Markos Papageorgiou
Thomas Bauschert
Lehrstuhl für Steuerungs- und Regelungstechnik
Technische Universität München
Postfach 20 24 20
D-8000 Munich, Germany

Stochastic Optimal Control of Moving Vehicles in a Dynamic Environment

Abstract

This article develops a fairly general framework for derivation of control strategies applying to moving objects, such as mobile robots or robot arms, in a dynamically changing environment. The basic idea is to consider stochastic terms for any uncertain future environment change and to apply stochastic dynamic programming for control strategy development. The method permits consideration of a number of possible missions such as collision avoidance or collision hitting and/or moving forward or following a trajectory. A number of examples demonstrate that the control strategies developed are capable of making efficient, human intelligence–like decisions in quite complicated conflict situations.

1. Introduction

In the past decade much research work has been devoted to the navigation and path control of moving vehicles such as intelligent mobile robots or robot arms. There are a number of challenging problems to be resolved in this context, including (1) information acquisition by use of (possibly multiple) sensors, (2) information interpretation and processing, (3) design of appropriate moving actions so as to carry out a prescribed mission, and (4) execution of the designed moving actions.

The present article addresses task (3) for which a further subdivision into different problem classes is possible. This subdivision may depend on the nature of the vehicle's environment (e.g., stationary or dynamic) and on the time information becomes available (e.g., a priori information or real-time information). Thus we may distinguish the following problem classes:

3a. The problem of planning a navigation plan in a known *stationary* environment. In this case the position and geometric shape of external objects are known before the plan generation. Moreover, it is assumed that they are stationary (i.e., they do not change with time) (Brooks and Lozano-Pérez 1983; Brooks 1983; Iyengar et al. 1986; Graglia and Meystel 1987).

3b. The problem of planning a navigation plan or a path in a known *dynamic* environment. In this case the position, the geometric shape, and the future movement (e.g., the speed) of external objects are known before plan generation (Erdmann and Lozano-Pérez 1986; Fujimura and Samet 1989).

3c. The problem of controlling the movement of the vehicle in a dynamic environment where only partial or no information is available beforehand, but real-time information becomes available as the vehicle moves. In other words, this kind of problem calls for a real-time control strategy capable of reacting to unexpected events on the vehicle's way (Khatib 1986; Griswold and Eem 1990).

Certainly, a combination of these problems may have to be resolved for one and the same vehicle. For example, one may plan a (stationary or dynamic) navigation plan based on a priori available information and then design a real-time control strategy that will suitably react if, on the vehicle's way, the vehicle's sensors indicate deviations between real-world and a priori information.

The present article addresses a problem from the class 3c above.[1] More precisely, we consider control of a moving vehicle in a p-dimensional Euclidean space that may include further (extraneous) moving or stationary objects. The control task consists of:

- Utilizing real-time information on the current environment state (such as number of extraneous objects and their respective dimensions and/or positions, orientations, and speeds) and on the current vehicle's state (position, orientation, and/or speed)
- Deciding on the way of moving the subject with the vehicle's moving capabilities

1. It is interesting to note that the mathematical problem formulation of this article is general enough to cover also problems 3a and 3b. However, there are certainly more economical methods for addressing these problems than the dynamic programming approach used here.

The International Journal of Robotics Research,
Vol. 13, No. 4, August 1994, pp. 343-354,

- Carrying out a prescribed mission (such as avoiding collision with or hitting extraneous objects, moving ahead, and/or following a prescribed trajectory).

Certainly, acquisition, interpretation, and processing of information in real-time is a difficult task. However, the present article concentrates exclusively on the *use* of real-time information for movement control. The kind of real-time information required by the control strategy will be made more precise in the following sections. It should also be stressed that although some real-time information is available, the future behavior of the extraneous objects is unknown to the control strategy.

For the sake of brevity, we will call the controlled moving vehicle "the vehicle," and the extraneous objects, "objects." Certainly, autonomously moving vehicles or robot arms fall into the defined "vehicle" class.

We assume that the control strategy utilizes the available real-time information in order to make its decision at discrete times $k = 0, 1, 2, \ldots$. To make our problem more precise, we organize the decision or input variables at time k (e.g., the vehicle's speed or acceleration to be applied at time k) in a vector $\mathbf{u}(k) \in \mathrm{R}^m$. Moreover, we construct a vector $\mathbf{x}(k) \in \mathrm{R}^n$ that includes the real-time measurements (e.g., position, dimensions, speed of the objects). Then, our problem is to design a stationary decision policy or control strategy or feedback law $\mathbf{r} : \mathrm{R}^n \Rightarrow \mathrm{R}^m$ such that

$$\mathbf{u}(k) = \mathbf{r}[\mathbf{x}(k)] \tag{1}$$

At this point, it is important to make the distinction between computational effort required

- Off-line: this effort is required for designing $\mathbf{r}$ and is therefore attributed to a mainframe computer; this effort is off-line because it is required only once, before any movement of the vehicle starts; once the strategy $\mathbf{r}$ has been derived in a mainframe computer, it is implemented in the vehicle's computer and does not change anymore unless new problem requirements (e.g., a new kind of vehicle's mission) call for development of a new strategy.
- On-line: this effort is required for execution of (1) at each time k during the motion of the vehicle; therefore, this effort is attributed to the vehicle's on-board computer.

Perhaps the most popular approach for real-time control strategy design is the artificial potential approach proposed in Khatib (1986). This remarkably simple and elegant approach, however, is mainly suitable for local obstacle avoidance with low-speed obstacles. Moreover, the flexibility with respect to the kind of possible vehicle missions appears rather limited. Last but not least, the artificial potential approach applies only to convex form objects and may lead to standstill of the vehicle owing to local minima. Further real-time strategies have been proposed to deal with unknown but stationary (immobile) objects (Borenstein and Koren 1990).

The basic approach of this article is to consider the unknown future behavior of the objects via appropriate stochastic terms. In other words, for the derivation of the control strategy, we assume that

- There is no a priori information available on how the objects will move exactly, but
- there is a range of objects' movement (e.g., a range of speeds, accelerations, etc.) with corresponding probability distributions that is known a priori. This assumption is fairly realistic, because for a given application environment (e.g., a factory, an airport, etc.) the external objects usually have certain typical movement characteristics.

The mathematical method used for derivation of the control strategy is stochastic dynamic programming (Bertsekas 1976). A similar approach regarding the stochastic future movement of unknown objects is considered in Griswold and Eem (1990). However, Griswold and Eem (1990) consider exclusively vehicles moving along a prescribed trajectory, with vehicle acceleration and deceleration being the only controllable variables. In the present article, a number of different vehicle models with more general control maneuvers, missions, and constraints are investigated.

