

# Evaluating Motion Strategies under Nondeterministic or Probabilistic Uncertainties in Sensing and Control

Steven M. LaValle

lvalle@robotics.stanford.edu

Dept. of Computer Science  
Stanford University  
Stanford, CA 94305

Seth A. Hutchinson

seth@uiuc.edu

Dept. of Elect. & Comp. Engineering  
Beckman Institute  
University of Illinois  
Urbana, IL 61801

## Abstract

*In this paper we provide a method for characterizing future configurations under the implementation of a motion strategy in the presence of sensing and control uncertainties. We provide general techniques which can apply to either nondeterministic models of uncertainty (as typically considered in preimage planning research) or probabilistic models. Information-space concepts from modern control theory are utilized to define the notion of a strategy in this general context. We have implemented algorithms and show several computed examples that generalize the forward projection concepts from tradition literature in this area.*

## 1 Introduction

It is important that a robotic system be able to accurately predict the outcomes of its actions when there is uncertainty in either execution (control) or sensing. In past research on manipulation planning, forward projections have been used for this purpose (e.g., [3, 7]).

Two basic representations of sensing and control uncertainty have been proposed in the manipulation planning literature. We refer to these as *nondeterministic uncertainty* and *probabilistic uncertainty*, as done in [5]. Under nondeterministic uncertainty, it is assumed that parameter uncertainties lie in a bounded set. This uncertainty representation is the most common in previous manipulation planning research (e.g., [7, 9, 10, 12]). Under probabilistic uncertainty, probability densities are used to represent uncertainty associated with parameters. This approach has been advocated for manipulation planning by Brost and Christiansen [2, 3, 4]. Each uncertainty representation offers advantages. For example, nondeterministic models do not require a statistical representation of the errors, and hence are often easier to specify.

In the remainder of the paper, we describe the specific models of uncertainty that we use (Section 2), describe nondeterministic forward projections and present computed examples (Section 3) and describe probabilistic forward projections and present computed examples (Section 4).

## 2 Modeling Uncertainty

As introduced in [11], we conceptualize the the robot's interaction with the environment as a dynamic game that is played between two players: the robot,  $\mathcal{A}$ , and nature. The robot has a general plan to achieve some goal, while nature makes some decisions that potentially interfere with  $\mathcal{A}$ . At an abstract level, this general view of robotic manipulation tasks has been advocated in [13]. In this section, we describe specific models for control and sensing uncertainties that we consider throughout this paper. The models that we use here can be specialized to those given in, e.g., [3, 7].

**The control model** Suppose the robot  $\mathcal{A}$  is a polygon translating in the plane amidst polygonal obstacles. The action set of  $\mathcal{A}$ , which we denote by  $U$ , is a set of commanded velocity directions, which can be specified by an orientation, yielding  $U = [0, 2\pi)$ . For a specified action,  $u_k$ , The robot will attempt to move a fixed distance  $\|v\|\Delta t$  (expressed in terms of a constant velocity modulus,  $\|v\|$ ) in the direction specified by  $u_k$ .

The action space of nature, which we denote by  $\Theta^\alpha$  is a set of angular displacements,  $\theta_k^\alpha$ , such that  $-\epsilon_\theta \leq \theta_k^\alpha \leq \epsilon_\theta$ , for some maximum angle  $\epsilon_\theta$ . Under nondeterministic uncertainty, any action  $\theta_k^\alpha \in [-\epsilon_\theta, \epsilon_\theta]$  can be chosen by nature. When using probabilistic uncertainty,  $p(\theta_k^\alpha)$  could be a continuous probability density function (pdf), which is zero outside of  $[-\epsilon_\theta, \epsilon_\theta]$ .

To describe the effect of a robot action with respect to state,  $x_k$ , we define a *state transition equation* as

$$x_{k+1} = f(x_k, u_k, \theta_k^\alpha). \quad (1)$$

Hence, given a robot action, nature's action, and the current state, the next state is deterministically specified. During execution, however,  $\mathcal{A}$  will not know the action of nature.

