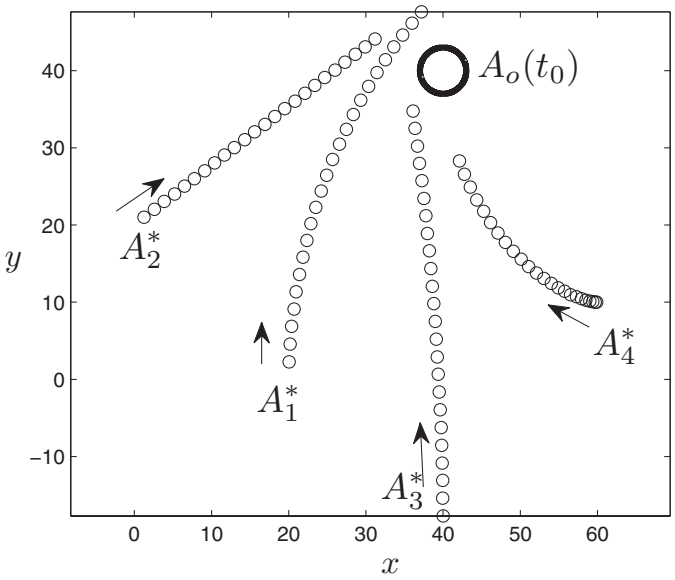


Fig. 16. Checking collision with multiple vehicles: VCS scenario.

the size instead of A_i. The probability of collision is shown in Fig. 17. The probability of collision with A_1, A_2, and A_3 decreases with time. The probability of collision with A_4 increases with time and reaches 0.5 at the final time. In the presence of multiple vehicles, the vehicle of interest needs to minimize the instantaneous probability of collision with all vehicles in its neighborhood. This can be solved as an optimization problem in the VCS.

Example 6. This example has two scenarios that are shown in Figs. 18(a) and (c). The equivalent VCS scenarios are shown in Figs. 18(b) and (d). The initial state for A_o and A_i are $[60, 15, -\pi/2, 1.1]^T$ and $[40, 40, -\pi/2, 1]^T$ in both cases. The only difference between the two scenarios is ω_i, which is $\omega_i = 0$ rad/s in Fig. 18(a) and $\omega_i = 0.05$ rad/s for Fig. 18(b). In both cases $\omega_o = -0.1$ rad/s. The time evolution of the probability of collision is shown in Fig. 19. In the

https://doi.org/10.1017/S0263574712000550 Published online by Cambridge University Press

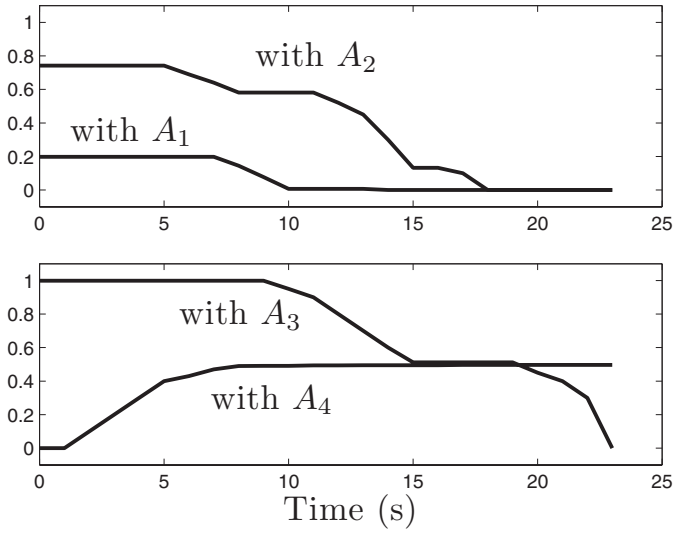


Fig. 17. Probability of collision with multiple vehicles.