2. A Basic Problem

2.1. Assumptions of the Basic Problem

We will first consider a basic problem of vehicle control in a p-dimensional space by relying on some particular assumptions that will be relaxed or altered in the subsequent sections. The assumptions of this section are:

(i) The vehicle and the objects move without changing their respective orientations, or they have a spherical (in two dimensions, circular) form, or they may be represented by a maximum or minimum spherical hull. Each of these assumptions implies that we may—without loss of generality—consider the vehicle to be a moving point, its dimensions being transferred to the objects. (Brooks and Lozano-Pérez 1983; Fujimura and Samet 1989). This assumptin will be removed in Section 6.

(ii) We consider at most one object at a time in the vehicle's environment. Certainly, this assumption is very restrictive, and it may only apply to environments with a very low density of object population. However, we will have more to say about this assumption in Section 5.3.

(iii) The difference $\chi \in \mathrm{R}^p$ of the vehicle (considered as a point according to (i)) to a reference point of the object is measurable at each discrete time k (i.e., in real time) if the object is inside a given neighborhood $\mathbf{X} \subset \mathrm{R}^p$.

(iv) The physical dimensions and the orientations of the objects that appear on the vehicle's way are the same for all objects, are constant in time, and are known a priori. For example, in application environments where the external objects are only humans or only other moving vehicles, their physical dimensions will be equal and known a priori. This assumption, which certainly limits applicability of the strategy, will be removed in Section 5.2.

(v) Between two successive discrete times k, $k+1$, the vehicle moves with an average speed $\mathbf{v}(k) \in \mathrm{R}^p$ (in world coordinates), which may be selected out of an admissible speed region $\mathbf{V} \subset \mathrm{R}^p$. The significance of this assumption is the following: if the control strategy orders at time k a speed $\mathbf{v}(k)$ for the vehicle, the vehicle should be able to move (see problem 4 in the Introduction) to a new position at time $k+1$ that is in accordance with the ordered speed. Moreover, the assumption says that the control strategy is allowed only to order speeds out of a region V that corresponds to the vehicle's moving capabilites. In Section 4.1 we will also consider the case of limited acceleration capabilities for the vehicle.

(vi) Between two successive discrete times k, $k+1$, the object moves with a stochastic world average speed $\varphi(k) \in \mathrm{R}^p$ (in world coordinates) that has an a priori known stationary probability distribution. The components of φ are mutually independent and have independent values at successive discrete time intervals.[2] Of course, these assumptions appear unrealistic in environments where objects usually move with more or less constant, rather than stochastic, speeds. In fact, we will be able to remove this assumption in Section 4.2. However, it should be stressed that generally the introduction of stochastic terms is crucial for the purposes of this article. In fact, the reason for introducing stochastic terms is to make the control strategy aware of the fact that the environment is subject to change in an unknown way but within certain known limits as they are expressed by the probability distributions. This is essential for a cautious behavior of the control strategy. On the other hand, even if these stochastic assumptions are not met exactly in real life, the control strategy may perform satisfactorily, as demonstrated by some of the following examples. In other words, the precise mathematical assumptions of stochasticity are just a means of obtaining control strategies that are robust enough to apply to fairly different real situations.

2. The same independence properties are assumed to be present for all stochastic processes that will be introduced later in the article.

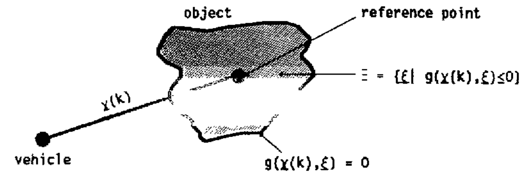

Fig. 1. Visualization of variables' definitions.

(vii) The task of the vehicle is to move in a given space direction (for example, v_1) with a speed that is as close as possible to a given maximum (or desired) speed $v_{1,\max}$, avoiding collisions with moving objects and unnecessary deviations. Let us comment on this assumption: First, the prescribed vehicle's task is present, for example, if the vehicle must move along a corridor or must cross a hall without following a precise path; the specification of the direction v_1 for vehicle movement is without loss of generality, because we may rearrange the components of $\mathbf{v}$ so that this part of the assumption is always fulfilled; the requirements of maximum (or desired) speed and of avoidance of unnecessary deviations aim at performing the task in short (or scheduled) time. Second, this assumption describes just one task out of a high number of possible tasks and we will address some more of them in Section 3; of course, each different task will lead to a different control strategy.

2.2. Object Representation

Assume that the reference point (see assumption (iii)) of an object is located at a difference $\chi(k)$ from the vehicle. Then, the p-dimensional piece of space in the vehicle's coordinates that is occupied by the object (augmented by the vehicle's dimensions according to (i)) may be expressed by the set (Fig. 1)

$$\Xi(k) = \{\boldsymbol{\xi} \in \mathrm{R}^p | \mathbf{g}(\chi(k), \boldsymbol{\xi}) \leq \mathbf{0}\}, \qquad (2)$$

where $\mathbf{g}$ is a vector of functions that describe the shape of the object. For example, in two dimensions ($p = 2$), if the object is a disc with radius R and its reference point is the disc center, we have

$$\Xi(k) = \{\boldsymbol{\xi} \in \mathrm{R}^2 | [\xi_1 - \chi_1(k)]^2 + [\xi_2 - \chi_2(k)]^2 - R^2 \leq 0\};$$

if the object is the upper half of the disc, we have

$$\Xi(k) = \{\boldsymbol{\xi} \in \mathrm{R}^2 | [\xi_1 - \chi_1(k)]^2 + [\xi_2 - \chi_2(k)]^2 \leq R^2; -\xi_2 + \chi_2(k) \leq 0\},$$

etc. Some implications of this fairly general definition of an object are discussed in Section 5.1.

For the time being, let us note that assumption (iv) asks for the functions **g** of all objects to be the same and known a priori.

2.3. Basic Problem Formulation: Problem P1

Given the above assumptions, we are looking for a control strategy (1) where $\mathbf{u}(k) = \mathbf{v}(k)$ is the input vector to be decided on the basis of real-time measurements $\mathbf{x}(k) = \boldsymbol{\chi}(k)$.

We will now develop a problem formulation that leads to the derivation of the control strategy in a straightforward way. We start with a suitable dynamic model of the moving process

$$\boldsymbol{\chi}(k+1) = \boldsymbol{\chi}(k) - \mathbf{v}(k) + \boldsymbol{\varphi}(k), \tag{3}$$

where the discrete time interval was normed to unity.