There are several cases to consider in defining the state transition equation,  $f$ . First consider the state transition equation when  $x_k \in \mathcal{C}_{free}$ , at a distance of at least  $\|v\|\Delta t$  away from the obstacles. If  $\mathcal{A}$  chooses action  $u_k$  from state

$x_k$ , and nature chooses  $\theta_k^a$ , then  $x_{k+1}$  is given by

$$f(x_k, u_k, \theta_k^a) = x_k + \|v\| \Delta t \begin{bmatrix} \cos(u_k + \theta_k^a) \\ \sin(u_k + \theta_k^a) \end{bmatrix}. \quad (2)$$

Let  $C_{\text{contact}}$  represent the boundary of  $C_{\text{free}}$  (hence  $C_{\text{contact}} = C_{\text{valid}} - C_{\text{free}}$ ). If  $x_k \in C_{\text{contact}}$ , with a distance of at least  $\|v\| \Delta t$  from the edge endpoints, then a compliant motion is generated by using the generalized damper model (see e.g., [14]) for certain choices of  $u_k$ . If  $u_k$  points into the obstacle edge with a sufficient angle to overcome friction, then the robot moves a fixed distance parallel to the edge. Otherwise, the robot either remains fixed, or moves away into  $C_{\text{free}}$ . The remaining cases describe when the robot moves from  $C_{\text{free}}$  to  $C_{\text{contact}}$ , from  $C_{\text{contact}}$  to  $C_{\text{free}}$ , or from one edge in  $C_{\text{valid}}$  to another. These cases are straightforward to define with the generalized damper model, see e.g., [7, 10]

**The sensing model** A sensor can be viewed as a mapping from states onto sensor values with potential interference that is caused by nature. We characterize the potential interference of nature by a set of ‘‘sensing action’’ denoted by  $\Theta^s$ . At every stage  $k$ , the robot makes an *observation* that is governed by the equation,

$$y_k = h_k(x_k, \theta_k^s), \quad (3)$$

which we term the *observation equation*, in which  $\theta_k^s \in \Theta^s$ .

Suppose that we are considering nondeterministic uncertainty. The set of possible values for  $x_k$  after only observing  $y_k$  can be determined from the observation equation as:

$$F_k(y_k) = \{x_k \in X | y_k = h(x_k, \theta_k^s), \theta_k^s \in \Theta^s\}. \quad (4)$$

Under probabilistic uncertainty, we assume that the pdf,  $p(\theta_k^s)$ , is known. By using the observation equation, we can obtain a pdf for  $x_k$ , which is represented by  $p(x_k | y_k)$ . As a simple example,  $h$  could represent a position sensor that measures  $x_k$  with Gaussian noise. If  $h(x_k, \theta_k^s) = x_k + \theta_k^s$ , and  $p(\theta_k^s)$  is a Gaussian density, then  $p(x_k | y_k)$  is Gaussian.

If  $Y = X$  and  $h_k$  is reduced to the identity map from  $X$  to  $Y$ , then the sensing model reduces to perfect state information. Equation (3) represents the output equation used in control theory, and is also similar to the projection of world states onto sensor values. Also, such transformations have been studied extensively in statistical image modeling and in sensor error modeling.

We now present a sensing model that is similar to that used in [3, 7, 10]. This sensing model will be used in Section 5. The robot  $\mathcal{A}$  is equipped with a position sensor and a force sensor. Assume that the position sensor is calibrated in the configuration space, yielding values in  $\mathbb{R}^2$ . The force sensor provides values in  $[0, 2\pi) \cup \{\emptyset\}$ , indicating either the direction of force, or no force (represented by  $\emptyset$ ).

We consider independent portions of the observation equation:  $h^p$  for the position sensor, and  $h^f$  for the force sensor (which together form a 3-dimensional vector-valued function). We partition the sensing action of nature,  $\theta_k^s$  into subvectors  $\theta_k^{s,p}$  and  $\theta_k^{s,f}$ , which act on the position sensor and force sensor, respectively. The observation for the position sensor is  $y_k^p = h^p(x_k, \theta_k^{s,p}) = x_k + \theta_k^{s,p}$ . Under nondeterministic uncertainty,  $\theta_k^{s,p}$  could be any value in  $\Theta_k^{s,p}$ . If probabilistic uncertainty is used, we could provide a density for nature as

$$p(\theta_k^{s,p}) = \begin{cases} \frac{1}{\pi \epsilon_p^2} & \text{for } \|\theta_k^{s,p}\| < \epsilon_p \\ 0 & \text{otherwise} \end{cases}, \quad (5)$$

for some prespecified radius  $\epsilon_p$ , and  $\theta_k^{s,p}$  is 2-dimensional.