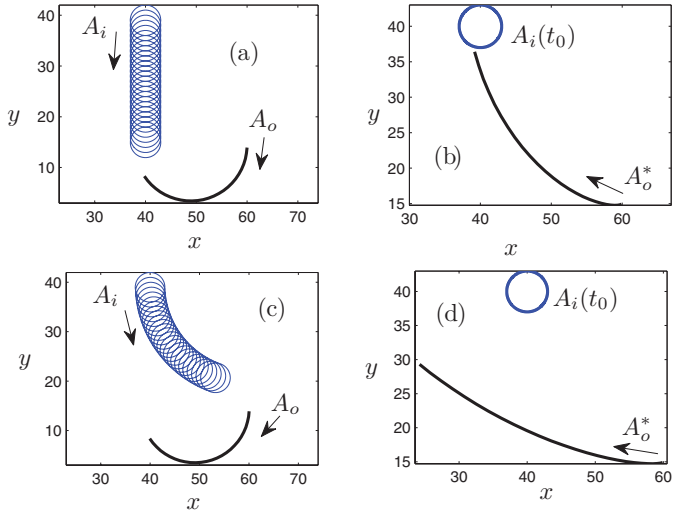


Fig. 18. (Colour online) Illustration for Example 6: (a) 1st scenario of Example 6, (b) VCS for the first scenario, (c) 2nd scenario of Example 6, (d) VCS for the first scenario.

first scenario, P_{col} is an increasing function. In the second scenario, P_{col} is a decreasing function and reaches zero in about 16 s. A comparison between different curves shows a slight difference between the probabilities obtained using the two approaches. The difference tends to be smaller values with time. The difference in the probability of collision is mainly due to the numerical calculations, the steps used to calculate the probability of collision, and the set of variables being used.

6. Conclusion

This paper presents new approaches for probabilistic collision risk assessment between moving vehicles. Uncertainties in motions are taken into account, where measured variables are assumed to follow a Gaussian distribution. Because of the nonlinear nature of kinematic equations, the output variables do not necessarily follow Gaussian distribution. The suggested solutions are based on

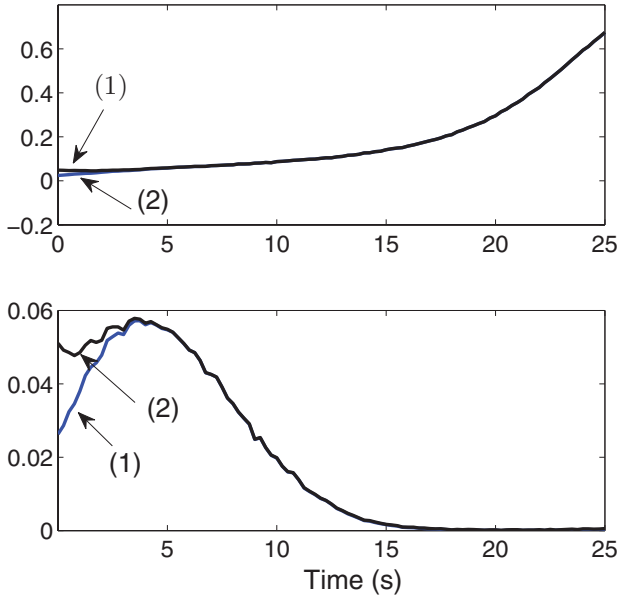


Fig. 19. (Colour online) Probability of collision for the scenarios of Example 6. Top: Probability of collision for the scenario of Fig. 18(a). Bottom: Probability of collision for the scenario of Fig. 18(c).

the rate of change of the visibility angles and the VCS. Error propagation is considered when calculating the probability of collision. An advantage of the visibility angle rates lies in the ability of the method to use the visibility angles that can be directly obtained from the sensory system. The VCS on the other side has a few more steps, which make it more prone to numerical approximation errors when calculating the probability of collision. However, the VCS has a better ability to solve the motion planning problem since it constructs probability distributions for the orientation and the speed of the vehicle. The next step in this research is to use the approaches discussed here for motion planning in a highly dynamic environment in the presence of uncertainties.

