Automatica 44 (2008) 2347–2351

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Brief paper

A framework and automotive application of collision avoidance decision making[☆]

Jonas Jansson[a], Fredrik Gustafsson[b,*]

[a] The Swedish National Road and Transport Research Institute, Linköping, Sweden
[b] Department of Electrical Engineering, Linköping University, Linköping, Sweden

ARTICLE INFO

Article history:
Received 6 December 2005
Received in revised form
16 November 2007
Accepted 11 January 2008
Available online 21 March 2008

Keywords:
Automotive control
Decision support
Decision theory
Collision avoidance
Non-linear filtering
Kalman filter

ABSTRACT

Collision avoidance (CA) systems are applicable for most transportation systems ranging from autonomous robots and vehicles to aircraft, cars and ships. A probabilistic framework is presented for designing and analyzing existing CA algorithms proposed in literature, enabling on-line computation of the risk for faulty intervention and consequence of different actions. The approach is based on Monte Carlo techniques, where sampling-resampling methods are used to convert sensor readings with stochastic errors to a Bayesian risk. The concepts are evaluated using a real-time implementation of an automotive collision mitigation system, and results from one demonstrator vehicle are presented.

1. Introduction

Collision Avoidance (CA) systems are being used in a wide range of different areas and under very different circumstances. Automotive manufacturers have started to introduce systems that give a warning and/or perform autonomous braking when a collision is imminent (Jansson et al., 2002; Tamura et al., 2001). Radar-based *air traffic control* (ATC) systems have been in use for several decades. These systems typically aim at helping pilots and air traffic controllers in keeping a regulated minimum separation between any two aircraft. In Kuchar and Yang (2000) an extensive review of methods for collision avoidance in ATC systems is given. In marine applications, radar systems are commonly used to detect other vessels in the vicinity (Sato & Ishii, 1998). Industrial robotic control and autonomous robots is also a discipline where CA plays an important role (Kyriakopoulos & Saridis, 1993). Here, one is often interested in finding a control law which guarantees reaching a goal state without colliding. Thus, the CA system is an integral part of a control system that has other mission objectives. Common approaches to this problem are potential field algorithms, which were introduced in Khatib (1986) and path planning methods (Latombe, 1991). The task of any CA system is ultimately to keep the system host from colliding with other objects by warning the operator or performing an autonomous avoidance maneuver. We will extend the notion of a CA system to also include systems trying to reduce the consequence of an imminent collision i.e. *Collision Mitigation* (CM) systems. An even broader interpretation of a CA system discussed in the sequel is as a *conflict* avoidance system, where the conflict might be defined by a safety zone or corridor. Any action performed by a CA system will be called an *intervention*.

In many cases, inaccurate sensor information leads to uncertain state information that influences the performance of a CA system, see Fig. 1. We will here take a probabilistic viewpoint, by accepting stochastic models for sensor errors and propagate these all the way to the decision making. The purpose of this paper is to introduce a general framework for decision making in collision avoidance applications. In particular, the use of Monte Carlo methods allows for systems with non-linear dynamics and non-Gaussian noise distributions. Thus, the framework serves to reduce the decision making problem for complicated systems into a simple decision rule. Our approach is motivated by the general progress of Monte Carlo techniques, since, as pointed out in Smith and Gelfand (1992), this approach basically solves all Bayesian inference problems asymptotically in computational load, with easily implemented numerical algorithms. This provides a framework for the question mark in Fig. 1, and the ultimate goal is to compute the Bayesian risk for any CA system on-line. For this purpose, a demonstrator

☆ The application part was presented at IFAC World Congress 2002, Barcelona. This paper was recommended for publication in revised form by Associate Editor Keum-Shik Hong under the direction of Editor Mituhiko Araki.

* Corresponding author.
E-mail addresses: jonas.jansson@vti.se (J. Jansson), fredrik@isy.liu.se (F. Gustafsson).

0005-1098/$ – see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.automatica.2008.01.016

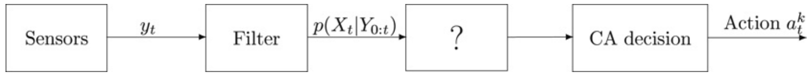

Fig. 1. Sensor measurements y_t are inherently uncertain, and the state estimate used for situational awareness becomes uncertain. On the other hand, decision making is usually based on rules, assuming a known state. The problem is how to interface these two parts.

vehicle equipped with radar and lidar has been used to evaluate a CM system based on Monte Carlo integration for decision making.

2. Theoretical framework

2.1. Conflict notation

Consider a state vector X_t comprising current tracking information. It may contain quantities as relative longitudinal position $\tilde{p}_{x,t}$, relative lateral position $\tilde{p}_{y,t}$ and relative heading angle $\tilde{\psi}_t$. The tilde ˜ denotes relative quantities as the relative position between the host and a threat (stationary or moving obstacle), or the deviation from a nominal trajectory. The latter case is common for aircraft and marine systems, where the host is supposed to follow a pre-defined path.

A near future conflict arises for certain trajectories $X_{t:t+T} = \{X_s; t \le s \le t+T\}$. Here, T is the time horizon of the CA system, which is a compromise between computation time and minimizing the risk to miss a conflict.