For the force sensor we obtain either: 1) A value in  $[0, 2\pi)$ , governed by  $y_k^f = h^f(x_k, \theta_k^{s,f}) = \alpha(x_k) + \theta_k^{s,f}$ , in which  $x_k \in C_{\text{contact}}$ , and the true normal is given by  $\alpha(x_k)$ , or 2) An empty value,  $\emptyset$ , when the robot is in  $C_{\text{free}}$ . When the robot configuration lies in  $C_{\text{contact}}$  and probabilistic uncertainty is in use, then we might choose

$$p(\theta_k^{s,f}) = \begin{cases} \frac{1}{2\epsilon_f} & \text{for } |\theta_k^{s,f}| < \epsilon_f \\ 0 & \text{otherwise} \end{cases}, \quad (6)$$

for some positive prespecified constant  $\epsilon_f < \frac{1}{2}\pi$ . We consider the random variables of  $\theta_k^{s,p}$  and  $\theta_k^{s,f}$  to be independent and identically distributed over all stages.

**Information State** For a given stage  $k$ , let  $\eta_k$  denote some subset:

$$\eta_k \subseteq \{u_1, u_2, \dots, u_{k-1}, y_1, y_2, \dots, y_k\}. \quad (7)$$

The value  $\eta_k$  is a set of past actions and observations that are known to  $\mathcal{A}$  at stage  $k$ , and is termed the *information state* of  $\mathcal{A}$ . For instance, we could consider a memoryless robot, in which  $\eta_k = y_k$ . As another example, we could have a sensorless robot as considered, in which  $\eta_k = \{u_1, \dots, u_{k-1}\}$ . We could also consider the stage index  $k$  as part of the information space for the purpose of developing robot strategies that involve timing; however, we will not explicitly consider  $k$  as part of  $\eta_k$  in this paper.

The set of values that  $\eta_k$  can assume is denoted by  $N_k$ , and is termed the *information space*. We define an *information structure* as the set of  $N_k$  for all  $1 \leq k \leq K$ . As it is presently defined, the dimension of the information space grows linearly with the number of stages, which appears impractical. It turns out that alternative representations of the information space can be determined.

**Motion strategies** At first it might seem appropriate to define some action  $u_k$  for each stage. In general, due to the control uncertainty, it is not possible to predict the trajectory of the robot for given motion commands. It is therefore advantageous to allow the robot to respond to information that becomes available during execution.

We consider robot strategies for two cases: perfect information and imperfect information. Suppose that the robot has perfect state information. We can implement a *state-feedback strategy at stage  $k$*  as a function  $g_k : X \rightarrow U$ . For each state,  $x_k$ , a strategy yields an action  $u_k = g_k(x_k)$ . The set of mappings  $\{g_1, g_2, \dots, g_K\}$  is denoted by  $g$  and termed a (robot) *strategy* of  $\mathcal{A}$ .

If the robot does not have direct access to state information, its actions are instead conditioned on the information state. In this case we define a *strategy at stage  $k$*  of  $\mathcal{A}$  as a function  $g_k : N_k \rightarrow U$ . For each information state,  $\eta_k$ , a strategy yields an action  $u_k = g_k(\eta_k)$ . In a sense, the “planning” actually occurs in this information space. These strategy concepts are equivalent to a feedback control law [1, 8], and are similar to a conditional multi-step plan in manipulation planning [10].

We also define a strategy,  $\gamma^\theta$ , for nature. Since nature is considered as a decision maker that can interfere with the robot, we allow nature’s actions to depend in general on the state,  $x_k$ , and the action of the robot,  $u_k$ . We can define a *pure* or *deterministic* strategy for nature as a deterministic mapping at each stage as  $\gamma_k^\theta : X \times U \rightarrow \Theta^a$ . Under nondeterministic uncertainty we will assume that nature implements a deterministic strategy that is unknown to the robot. We will use the notation  $\Gamma^\Theta$  to refer to the space of strategies that are available to nature under nondeterministic uncertainty.