The vehicle's task, as expressed under (vii) previously, may be fulfilled by minimization of the following performance criterion:

$$J = \mathrm{E}\left\{ \sum_{k=0}^{\infty} \left[\alpha_1 (v_{1,\max} - v_1(k))^2 + \sum_{i-2}^{p} \alpha_i v_i(k)^2 + \psi[\mathbf{g}(\boldsymbol{\chi}(k), \mathbf{0})] \right] \right\} \tag{4}$$

With reference to the performance criterion (4), some clarifications are required:

- The first term in the sum penalizes any retarding of the vehicle along its moving direction v_1.
- The second term penalizes any deviations along the other directions $v_2, \ldots, v_p$.
- For the third term, let us first define the function ψ as follows:

$$\psi = \begin{cases} 1 & \text{if } \mathbf{g}[\boldsymbol{\chi}(k), \mathbf{0}] \leq \mathbf{0} \text{ holds} \\ 0 & \text{if for some } i : g_i[\boldsymbol{\chi}(k), \mathbf{0}] > 0 \text{ holds} \end{cases}$$

  With this definition, the third term penalizes collision with the object. To see this, we note that according to (2) a collision occurs iff $\mathbf{g}[\boldsymbol{\chi}(k), \mathbf{0}] \leq \mathbf{0}$ (i.e., if the origin falls into the space occupied by the object). In this case, ψ becomes 1 and is penalized in (4); otherwise ψ remains zero.
- Taking squares in the first two terms leads to a stronger penalization of larger deviations. However, absolute values or any other positive definite penalizing function may be used instead.
- The choice of the weighting parameters $\alpha_1, \ldots, \alpha_p$ determines the relative importance of delays and deviations as compared with the collision risk. Too-high values of these parameters may lead to a risky policy, whereas too-low values may lead to a hesitating moving. The weighting parameters must be specified off-line by a trial-and-error procedure.
- The optimization horizon (i.e., the upper bound of the k-sum) is taken to be infinite in order to obtain a stationary control strategy. This choice is in accordance with the problem's physical background, since the overall time required for a conflict resolution is unknown a priori.
- In view of the object's stochastic speed $\boldsymbol{\varphi}(k)$ (see eq. (3)), we are forced to consider expectation of the time-sum over the known probability distribution of $\boldsymbol{\varphi}(k)$.

We are now in a position to formulate a stochastic optimization problem P1, whose solution leads to derivation of a control strategy (1) for the vehicle's movement:

> Find a control strategy (1) that minimizes the performance criterion (4) subject to the process equation (3), subject to the known probability distribution of $\boldsymbol{\varphi}(k)$ and to $\mathbf{v}(k) \in V$, $\boldsymbol{\chi}(k) \in X$.

We recall that this problem must be resolved off-line on a mainframe computer.

A number of additional assumptions applying to corresponding practical applications may be viewed as special cases of problem P1. For example, if the vehicle has a constant, noncontrollable forward speed $v_1(k) = \bar{v}_1$, then the control vector reduces to $\mathbf{u} = [v_2 \ldots v_p]^T$, and the first term in the performance criterion (4) is not needed. Another interesting situation occurs if $\varphi_1(k) = 0$ (i.e., the object moves in a $(p-1)$–dimensional plane that is orthogonal to the v_1 direction). In this case the vehicle attempts to cross the plane without collision.

2.4. The Solution Method: Stochastic Dynamic Programming

To be able to solve problem P1, we must first proceed to a discretization of the problem variables $\boldsymbol{\chi}$, $\mathbf{v}$, $\boldsymbol{\varphi}$. Let us consider, for example, discretization of $\boldsymbol{\chi} \in X$ in two dimensions ($p = 2$). For simplicity, we assume that X is a rectangular neighborhood around the vehicle with dimensions $X_1 \times X_2$. Then we may subdivide X in $\nu_{\chi^1} \cdot \nu_{\chi^2}$ rectangular cells, where $\nu_{\chi^1}, \nu_{\chi^2}$ are the numbers of discrete values for χ_1 and χ_2, respectively. The significance of discretization for the control strategy is the following: if the coordinates of an object's reference point lie in a given cell, the control strategy will react in the same way, regardless of their exact position inside the cell. Hence, a more dense discretization leads to a more refined reaction of the resulting control strategy. In the same way, we may define ν_{v^i}, ν_{φ^i} as the numbers of discrete values for the ith component of the vectors **v**, φ, respectively.

After discretization, the problem formulation P1 fits exactly the requirements of the stochastic dynamic programming (SDP) method, and hence, solution of P1 is straightforward. Because the SDP algorithm is simple and well known (Bertsekas 1976), there is no need to present it here. It suffices to say that the result of the corresponding off-line calculations is a control strategy (1). This control strategy (to be applied in real time) tells in a tabular form what the value of $\mathbf{v}(k)$ should be for a given $\boldsymbol{\chi}(k)$. More precisely, for each discrete $\boldsymbol{\chi}$-cell, the control strategy tells what the corresponding vehicle speed values (by speed coordinate) should be that lead to a global minimum of the performance criterion.

Let us now comment on the off-line and on-line computational effort:

1. Off-line effort for *derivation* of the control strategy in a mainframe computer: The required computation time is known (Bertsekas 1976) to be proportional to
$$\sqcap_{i=1}^{p}\nu_{\chi i}\cdot\sqcap_{i=1}^{p}\nu_{vi}\cdot\sqcap_{i=1}^{p}\nu_{\varphi i}.$$
This exponential complexity formula indicates that the off-line computation time required for derivation of the control strategy may become overwhelming for high-order problems. However, some savings in both off-line and on-line calculations can be achieved by taking advantage of the symmetry of the control strategy, which will become apparent in the examples. Further savings may be achieved if the $\boldsymbol{\chi}$-discretization is chosen sufficiently fine in the very neighborhood of the vehicle, but coarser farther away. Finally, some recent developments in computer technology may help to retard the computational burdens imposed by the notorious "curse of dimensionality" of the dynamic programming approach (Chung and Hanson 1990).
2. On-line effort for *execution* of the control strategy in an on-board computer: Given the real-time measurements $\chi(k)$ (corresponding to the current position of the object), the computer must specify the corresponding χ-cell and must pick up the corresponding **v**-values that have been assigned to this cell. This task may be easily performed in real time.

Let us also note that owing to the feedback nature of the control strategy, sensitivity with respect to model simplifications, discretization, motion inaccuracies, measurement errors, and further disturbances are expected to be low. For example, if the vehicle speed $\mathbf{v}(k)$ that has been ordered by the control strategy at time k is not performed accurately, then in the next time step $k+1$, the new measurement $\boldsymbol{\chi}(k+1)$ will take account of the *real* vehicle's position so that the control strategy will respond adequately and so forth. In other words, the feedback law avoids an accumulation of errors on the vehicle's way.

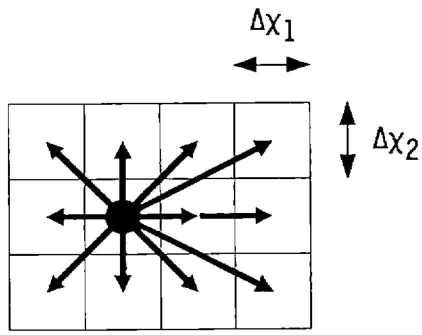

Fig. 2. Possible vehicle movements.