References

1. H. Blom and G. Bakker, "Conflict Probability and Incrossing Probability in Air Traffic Management," **In**: *Proceedings of the IEEE International Conference on Decision and Control*, Las Vegas (Dec. 2002) pp. 2421–2426.
2. R. Irvine, "A simplified approach to conflict probability estimation," *EEC Technical/Scientific Report No. 2001-017*, (EUROCONTROL Exp. Centre, Bretigny-sur-Orge, France, May 2001).
3. A. Bashllari, N. Kaciroti, D. Nace and A. Fundo, "Conflict Probability Estimations Based on Geometrical and Bayesian Approaches," **In**: *Proceedings of the IEEE Intelligent Transportation Systems Conference*, Seattle, WA (Sep. 2007) pp. 479–484.
4. A. Vela, E. Salaun, S. Solak, E. Feron, W. Singhose and J. Clarke, "A Two-Stage Stochastic Optimization Model for Air Traffic Conflict Resolution Under Wind Uncertainty," **In**: *Proceedings of the IEEE/AIAA 28th Digital Avionics Systems Conference*, Orlando, FL (Oct. 2009) pp. 2E5-1–2E5-13.
5. M. Prandini, J. Lygeros, A. Nilim and S. Sastry, "Randomized Algorithms for Probabilistic Aircraft Conflict Detection," **In**: *Proceedings of the IEEE International Conference on Decision and Control*, Phoenix, AZ (Dec. 1999) pp. 2444–2450.
6. J. Shen, N. H. McClamroch and E. G. Gilbert, "A Computational Approach to Conflict Detection Problems for

https://doi.org/10.1017/S0263574712000550 Published online by Cambridge University Press