Under probabilistic uncertainty, we consider a *randomized* or *mixed* strategy for nature, in which the action of nature is represented by a pdf,  $p(\theta_k)$  (or we can more generally consider  $p(\theta_k|x_k, u_k)$ ). The specific action of nature at stage  $k$  is denoted by  $\theta_k$ , sampled from the random variable  $\Theta_k$ . Therefore the robot is given a pdf,  $p(\theta_k)$ , that characterizes the action taken by nature at stage  $k$ . Although the randomized *strategy* is known by the robot, the *actions* that will be chosen are sampled from a random variable at each stage.

### 3 Nondeterministic Forward Projections

Under nondeterministic uncertainty, the strategy of nature  $\gamma^\theta$  is deterministic, but unknown to the robot. The resulting nondeterministic forward projection includes all of the system states that could result from the various actions of nature (including nature’s sensing actions). In this way, it yields a set of possible futures under the implementation of a strategy.

#### 3.1 The perfect information case

We use the notation  $F_j(x_i, g)$  to denote the minimal subset of  $X$  that is guaranteed to contain  $x_j$ , if the system begins in state  $x_i$  at stage  $i$  and strategy  $g$  is implemented up to stage  $j$ .

Assume that some  $g$  is given, and that at stage  $k$ , the state,  $x_k$ , is known. The action taken by the robot at stage  $k$  is known to be  $u_k = g_k(x_k)$ . Therefore we can write

$$F_{k+1}(x_k, g) = F_{k+1}(x_k, g_k(x_k)) = F_{k+1}(x_k, u_k), \quad (8)$$

in which  $F_{k+1}(x_k, u_k)$  is given by

$$F_{k+1}(x_k, u_k) = \{f(x_k, u_k, \theta_k^a) \in X | \theta_k^a \in \Theta^a\}. \quad (9)$$

Although the action is known, the resulting next state  $x_{k+1}$  is nondeterministic because of nature,  $\theta_k^a \in \Theta^a$ .

Suppose that we wish to determine the outcome at stage  $x_{k+2}$ , if we know  $x_k$ . From (9), we already know that  $x_{k+1} \in F_{k+1}(x_k, u_k)$ . The nondeterministic action of nature at stage  $k+1$  must next be taken into account to yield

$$F_{k+2}(x_k, g) = \quad (10)$$

$$\{f(x_{k+1}, u_{k+1}, \theta_{k+1}^a) \in X | x_{k+1} \in F_{k+1}(x_k, g), \theta_{k+1}^a \in \Theta^a\}. \quad (11)$$

This forward projection can also be expressed with a set union as

$$F_{k+2}(x_k, g) = \bigcup_{x_{k+1} \in F_{k+1}(x_k, g)} F_{k+2}(x_{k+1}, g). \quad (13)$$

One interpretation for this representation is that from each possible state in the single-stage forward projection from stage  $k$  to stage  $k+1$ , the single-stage forward projection from stage  $k+1$  to stage  $k+2$  is possible. The resulting  $x_{k+2} \in X$  represents the union of all of these single-stage forward projections (see Figure 1).

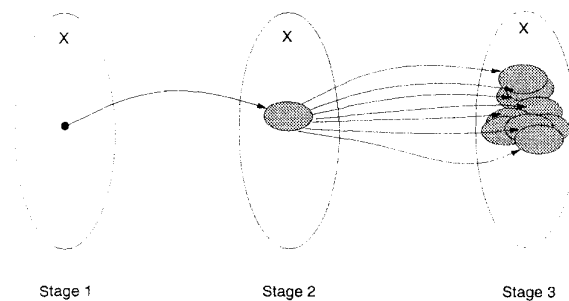


Figure 1. A depiction of a two-stage forward projection under nondeterministic control uncertainty.