2.5. An Example

Assume the vehicle is moving in a two-dimensional space ($p = 2$) that includes rectangular objects with dimensions $l_1 = 1.6$, $l_2 = 4.0$. The reference point of the objects is taken to be the center of the rectangle. The vehicle's admissible speed region V is defined by $v_1 \in \{-0.4, 0, 0.4, 0.8\}$ and $v_2 \in \{-0.4, 0, 0.4\}$. In other words, the maximum forward speed is 0.8, but the vehicle may reduce its forward speed to 0.4 or it may stop or backtrack. Moreover, at each discrete time, the vehicle is allowed to deviate to the left or to the right by 0.4. Figure 2 visualizes the possible vehicle movements at each discrete time k. For derivation of the control strategy, the object is assumed to move with speed $\varphi_i, i = 1, 2$, taken out of the set $\{-0.8, -0.4, 0, 0.4, 0.8\}$ with equal probability distribution. In view of these settings, the performance criterion becomes

$$J = \mathrm{E}\left\{\sum_{k=0}^{\infty}[\alpha_1(0.8 - v_1(k))^2 + \alpha_2 v_2(k)^2 + \Psi(\chi(k))]\right\} \tag{5}$$

where

$$\Psi(\chi) = \begin{cases} 1 & \text{if } |\chi_1| \le l_1/2 \text{ and } |\chi_2| \le l_2/2 \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

The weighting parameters were chosen as $\alpha_1 = 0.05$, $\alpha_2 = 0.1$. In accordance with the discretization of **v** and φ, we have chosen the space increments $\Delta\chi_1 = \Delta\chi_2 = 0.4$. Solving problem P1 (Bauschert 1990) with these specifications required 25 minutes 27 seconds CPU time on a VAX-Station 2000 without taking advantage of the existing symmetry with respect to the χ_1-axis, which

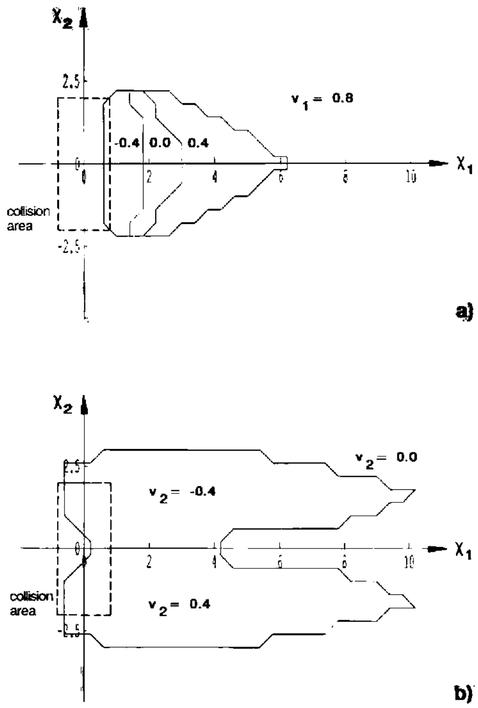

Fig. 3. Control strategy for the basic problem A, $v_1(\chi_1, \chi_2)$. B, $v_2(\chi_1, \chi_2)$.

is apparent in view of the control strategy depicted in Figure 3.

Figure 3 is understood as follows: The origin corresponds to the vehicle's location. The dashed-line rectangle around the origin indicates the collision area. The two-dimensional space χ is subdivided into regions corresponding to the discrete values of the vehicle's speed. More precisely, at each discrete time k the reference point of the object falls into a certain region that specifies the indicated v_1 (Figure 3*A*) and v_2 (Figure 3*B*) values for the vehicle moving during the kth time interval.

Figure 3*A* depicts a fairly plausible vehicle's moving strategy. The vehicle gradually reduces its speed v_1 if the object is located in corresponding distances in front of it. As for the speed v_2 (Figure 3*B*), the vehicle deviates if a collision appears probable owing to the object's location. An interesting, and perhaps unexpected, result is that, if the object is located very near the χ_1-axis, the vehicle deviates rather late. This relaxed behavior relies on the assumption that the object is mobile, and eventually there is a high probability that it may move away from the vehicle's way.

It is interesting to note that the regions of Figure 3 specify the useful real-time information required for vehicle control. If information (e.g., visibility region of a mobile robot) is available for a larger region around the vehicle, there will be no use for it. If less information is available, then the resulting strategy will be a suboptimal one unless the whole problem is considered as an incomplete information problem (Bertsekas 1976), and appropriate estimators are introduced.

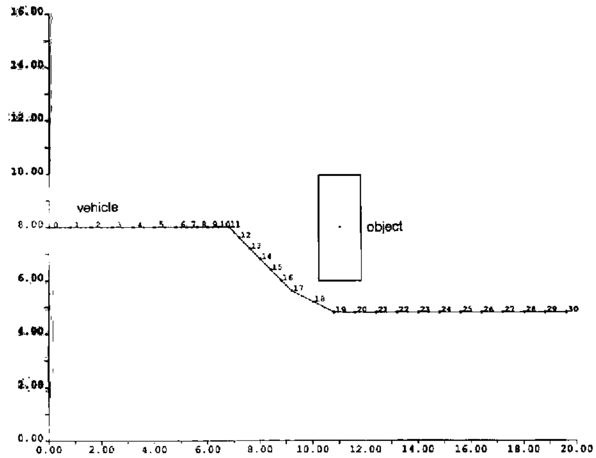

Fig. 4. Scenario with an immobile object.

To demonstrate the resulting vehicle's behavior, we will present two scenarios out of a high number produced in Bauschert (1990). Both scenarios have been produced on the basis of the same control strategy shown in Figure 3. Visualization of all scenarios of the article will be based on two-dimensional world coordinates.

Figure 4 depicts the vehicle trajectory for a scenario with an immobile object. The digits along the trajectory indicate the corresponding discrete times. The vehicle reaction to the immobile object located on its way is correct:

- The vehicle deviates to bypass the obstacle and at the same time
- reduces slightly its forward speed while navigating around the obstacle.

The second scenario, depicted in Figure 5, includes an object that moves with a stochastic speed $\varphi_i \in [-1.0, 1.0], i = 1, 2$, with equal probability distribution. Note that this is a different behavior than what was assumed for derivation of the control strategy. Note also that the object's maximum speeds in both directions are higher than the vehicle's maximum speeds. In Figure 5, the object's dimensions are indicated merely at time $k = 0$ by a dashed rectangle.