A binary conflict function is defined as

$$C(X_{t:t+T}) = \begin{cases} 1, & \text{conflict,} \\ 0, & \text{no conflict.} \end{cases} \tag{1}$$

An example of a conflict function is given by

$$C(X_{t:t+T}) = I\left(\min_{s\in[t,t+T]} \|(\tilde{p}_{x,s}, \tilde{p}_{y,s})\| < R_{\text{conf}}\right), \tag{2}$$

where $I(A) \in \{0, 1\}$ denotes the indicator function which is one if A is true and zero otherwise, R_{conf} denotes a distance defining the conflict region and the norm denotes the spatial distance between the host and obstacle k. Here, a conflict is defined as any position where the centers of the host and threat are closer than R_{conf}. A more general conflict function takes relative orientation into account.

If the CA system could look into the future, it would intervene whenever $C(X_{t:t+T}) = 1$. However, from causality it has to make the decision on past measurements $Y_{0:t}$. For this reason, the CA system can be considered to base its decisions on a decision function $g(Y_{0:t})$. Without limiting the scope of application, the construction of this function consists of the following two steps:

(1) Navigation and tracking algorithms give the *a posteriori* probability density function (PDF) $p(X_t|Y_{0:t})$ for the state vector of the host and threats.
(2) Prediction gives the PDF for future trajectories $p(X_{t:t+T}|Y_{0:t})$.

For this discussion, we do not need to specify the navigation, tracking and prediction models, or whether these are given in continuous or discrete time.

2.2. Conflict detection

In a Bayesian perspective, the state trajectory given the measurements is a stochastic process, and (1) becomes a hypothesis test:

$$H_0 : C(X_{t:t+T}) = 0, \tag{3a}$$

$$H_1 : C(X_{t:t+T}) = 1. \tag{3b}$$

It is well-known from the Neyman–Pearson theorem (Kay, 1998) that the likelihood ratio

$$g(Y_{0:t}) = \frac{p(Y_{0:t}|H_1)}{p(Y_{0:t}|H_0)} = \frac{p(Y_{0:t}|C(X_{t:t+T}) = 1)}{p(Y_{0:t}|C(X_{t:t+T}) = 0)} \gtrless \lambda \tag{4}$$

is the optimal decision function for intervention, in that it maximizes the probability of conflict detection, P_D

$$P_D = \text{Prob}(g(Y_{0:t}) > \lambda|H_1), \tag{5}$$

under the constraint of constant false alarm probability P_{FA},

$$P_{FA} = \text{Prob}(g(Y_{0:t}) > \lambda|H_0). \tag{6}$$

This constraint specifies the detection threshold λ. Using Bayes' theorem, (4) can be re-written

$$\begin{aligned} g(Y_{0:t}) &= \frac{\text{Prob}(H_1|Y_{0:t})}{\text{Prob}(H_0|Y_{0:t})} \frac{\text{Prob}(H_1)}{\text{Prob}(H_0)} \\ &= \frac{\text{Prob}(C(X_{t:t+T}) = 1|Y_{0:t})}{\text{Prob}(C(X_{t:t+T}) = 0|Y_{0:t})} \frac{\text{Prob}(H_1)}{\text{Prob}(H_0)} \gtrless \lambda. \end{aligned} \tag{7}$$

Here, $\text{Prob}(H_1)$ is interpreted as the prior of conflict, and $\text{Prob}(H_0) = 1 - \text{Prob}(H_1)$. It can be shown (Kay, 1998) that choosing $\lambda = 1$ in (7), gives the test that minimizes the probability of incorrect decision

$$\begin{aligned} P_e &= (1 - P_D) + P_{FA} = \text{Prob}(g(Y_{0:t}) < 1|H_0)p(H_0) \\ &\quad + \text{Prob}(g(Y_{0:t}) < 1|H_1)p(H_1). \end{aligned} \tag{8}$$

2.3. Bayesian risk and cost

The consequence of false alarm and missed detection is different for warning, avoidance and mitigation systems. The threshold λ in (4) or the prior $\text{Prob}(H_1)$ in (4) can be used to tune this trade-off. However, a more convenient tool to design CA systems in the same framework is the Bayes' risk, or expected cost of a decision. Define a cost c_{ij} to each decision. That is, c_{10} is the false alarm cost and c_{01} is the missed detection cost. Here, the false intervention cost c_{10} is higher for avoidance than for warning systems. The Bayes risk or expected cost is given by

$$\mathcal{R} = \sum_{i=0}^{1} \sum_{j=0}^{1} c_{ij} P(H^i|H^j) p(H^j).$$

This risk is minimized by modifying the test (7) to

$$g(Y_{0:t}) = \frac{\text{Prob}(C(X_{t:t+T}) = 1|Y_{0:t})}{\text{Prob}(C(X_{t:t+T}) = 0|Y_{0:t})} \tag{9a}$$

$$\gtrless \frac{(c_{10} - c_{00})}{(c_{01} - c_{11})} \frac{\text{Prob}(H_0)}{\text{Prob}(H_1)}. \tag{9b}$$

The cost for correct decisions can be taken as $c_{00} = c_{11} = 0$, and the risk ratio c_{10}/c_{01} has the same influence as the prior probability ratio. These fundamental expressions for probability and risk for incorrect decisions are for instance found in Kay (1998).

2.4. Implementation

The question is how to compute $\text{Prob}(C(X_{t:t+T}) = i|Y_{0:t}), i = 0, 1$. For linear Gaussian models, $p(X_{t+\tau}|Y_{0:t}), 0 \le \tau \le T$ is Gaussian, and an analytical expression may exist in theory. Generally, $C(X_{t:t+T})$ is a non-linear function of state, and cannot be evaluated analytically. The proposed approach is based on Monte Carlo integration, using