The forward projection for a finite number of stages from stage 1 can be considered as an iterated union,

$$F_k(x_1, g) = \bigcup_{x_2 \in F_2(x_1, g)} \dots \bigcup_{x_{k-1} \in F_{k-1}(x_{k-2}, g)} F_k(x_{k-1}, g), \quad (14)$$

which is an extension of (13). The projection from any stage  $k$  to stage  $k+N$  can be similarly defined.

The classical reachability concept [6] can be defined using our framework. We say that the goal is *reachable at stage  $k$*  if  $F_k(x_1, g) \subseteq G$ . In other words, if the strategy is guaranteed to bring the robot into the goal region for some  $k$ , then reachability at stage  $k$  holds. We can also define a reachability that does not depend  $k$ . We can say that the goal is *reachable* if for every possible state trajectory,  $\{x_1, \dots, x_{K+1}\}$  (under the implementation of a given  $g$ ), there exists a  $k$  such that  $x_k \in G$ .

### 3.2 The imperfect information case

We consider, as in the perfect information case, a deterministic strategy for nature,  $\gamma^\theta$ , which is unknown to the robot. We will define the forward projection in a manner similar to the perfect information case.

The previous forward projection (14) provided a subset of  $X$  in which the system state will lie after the execution of a strategy. With imperfect sensing we can consider the motions to occur in the information space. In fact, we can consider the information space as a new “state space” in which there is perfect “state” information. For this reason, a forward projection can also be defined directly on the information space.

It is assumed for the forward projection that the history has not yet been given. Suppose that an information state,  $\eta_k \in N_k$ , is given. Under the implementation of  $g$ , the action  $u_k = g_k(\eta_k)$  is known.

We now define the *information forward projection* for a single stage. This will be an intermediate concept that is used to define the forward projection as a subset of the state space. We have previously used  $F$  to represent a subset of  $X$ , and we will use  $\tilde{F}$  to refer to a subset of the information space. After applying an action  $u_k$  and receiving sensor observation  $y_{k+1}$ , we obtain

$$\begin{aligned} \tilde{F}_{k+1}(\eta_k, g(\eta_k)) &= \tilde{F}_{k+1}(\eta_k, u_k) = \\ &\{\eta_{k+1} \in N_{k+1} | \eta_k \cup \{u_k, y_{k+1}\} \subset \eta_{k+1}, \\ &x_{k+1} \in F_{k+1}(x_k, u_k) \cap F_{k+1}(y_{k+1}), \\ &x_k \in F_k(\eta_k)\}, \end{aligned} \quad (15)$$

which depends on (9) and (4), and  $F_k(\eta_k) \subseteq X$  is the subset representation of the information state from Section 2.

To obtain the information forward projection from stage 1 to some stage  $k$ , we can iteratively apply (15).

The information forward projection can be mapped to subsets of the state space. For a given  $\tilde{F}_k(\eta_1, g)$ , the subset of  $X$  in which the system state will lie is

$$\bigcup_{\eta_k \in \tilde{F}_k(\eta_1, g)} F_k(\eta_k). \quad (16)$$

The goal is *reachable at stage  $k$*  if set defined in (16) is a subset of  $G$ .

## 4 Probabilistic Forward Projections

Under nondeterministic uncertainty, the forward projections yielded subsets of the state space; however, for probabilistic uncertainty, the forward projections will be specified by pdf's on the state space. We use the notation  $p(x_j | x_i, g)$  in this section to represent the density that is obtained if the system begins at state  $x_i$  at stage  $i$  and strategy  $g$  is implemented. This density follows directly from the state transition equation, and the densities for nature of the form,  $p(\theta_k^g)$ .

### 4.1 The perfect information case

The following development parallels the development of the forward projection in Section 3.1. Assume that some

$g$  is given, and that at stage  $k$ , the state,  $x_k$ , is known. The action taken by the robot at stage  $k$  is known to be  $u_k = g_k(x_k)$ . Therefore we can write

$$p(x_{k+1} | x_k, g) = p(x_{k+1} | x_k, g_k(x_k)) = p(x_{k+1} | x_k, u_k). \quad (17)$$

Recall from Section 2 that  $p(x_{k+1} | x_k, u_k)$  can be determined from the state transition equation.