The second scenario gives rise to a fairly complicated conflict situation: Initially the vehicle is moving straight ahead. At $k = 4$ the object lies on the right of the vehicle, which deviates to the left. However, during the time period 4–9, the object moves quickly upward, and at $k = 9$ it blocks the way of the vehicle, which feels forced to backtrack. During the time period 9–19, the vehicle deviates upward and looks for an opportunity to pass. A first

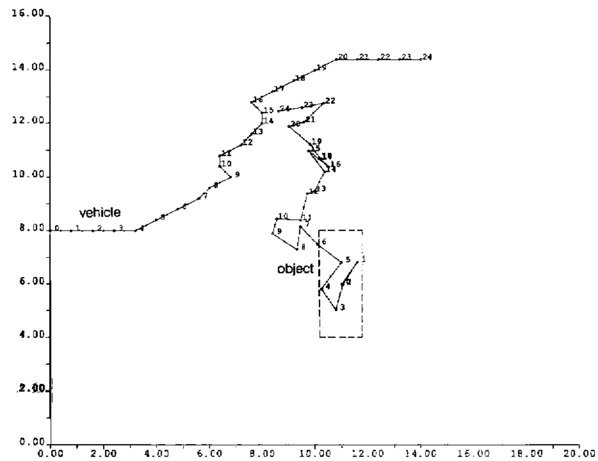

Fig. 5. Complicated scenario with a mobile robot object.

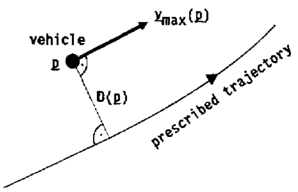

Fig. 6. Following a trajectory.

attempt leads to a new backtracking ($k = 15$), but a second attempt is finally successful. A similarly "intelligent" vehicle behavior was visible in a number of additional investigated scenarios.

3. Different Vehicle's Tasks

Task (vii) described in Section 2.1 is not binding for the method presented in this article. Of the many different tasks that may be considered (e.g., move to a destination point, cross a road, etc.), we will briefly discuss two: following a trajectory and a Watchdog task.

3.1. Following a Trajectory: Problem P2

Assume the vehicle's task is to move with a speed that is as close as possible to maximum speed, but on a given trajectory instead of just moving forward, as was assumed in the basic problem. Again, collisions with objects and unnecessary deviations from the trajectory should be avoided.

The dynamic equation for the position $\mathbf{p}$ of the vehicle in world coordinates is (Fig. 6):

$$\mathbf{p}(k+1) = \mathbf{p}(k) + \mathbf{v}(k). \tag{7}$$

Because the prescribed trajectory is known, it is possible to derive (off-line) a function $D(\mathbf{p})$ that gives the shortest distance from a point $\mathbf{p}$ to the prescribed trajectory (Fig. 6). Let us define the vehicle's *deviation* from the trajectory by $d(k) = D[\mathbf{p}(k)]$. Then we have

$$\begin{aligned} d(k+1) &= D[\mathbf{p}(k+1)] = D[\mathbf{p}(k) + \mathbf{v}(k)] \\ &\approx D[\mathbf{p}(k)] + \mathbf{v}(k)^T \mathbf{D}'[\mathbf{p}(k)] \end{aligned}$$

where the prime denotes derivation. From this relation we directly obtain the following linearized model of the dynamics of the trajectory deviation d:

$$d(k+1) = d(k) + \mathbf{v}(k)^T \mathbf{D}'(\mathbf{p}(k)). \tag{8}$$

A modified performance criterion, taking into account the new task, should include as a first and second term

$$\alpha_1 \|\mathbf{v}_{\max}(\mathbf{p}(k)) - \mathbf{v}(k)\|^2 + \alpha_2 d(k)^2,$$

instead of the corresponding first two terms of (4), the third term therein remaining equal. The known function $\mathbf{v}_{\max}(\mathbf{p})$ stands for the maximum speed in the direction of move indicated by the prescribed trajectory (see Figure 6).

This completes the definition of the "Following a Trajectory" problem, or Problem P2. Owing to (7) and (8) we now have an augmented state vector $\mathbf{x} = [\boldsymbol{\chi}^T \mathbf{p}^T d]^T$ of the order $2p+1$, and hence (in addition to $\boldsymbol{\chi}$), the vehicle's world position $\mathbf{p}$ and its deviation d from the prescribed trajectory must now also be measurable in real time. Naturally, a heuristic solution of this problem may be to apply the control strategy P1 if an object is visible by the vehicle's sensors and to drive toward or on the prescribed trajectory if it is not.

3.2. Watchdog Task

Assume the vehicle's task is to hit an object, if any object is included in X, and stand still if there is no visible object in X. This task may be achieved by minimizing

$$J = \mathrm{E}\left\{ \sum_{k=0}^{\infty} [\alpha \|\mathbf{v}(k)\|^2 + 1 - \psi[\mathbf{g}(\chi(k), \mathbf{0})]] \right\}, \tag{9}$$

where the first term has been introduced to avoid useless moving. The term $1 - \psi$ in (9) obviously penalizes the vehicle's movement if the object has not yet been intercepted. We have also produced scenarios of P1 vehicles against Watchdog vehicles and of each of them against human players (Bourdé 1990). Clearly, in these cases we encounter stochastic game problems for which a fairly extensive literature exists (Mycielski 1988).

3.3. An Example

We consider the same situation discussed in Section 2.5, but now the task of the vehicle is to move along a prescribed trajectory that is parallel to the v_1-axis. Under

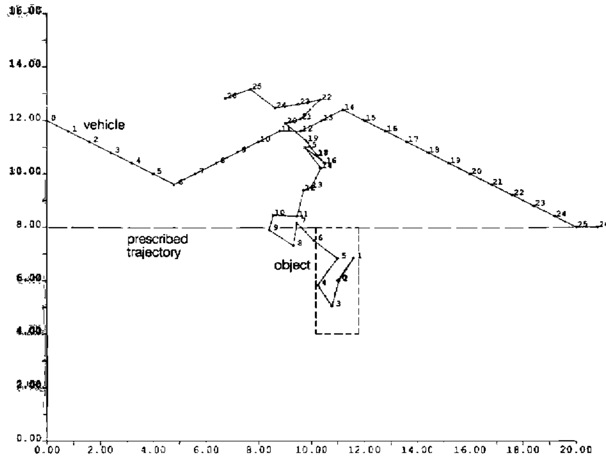

Fig. 7. Prescribed trajectory task: scenario 1.