Air Traffic Control," **In**: *Proceedings of the American Control Conference*, San Diego, CA (Jun. 1999) pp. 1445–1449.
7. A. Lambert, D. Gruyer and G. S. Pierre, "A fast Monte Carlo algorithm for collision probability estimation," **In**: *Proceedings of the 10th International Conference on Control, Automation, Robotics and Vision*, Hanoi, Vietnam (Dec. 2008) pp. 406–411.
8. J. Janssona and F. Gustafssonb, "A framework and automotive application of collision avoidance decision-making," *Automatica* **44**(9), 2347–2351 (2008).
9. N. Achour, N. Msirdi and R. Toumi, "Reactive Path Planning with Collision Avoidance in Dynamic Path Planning," **In**: *Proc.eedings of the IEEE International Workshop on Robot and Human Interactive Communication*, Berlin, Germany (Sep. 2001) pp. 62–67.
10. C. Fulgenzi, A. Spalanzani and C. Laugier, "Dynamic Obstacle Avoidance in Uncertain Environment Combining PVOs and Occupancy Grid," **In**: *Proceedings of the IEEE Intelligent Conference on Robotics and Automation*, Rome (Apr. 2007) pp. 1610–1616. [Online]. Available at: http://emotion.inrialpes.fr/bibemotion/2007/FSL07.
11. L. Blackmore, H. Li and B. Williams, "A Probabilistic Approach to Optimal Robust Path Planning with Obstacle," **In**: *Proceedings of the American Control Conference*, Minneapolis, MN (Jun. 2006) pp. 2831–2837.
12. P. Burlina, D. DeMenthon and L. Davis, "Navigation with Uncertainty: Reaching a Goal in a High Collision Risk Region," **In**: *Proceedings of the IEEE International Conference on Robotics and Automation*, Nice, France (May 1992) pp. 2440–2446.
13. P. Payeur, "Improving Robot Path Planning Efficiency with Probabilistic Virtual Environment Models," **In**: *Proceedings of the IEEE International Conference on Virtual Environments, Human-Computer Interfaces and Measurement Systems*, Boston, MA (Jul. 2004) pp. 13–18.
14. M. Juny and R. D'Andrea, "Probability Map Building of Uncertain Dynamic Environments with Indistinguishable Obstacles," **In**: *Proceedings of the American Control Conference*, Denver, CO (Jun. 2003) pp. 3417–3422.
15. A. Lambert, D. Gruyer, G. S. Pierre and A. Ndjent, "Collision Probability Assessment for Speed Control," **In**: *Proceedings of IEEE International Conference on Intelligent Transportation Systems*, Beijing, China (Oct. 2008) pp. 1043–1046.
16. A. Vahidi and A. Eskandarian, "Research advances in intelligent collision avoidance and adaptive cruise control," *IEEE Trans. Intell. Transp. Syst.* **4**(3), 143–153 (2003).
17. F. Wang, M. Yang and R. Yang, "Conflict-probability-estimation-based overtaking for intelligent vehicles," *IEEE Trans. Intell. Transp. Syst.* **10**(2), 366–370 (2009).
18. C. E. van Daalen and T. Jones, "Fast conflict detection using probability flow," *Automatica* **45**(8), 1903–1909 (2009).
19. M. Althoff, O. Stursberg and M. Buss, "Model-based probabilistic collision detection in autonomous driving," *IEEE Trans. Intell. Transp. Syst.* **10**(2), 29–310 (2009).
20. M. Althoff, O. Stursberg and M. Buss, "Verification of Uncertain Embedded Systems by Computing Reachable Sets Based on Zonotopes," **In**: *Proceedings of IFAC World Congress*, Coex, South Korea (Jul. 2008) pp. 5125–5130.
21. C. Schmidt, F. Oechsle and W. Branz, "Research on Trajectory Planning in Emergency Situations with Multiple Objects," **In**: *Proceedings of IEEE Intelligent Transportation Conference*, Toronto, Canada (Sep. 2006) pp. 988–992.
22. C. Yang, E. Blasch and I. Kadar, "Geometric Factors in Target Positioning and Tracking," **In**: *Proceedings of International Conference on Information Fusion*, Seattle, WA (Jul. 2009) pp. 85–92.
23. E. Edwan and R. Fierro, "A Low Cost Modular Autonomous Robot Vehicle," **In**: *Proceedings of the 38th Southeastern Symposium on System Theory*, Cookeville, TN (Mar. 2006) pp. 245–249.
24. D. Stronger and P. Stone, "Maximum Likelihood Estimation of Sensor and Action Model Functions on a Mobile Robot," **In**: *Proceedings of IEEE International Conference on Robotics and Automation*, Pasadena, CA (May 2008), pp. 2104–2109.
25. G. S. Fishman, *Monte Carlo: Concepts, Algorithms, and Applications* (Springer-Verlag, New York, 1996).
26. F. Belkhouche and B. Belkhouche, "Kinematics-based characterization of the collision course," *Int. J. Robot. Autom.* **23**(2), 206–3123 (2008).
27. S. Chen and Y. Li, "Automatic sensor placement for model-based robot vision," *IEEE Trans. Syst. Man Cybern.* **34**(1), 393–408 (2004).
28. F. Belkhouche, "Reactive path planning in a dynamic environment," *IEEE Trans. Robot.* **25**(4), 902–911 (2009).
29. A. Chakravarthy and D. Ghose, "Obstacle avoidance in a dynamic environment: A collision cone approach," *IEEE Trans. Syst. Man. Cybern. A: Systems and Humans* **28**(5), 562–574 (1998).
30. G. Welch and G. Bishop, "An introduction to the Kalman filter," *Technical Report TR 95–041* (University of North Carolina at Chapel Hill, NC, 2006).
31. A. Farina, B. Ristic and D. Benvenuti, "Tracking a ballistic target: Comparison of several nonliear filters," *IEEE Trans. Aerosp. Electron. Syst.* **38**(3), 854–867 (2002).
32. A. Cedilik, K. Kosmelj and A. Blejec, "The distribution of the ratio of jointly normal variables," *Metodoloski Zvezki: Adv. Methodol. Stat.* **1**(1), 99–108 (2004).
33. D. Kamerud, "The random variable x/y, x, y normal," *Am. Math. Mon.* **85**, 207 (1978).

https://doi.org/10.1017/S0263574712000550 Published online by Cambridge University Press