Next consider predicting the outcome at stage  $k+2$ , if we begin at stage  $k$  and apply  $g$ :

$$p(x_{k+2} | x_k, g) = \int p(x_{k+2} | x_{k+1}, g_{k+1}(x_{k+1})) p(x_{k+1} | x_k, g_k(x_k)) dx_{k+1} \quad (18)$$

The result after applying two actions is a posterior density on  $X$ . Figure 2 depicts the forward projection; this can be contrasted to Figure 1, which showed the forward projection under nondeterministic uncertainty.

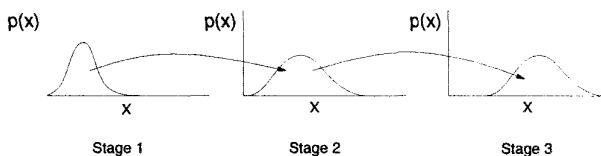


Figure 2. A depiction of a two-stage forward projection under probabilistic control uncertainty.

The forward projection for a finite number of stages from stage 1 results in the posterior:

$$p(x_k | x_1, g) = \int p(x_k | x_{k-1}, g_{k-1}(x_{k-1})) p(x_{k-1} | x_{k-2}, g_{k-2}(x_{k-2})) \cdots p(x_2 | x_1, g_1(x_1)) dx_2 dx_3 \cdots dx_{k-1}. \quad (19)$$

The projection from any stage  $k$  to stage  $k+N$  can be similarly defined.

We can now define probabilistic notions of reachability. The probability that the goal is reached at stage  $k$  is given by

$$\int_G p(x_k | x_1, g) dx_k, \quad (20)$$

in which the region of integration is the goal region,  $G \subseteq X$ .

### 4.2 The imperfect information case

In this section we develop the forward projections for the case in which there is probabilistic uncertainty in both sensing and control. The forward projection for this case will be considered as a density on  $X$ , which is conditioned on a particular strategy and initial state (either  $x_1$  or  $\eta_1$ ). This density indicates where the robot will be likely to end up when a fixed  $g$  is implemented, at some specified stage. Note that we could also derive  $p(\eta_k | \eta_1, \gamma)$ , resulting in a pdf on the information space.

At stage  $k$ , the density on  $X$  after starting at  $\eta_1$  is given by

$$p(x_k|\eta_1, g) = \int p(x_k|\eta_{k-1}, g_{k-1}(\eta_{k-1}))p(\eta_{k-1}|\eta_{k-2}, g_{k-2}(\eta_{k-2})) \cdots p(\eta_2|\eta_1, g_1(\eta_1))d\eta_{k-1} \cdots d\eta_2. \quad (21)$$

The first term in the integrand can be determined using

$$p(x_{k+1}|\eta_k, u_k) = \int p(x_{k+1}|x_k, u_k)p(x_k|\eta_k)dx_k. \quad (22)$$

Each of the remaining terms can be reduced to

$$\begin{aligned} p(\eta_{k+1}|\eta_k, g_k(\eta_k)) &= p(y_1, \dots, y_{k+1}, u_1, \dots, u_k | y_1, \dots, y_k, u_1, \dots, u_k) \\ &= p(y_{k+1}|\eta_k, u_k). \end{aligned} \quad (23)$$

This reduction occurs because most of the sensing and action history appears on both sides of the density expression. The right side of (23) can be further reduced to

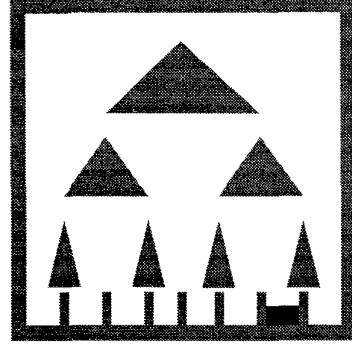
$$\begin{aligned} p(y_{k+1}|\eta_k, u_k) &= \int p(y_{k+1}|x_{k+1})p(x_{k+1}|\eta_k, u_k)dx_{k+1} = \\ &= \int \int p(y_{k+1}|x_{k+1})p(x_{k+1}|x_k, u_k)p(x_k|\eta_k)dx_k dx_{k+1} \end{aligned} \quad (24)$$

in which all three terms in the final integrand are known. The density  $p(y_{k+1}|x_{k+1})$  is inferred from the sensing model;  $p(x_{k+1}|x_k, u_k)$  is inferred from the control model;  $p(x_k|\eta_k)$  is the density representation of the current information state.