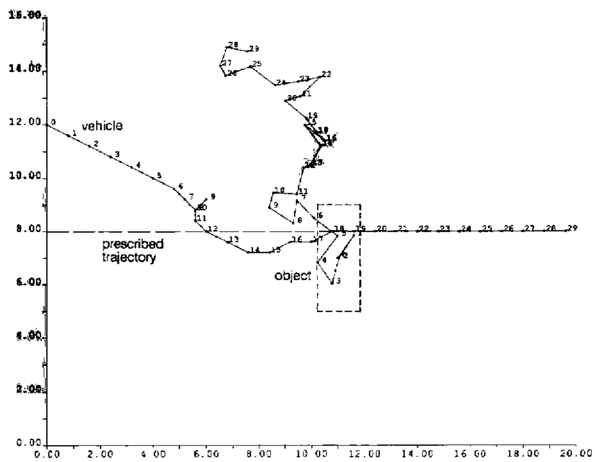

Fig. 8. Prescribed trajectory task: scenario 2.

these conditions the deviation equation (8) becomes

$$d(k+1) = d(k) + v_2(k), \tag{10}$$

and the position equation (7) is not needed. Moreover, $\mathbf{v}_{\max} = [v_{1,\max} 0]^T$ holds, and we have chosen the weighting factors $\alpha_1 = 0.05$, $\alpha_2 = 0.001$. With these arrangements we have produced a new control strategy P2, for which we now give some interesting scenarios.

Figure 7 depicts the same conflict situation shown in Figure 5 but applying the new control strategy P2. Initially the vehicle moves straight toward its trajectory, but in view of the object it feels forced to deviate to the left. After it has bypassed the object, the vehicle drives straight toward its trajectory.

A more complicated situation is created if we slightly move the initial position of the object upward (Fig. 8). At time $k = 8$, the vehicle attempts to deviate to the left, but the object's movement makes the vehicle change its mind and backtrack before deviating to the right.

4. Acceleration Considerations

4.1. Vehicle Acceleration

Assumption (v) provides to the control strategy complete freedom of selecting the vehicle's speed from a set V at each discrete time k. This may be unrealistic for some applications because of the vehicle's inertia. A more realistic approach may be to consider $\mathbf{v}$ as a measurable state variable described by the dynamic equation

$$\mathbf{v}(k+1) = \mathbf{v}(k) + \mathbf{a}(k), \tag{11}$$

with the new input vector consisting of the vehicle's acceleration $\mathbf{a}$. For Example 2.5, the discretized admissible control region eventually becomes

$$a_i(k) \in \begin{cases} \{0, -0.4\} & \text{if } v_i(k) = v_{i,\max} \\ \{0.4, 0, -0.4\} & \text{if } v_{i,\min} \le v_i(k) \le v_{i,\max} \\ \{0.4, 0\} & \text{if } v_i(k) = v_{i,\min} \end{cases} \tag{12}$$

4.2. Object Acceleration: Problem P3

Assumption (vi) considers the object's speed to be a stochastic process. For certain applications, however, objects are moving with more or less constant speeds rather than effecting a random walk, as implied by assumption (vi). In this case one may consider the object's speed φ to be a measurable state variable described by the dynamic equation

$$\varphi(k+1) = \varphi(k) + \boldsymbol{\beta}(k), \tag{13}$$

with a stochastic acceleration $\boldsymbol{\beta}$ of known stationary distribution. Thus, adding (13) to the problem formulation of Section 2.3, we obtain the extended problem P3. The corresponding control strategy P3 reacts to real-time measurements of position $\chi(k)$ and, additionally, of speed $\varphi(k)$ of the object.

Certainly, the strategy of problem P1 may be applied to the case of object's stochastic acceleration, and a number of scenarios (Bauschert 1990) have shown that it performs satisfactorily in most situations. However, we have been able to construct some specific conflict situations where strategy P1's resulting behavior appears quite stupid under the new conditions. One of these scenarios will be presented in the following example.

4.3. An Example

We consider the same situation discussed in Example 2.5, but we include in the problem formulation the stochastic acceleration equation (13) for the φ_2-direction with $\beta_2 \in \{-0.4, 0, 0.4\}$ equally distributed. In other words, we supply the control strategy with the knowledge that the object moves upward or downward with stochastic acceleration rather than with stochastic speed.

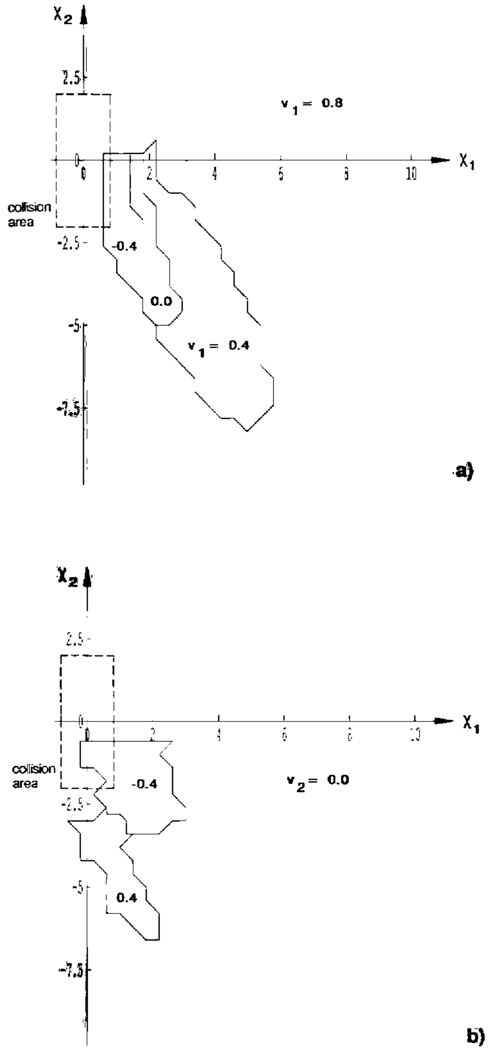

Fig. 9. Control strategy for problem P3 and $\varphi_2 = 1.2$.

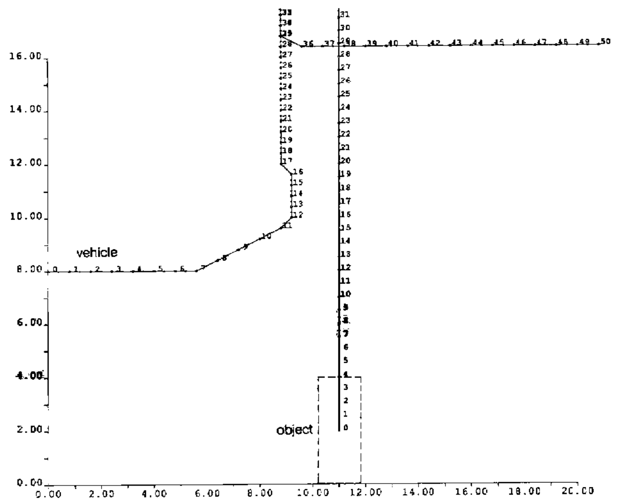

Fig. 10. P1 strategy applied to a constant speed object.

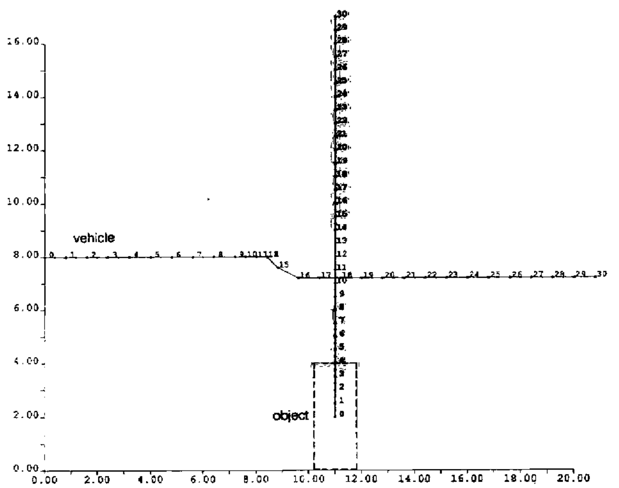

Fig. 11. P3 strategy applied to a constant speed object.

The control strategy that results from this extended problem P3 depends on three state variables: $\chi_1, \chi_2, \varphi_2$. Figure 9 depicts the corresponding speed regions for a fixed value $\varphi_2 = 1.2$. When compared with Figure 3, the P3 control strategy includes distorted control regions according to the object's current speed φ_2. Moreover, the deviation region for v_2 is reduced when compared with that in Figure 3*B*, probably owing to the fact that the vehicle feels safer under the new object speed conditions.

We will now present a particular conflict scenario for which application of the P1 strategy appears rather stupid. More precisely, we assume that the object is moving upward with a constant speed $\varphi_2 = 0.5$. Then, for some particular initial positions of the vehicle and the object, we obtain the behavior depicted in Figure 10. At time $k = 7$, the vehicle attempts to deviate to the left. However, the object moves upward with a higher speed and blocks the vehicle's way. The vehicle does decelerate but continues moving upward. The conflict is resolved only after the object has overtaken the vehicle, which now can pass on the right side. Obviously, the vehicle's behavior is due to its expectation of a randomly walking object rather than an object with constant speed.

Figure 11 depicts the results of application of the P3 strategy to the same conflict situation. These results demonstrate very reasonable vehicle behavior. Owing to the distorted regions of Figure 9, the vehicle does not deviate to the left, but drives straight ahead until it reaches a certain safety distance; then it waits (!) during times 11–14 for the object to pass before continuing its way with a slight deviation to the right.

5. Different Objects

5.1. *On the Notion of an Object*

We first note that the definition of an object, as expressed in (2), is a fairly general one. Essentially, equation (2) requires that the object's shape be constant in time but permits any kind of shape, including discontinuous ones. For example, the entire stationary environment of a mobile robot (e.g., a hall with furniture, etc.) or of a robot arm may be considered as one single object. Hence, the method as discussed so far is applicable to all these situations, provided the object's dimensions are constant and known. However, in the next section we will relax these two assumptions by replacing them with the requirement of measurability in real time.

5.2. *Changing Objects*

We will now consider objects with unknown and possibly time variant but measurable dimensions. This includes, for example, (1) objects that blow up or shrink, (2) corridors with changing but unknown dimensions (e.g., in two dimensions: corridors with changing but unknown width), (3) obstacles with different dimensions appearing on the vehicle's way, and (4) incompletely known objects (i.e., the knowledge of the shape of the objects may improve in real time).

Generally speaking, the unknown part $\mathbf{l}$ of the object's dimensions is modeled by introducing a stochastic equation

$$\mathbf{l}(k+1) = \mathbf{l}(k) + \boldsymbol{\lambda}(k), \tag{14}$$

with $\boldsymbol{\lambda}$ being a stochastic process with a given stationary probability distibution. For example 2.5, equation (14) may be used with $\boldsymbol{\lambda}(k) \equiv \mathbf{0}$ if the appearing object has a priori unknown but measurable dimensions, in which case we obtain an augmented state vector $\mathbf{x} = [\chi_1 \chi_2 l_1 l_2]^T$.

It should be noted that in all cases of changing dimensions, the relation $\mathbf{g}$ in (2) depends also on $\mathbf{l}$, which therefore enters the corresponding term in the performance criterion.

5.3. *More Objects: Problem P4*

Increasing the number of considered objects in the developed methodology is straightforward. Actually, it suffices to duplicate the corresponding equation (3) and, if necessary, (13) and/or (14). Needless to say, this augments the order of the state vector accordingly. Moreover, the shape relations $\mathbf{g}$ of the additional objects must be added to the performance criterion. This leads to an increased problem P4 of vehicle movement.

5.4. *An Example*

We will again consider example 2.5, but with two objects appearing in the vehicle's way, one of them being immobile. Moreover, and in distinction to the example 2.5, we restrict the vehicle's admissible speed region by $v_i(k) \in \{-0.8, 0, 0.8\}$, $i = 1, 2$. Note that collision avoidance with a stationary object may be considered either by adding the object's shape relation to the performance criterion or by restricting the admissible state region accordingly. The first approach was utilized in this example, although the second approach may lead to a reduction of the computational effort.

Certainly, a suboptimal, "one-object-at-a-time" strategy consists of applying P1 control, considering at each time k only one object (i.e., the one that is nearest or the one that is nearest ahead of the vehicle). As in any heuristic strategy, however, care should be taken for the strategy to remain efficient for all possibly occurring situations (Bauschert 1990).

Considering explicitly the two objects as outlined in Section 5.3, leads to a four-dimensional P4-control strategy $\mathbf{r}(\chi_1^I, \chi_2^I, \chi_1^{II} \chi_2^{II},)$, where upper indices I, II stand for the two distinct objects. Clearly, if one of the objects is sufficiently far, the P4 strategy looks very much like the

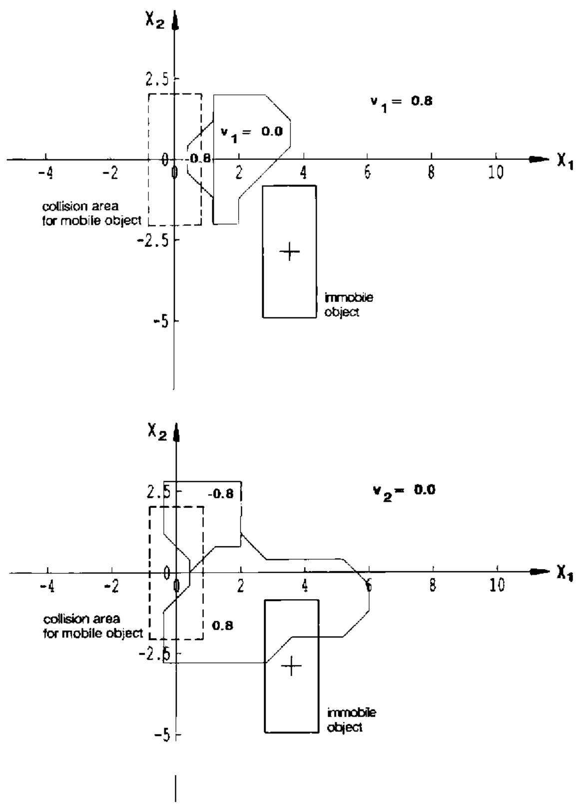

Fig. 12. P4 strategy (two objects) for fixed coordinates of the immobile object.