## 5 Computed Examples

In this section we present computed examples that illustrate the forward projection concepts. These forward projections are provided under the assumption that constant motion commands are given to the robot. In other words, some  $u \in U$  is chosen, and a strategy is defined as  $g_k \equiv u$  for all  $k \in \{1, \dots, K\}$ . This will make the comparison of our forward projections to previous research more clear.

We have computed forward projection examples in a straightforward way, by using a discretized, array representation for the state space. Under nondeterministic uncertainty, this can be considered as a bit-map representation of the forward projection. Under probabilistic uncertainty, the representation approximates a pdf on  $X$  by using a fine grid. In the first step of the computation, the array is initialized to reflect the uncertainty associated with the initial state. At each additional step, the forward projection for the next stage is represented in a new array, which is determined by applying the given strategy to the elements in the previous array (in the implementation, only two copies of the array are needed at any given time).



**Figure 3.** The example environment that is considered in this section. The obstacles in the workspace are indicated by gray regions, and the black region represents the goal.

We have found this computational technique to produce reasonable representations of forward projections.

The example depicted in Figure 3 is designed to spread the possible locations of the robot over a large portion of the state space. We use the control model that is discussed in Section 2. The given strategy is  $g_k \equiv \frac{3}{2}\pi$  for all  $k \in \{1, \dots, K\}$  (i.e., move down).

Figure 4 shows the forward projections at several different stages, under nondeterministic uncertainty. Figure 5 illustrates the forward projection under probabilistic uncertainty. For these examples, we assume that  $p(\theta^a)$  is uniform on the interval  $[-\epsilon_\theta, \epsilon_\theta]$ . Initially, the pdf is sharply peaked; however, as control uncertainty accumulates, the density becomes more diffuse. Whenever compliance is possible, the density becomes narrower in the direction perpendicular to the edge. The compliant motions have the effect of “funneling” the probability mass into smaller regions. The pdf values become larger since the density must integrate to one. This effect can be seen in Figure 5 as a triangular obstacle causes the probability mass to divide. In the final stages, there is also a peaking effect; this corresponds to the robot sticking at some final state. Maximizing the probability that the goal will be achieved can be thought of as causing as much of the probability mass to stay in the goal as possible.

A significant distinction between probabilistic and nondeterministic forward projections becomes clear after examining these results. The nondeterministic forward projections indicate that very little prediction is possible since the set of possible states grows very quickly. By observing the probabilistic forward projection, however, most of the probability mass appears to terminate in the goal region. This corresponds closely to the arguments about worst-case analysis eliminating many reasonable motion plans; these arguments were given in [11] and also in [3, 4].

## 6 Conclusion

In this paper, we have presented forward projections as a construction that can be used to characterize the performance of a robot system operating in the presence of

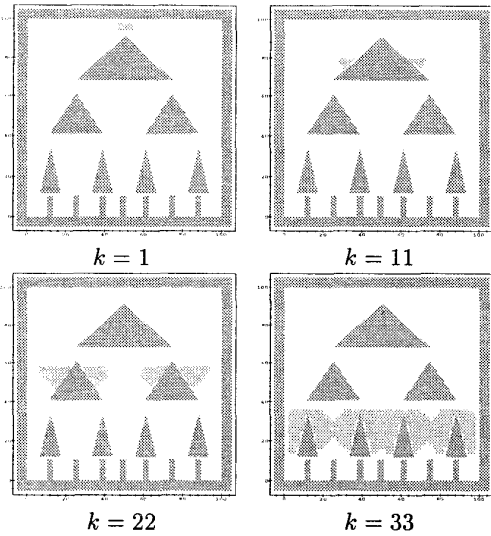


Figure 4. The nondeterministic forward projection is represented by the lightly-shaded regions.