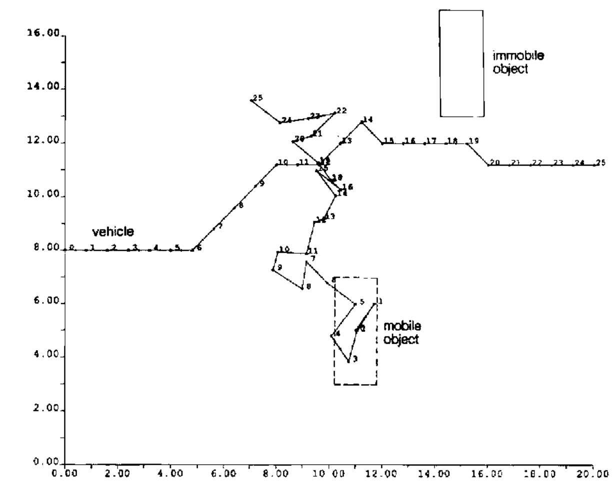

Fig. 13. A scenario with two objects applying the P4 strategy.

P1 strategy. Figure 12 depicts the P4 control strategy depending on χ_1^I, χ_2^I, (the coordinates of the mobile object) when the coordinates of the immobile object are fixed to $\chi_1^{II} = 3.6$, $\chi_2^{II} = -2.8$. The interpretation of these results is left to the reader.

Figure 13 depicts the results of a scenario selected out of a high number of experiments included in Bauschert (1990). Clearly, the P4 strategy is capable of performing some kind of slalom between the two objects in the scene, avoiding collisions in an elegant way.

6. Different Geometry

The only assumption of the basic problem that has not been relaxed or altered so far is assumption (i), which excluded consideration of the vehicle's and/or object's orientation. This assumption can now be relaxed in a fairly straightforward way. Denoting by π, $\sqcap \in \mathrm{R}^{p-1}$ the orientations of the vehicle and the object, respectively, the p-dimensional piece of space in the vehicle's coordinates that is occupied by the object and the vehicle is now given by

$$\Xi(k) = \{\xi \in \mathrm{R}^p | \mathbf{g}(\chi(k), \pi(k), \sqcap(k), \xi) \leq \mathbf{0}\}, \quad (15)$$

with $\mathbf{g}$ again being known functions to be included in the performance function. The vehicle's orientation $\pi(k)$ may be considered as an additional input variable, whereas the object's orientation $\sqcap(k)$ may be stochastically modeled by

$$\sqcap(k+1) = \sqcap(k) + \omega(k), \quad (16)$$

where $\omega(k)$ is a stochastic process with known stationary probability distribution.

7. Conclusions

We have presented a fairly general framework for control of moving objects in a dynamically changing environment. The basic approach relies on (1) availability of real-time measurements, (2) modeling of unknown future changes by the use of appropriate stochastic terms, and (3) applying stochastic optimization techniques for derivation of stationary control strategies.

Four examples of possible control formulations have been presented to demonstrate the flexibility of the approach: a basic problem of moving forward (P1), the problem of following a trajectory (P2), the case of measurable object speeds (P3), and the case of multiple objects (P4).

Some fairly complicated test scenarios have demonstrated the capability of the resulting control strategies to make efficient, human intelligence–like decisions. The main drawback of the approach is the computational effort required for derivation of the control strategy. Although this computational effort is only required off-line, it may become overwhelming for systems of order higher than 5 or 6. Nevertheless, the approach may be used directly for a broad class of practical problems, or it may provide a good basis for systematic development of heuristic strategies in high-order problem environments.

Acknowledgments

The authors would like to thank C. Fröhlich and P. Kampmann for useful discussions in the context of this research.

References

Bauschert, T. 1990. Verfahren der stochastischen Regelung und deren Anwendung. Diploma thesis 857, Technische Universität München, Lehrstuhl für Steuerungs- und Regelungstechnik.

Bertsekas, D. P. 1976. *Dynamic Programming and Stochastic Control.* New York: Academic Press.

Borenstein, J., and Koren, Y. 1990 (May 13–18, Cincinnati, OH). Real-time obstacle avoidance for fast mobile robots in cluttered environments. *Proc. 1990 IEEE Int. Conference on Robotics and Automation.* New York: IEEE, pp. 512–577.

Bourdé, M. 1990. Untersuchung eines Regelungskonzeptes für mobile Objekte in dynamischer Umgebung. Diploma thesis 865, Technische Universität München, Lehrstuhl für Steuerungs- und Regelungstechnik.

Brooks, R. A. 1983. Solving the find-path problem by good representation of free space. *IEEE Trans. Sys. Man Cybernet.* SMC-13:190–197.

Brooks, R. A., and Lozano-Pérez, T. 1983 (Aug. 8–12, Karlsruhe, FRG). A subdivision algorithm in configuration space for find-path with rotation. *Proc. 8th Int. Joint Conf. on Artificial Intelligence*, Vol. 2, pp. 799–806.

Chung, S. L., and Hanson, F. B. 1990 (Dec. 5–7, Honolulu, HI). Optimization techniques for stochastic dynamic programming. *Proc. 29th IEEE Conf. on Decision and Control*. New York: IEEE, pp. 2450–2455.

Erdmann, M., and Lozano-Pérez, T. 1986. On multiple moving objects. *Proc. 1986 IEEE Conference on Robotics and Automation*, pp. 1419–1424.

Fujimura, K., and Samet, H. 1989. A hierarchical strategy for path planning among moving obstacles. *IEEE Trans. Robot. Automation* 5:61–69.

Graglia, P., and Meystel, A. 1987 (Jan. 19–20, Philadelphia, PA). Planning minimum time trajectory in the transversality space of a robot. *Proc. IEEE Int. Symposium on Intelligent Control 1987*. New York: IEEE, pp. 82–87.

Griswold, N. C., and Eem, J. 1990. Control of mobile robots in the presence of moving objects. *IEEE Trans. Robot. Automation* 6:263–268.

Iyengar, S. S., Jorgensen, C. C., Rao, S. V. N., and Weisbin, C. N. 1986. Robot navigation algorithms using learned spatial graphs. *Robotica* 4:93–100.

Khatib, O. 1986. Real-time obstacle avoidance for manipulators and mobile robots. *Int. J. Robot Res.* 5:90–98.

Mycielski, I. 1988. Theories of pursuit and evasion. *J. Optimization Theory Applications* 56:271–284.