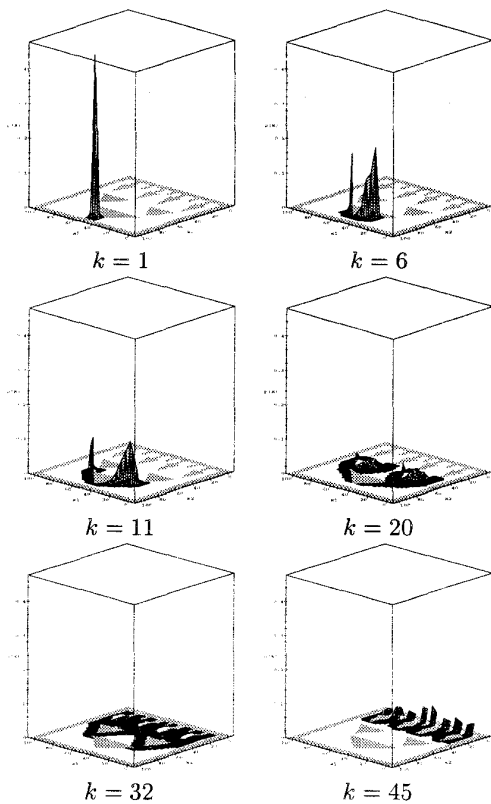


Figure 5. The forward projection at several stages, under probabilistic uncertainty.

uncertainties in both sensing and control. More specifically, we have developed formalisms for nondeterministic forward projections and for probabilistic forward projections, and presented computed examples of each.

### Acknowledgments

We gratefully acknowledge Narendra Ahuja, Tamer Başar, Randy Brost, Becky Castaño, Mike Erdmann, Peter Leven, Kevin Nickels, Jean Ponce, Rajeev Sharma, Steve Sullivan, and Mark Spong, for their helpful comments and suggestions. This work was sponsored by NSF under grant #IRI-9110270, a Beckman Institute research assistantship, and a Mavis Fellowship.

### REFERENCES

- [1] D. P. Bertsekas. *Dynamic Programming: Deterministic and Stochastic Models*. Prentice-Hall, Englewood Cliffs, NJ, 1987.
- [2] R. C. Brost. *Analysis and Planning of Planar Manipulation Tasks*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, 1991.
- [3] R. C. Brost and A. D. Christiansen. Probabilistic analysis of manipulation tasks: A research agenda. In *IEEE Int. Conf. Robot. & Autom.*, volume 3, pages 549–556, 1993.
- [4] R. C. Brost and A. D. Christiansen. Probabilistic analysis of manipulation tasks: A computational framework. Technical Report SAND92-2033, Sandia National Laboratories, Albuquerque, NM, January 1994.
- [5] M. Erdmann. Randomization in robot tasks. *Int. J. Robot. Res.*, 11(5):399–436, October 1992.
- [6] M. Erdmann. Randomization for robot tasks: Using dynamic programming in the space of knowledge states. *Algorithmica*, 10:248–291, 1993.
- [7] M. A. Erdmann. On motion planning with uncertainty. Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, August 1984.
- [8] P. R. Kumar and P. Varaiya. *Stochastic Systems*. Prentice-Hall, Englewood Cliffs, NJ, 1986.
- [9] J.-C. Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, Boston, MA, 1991.
- [10] J.-C. Latombe, A. Lazanas, and S. Shekhar. Robot motion planning with uncertainty in control and sensing. *Artif. Intell.*, 52:1–47, 1991.
- [11] S. M. LaValle and S. A. Hutchinson. An objective-based stochastic framework for manipulation planning. In *Proc. IEEE/RSJ/GI Int'l Conf. on Intelligent Robots and Systems*, pages 1772–1779, September 1994.
- [12] T. Lozano-Pérez, M. T. Mason, and R. H. Taylor. Automatic synthesis of fine-motion strategies for robots. *Int. J. Robot. Res.*, 3(1):3–24, 1984.
- [13] R. H. Taylor, M. T. Mason, and K. Y. Goldberg. Sensor-based manipulation planning as a game with nature. In *Fourth International Symposium on Robotics Research*, pages 421–429, 1987.
- [14] D. Whitney. Force feedback control of manipulator fine motions. *Trans. ASME J. of Dyn. Sys., Meas., & Contr.*, 99:91–97, 1977.