Proceedings of the 1991 IEEE
International Conference on Robotics and Automation
Sacramento, California - April 1991

Sensor Interpretation and Task-Directed Planning Using Perceptual Equivalence Classes

Bruce Donald* **Jim Jennings†**

Cornell Computer Science Department

Ithaca, NY

Abstract

We consider how a robot may interpret its sensors and direct its actions so as to gain more information about the world, and to accomplish manipulation tasks. The key difficulty is uncertainty, in the form of noise in sensors, error in control, and unmodelled or unknown aspects of the environment. Our research focuses on general techniques for coping with uncertainty, specifically, to sense the state of the task, adapt to changes, and reason to select actions to gain information and achieve the goal.

Sensors yield partial information about the world. When we interrogate the environment through our sensors, we in effect view a projection of the world onto the space of possible sensor values. We investigate the structure of this sensor space and its relationship to the world. We observe that sensors partition the world into perceptual equivalence classes, that can serve as natural "landmarks." By analyzing the properties of these equivalence classes we develop a "lattice" and a "bundle" structure for the information available to the robot through sensing and action. This yields a framework in which we develop and characterize algorithms for sensor-based planning and reasoning.

1 Introduction

In recent years, robotics researchers have focused considerable attention on the key problem of uncertainty (at many levels): for example, error or noise in modelling, actuation, and sensing [LMT, Mason, DW, Brooks, Mat, Brost, Buc, Caine, Can, E89, Don, LLS]. The domain of applicability for strategies for reducing uncertainty, gaining information, and directing actions to accomplish manipulation tasks is very broad. It includes (for example) *manipulation and mechanical assembly, mobile robot navigation and planning,* and *design and layout.*

In assembly, we may have sensing and control uncertainty, and sparse or incomplete environmental models. Work on synthesizing assembly strategies is particularly applicable during fine-motion and tight assemblies. Mobile robots may have noisy sensors and operate in dynamic, uncertain environments. In design for assembly, we wish to develop a physical device that will be functional in a variety of physical situations; this "variety" may be viewed as a parametric form of uncertainty.

Our main idea is as follows. For reasons of algorithmic tractability, much of our previous work has concentrated on planning (and execution) that emphasized *reachability* (the places a robot can reach during an action) while using very simple models of *recognizability* (the places a robot can distinguish using sensors). Erdmann first introduced this mathematical distinction in [Erd]. Indeed, some foundational work has indeed assumed that recognizability (e.g., recognizing attainment of a goal during a continuous control regime) was given by some "oracle" and that only reachability need be considered [Don90,E89,Bri,FHS]. This effectively reduces planning to a backwards simulation (under control uncertainty), which is an easier (although important) problem.

However, due to the sensitivity of simulation to modelling errors, this focus on reachability tightly coupled the planning process to relatively accurate models of the environment. To consider sparse or incomplete models of the environment, or mobile robots operating in very uncertain and dynamic domains, we take a different tack. In very uncertain domains, reachability computations (i.e., forward- and back-simulations) are essentially valid only locally, since they can only rely on local models of the world. We are trying to develop a framework for sensing and action in which recognizability is the "senior" partner. To say this precisely: in previous work, we attempted in effect to "reduce" recognizability computations to reachability computations. Hence, the fundamental geometric building blocks for this theory were the reachable sets (from some initial conditions, under particular controls). Since these sets had considerable algebraic and geometric structure, geometric algorithms could be obtained to generate control and sensing strategies—plans, if you will.

In this paper we propose a theory of planning, sensing and action in which the fundamental building blocks are, in effect, the "recognizable sets"—that is, the places in the world that the robot can recognize and distinguish between. To this end we observe that viewing the world through sensors partitions the world (locally) into "perceptual equivalence classes."[1] The more information the sensors provide, the "finer" this partition is. Various possible partitions of the world fit into a lattice structure (where one partition is "higher" than another if it is finer). This lat-

*This paper describes research done in the Computer Science Robotics Laboratory at Cornell University. Support for our robotics research is provided in part by the National Science Foundation under grants No. IRI-8802390, IRI-9000532 and by a Presidential Young Investigator award, and in part by the Mathematical Sciences Institute and AT&T Bell laboratories.
†Supported in part by an NSF graduate fellowship.

[1] This term was suggested to us by Dan Huttenlocher.

tice structure captures the information or knowledge state about the world; the lattice is related to the "version space" of Mitchell [Mit]. The recognizable sets are related to the "signature neighborhoods" in [MC]. They are called "sensor equivalence classes" in [CMM], and "perceptual aliases" in [BW, S].

Hence, a particular partition of the world into perceptual equivalence classes is an entry in the lattice. Sensing actions that "gain information" allow us to *ascend* to a "higher" (hence finer) partition in the lattice. Motion of the robot (in general) may induce a move between perceptual equivalence classes in the *same* partition. If the motion "gains" information, then we ascend in the lattice as well. Of course, it is possible for actions (and even sensing operations) to "lose" information (*descend* to a more ambiguous partition) as well. For example, if the robot is touching a wall and breaks contact, then the uncertainty in the robot's position may be increased.

Using this theory we develop notions of task-directed planning and show how history can be used to direct actions and gain information. We investigate the structure of the recognizability lattice and its relationship to the world. We consider how to compute perceptual equivalence classes both from a geometric world model,[2] and without a model, incrementally, *from* the world. Finally, we'll investigate mathematical properties that can help the robot generate disambiguating strategies to gain information about the world to accomplish a task. We believe our framework is useful in understanding, programming, and planning for other mobots, such as Mataric's [Mat] robot Toto as well. We are implementing our theories on a mobile autonomous platform equipped with sensors and several Cornell Generic Controllers in a ring-network [OdD; BSD]. Connections with experimental work appear in the full version of this paper, [DJ].

2 Perceptual Equiv. Classes

2.1 Quotient Structure

To illustrate our ideas we will develop a general albeit somewhat simplified model of sensing. In a particular, we will develop examples of perceptual equivalence classes and their structure. On a real mobot our examples generalize, but are more complicated [DJ].

One key observation is that there are places in the environment that are distinct, yet appear the same when viewed through the robot's sensors. For example, consider an (idealized) mobile robot with a range sensor.[3] Suppose the range sensor only outputs discrete range values, so that objects d length units away register as value $\lfloor d \rfloor$. Now, we'll model the robot's sensing operations as follows. Suppose the robot is at a fixed configuration x, and it points the range sensor in direction θ (which it knows). See fig. 1. Consider a point y in the world, which is in direction θ from x. We imagine the robot "interrogating" y. We can represent the sensor "value" for y as being one of the following (see fig. 2):

1. An integer $\lfloor d \rfloor$, if there is an obstacle at y and there are no obstacles on the line segment $[x, y)$.

[2] By *world model* we mean an *a priori* geometric representation of the robot's environment; for example, a CAD solid model provided by a human. Our robots could never build such a model, since it would contain imperceptible and hence unconstructible features. In particular, our *B-encoded RR-graph* (below) is not a world model, but a knowledge-representation of sensor and control history.

[3] There are other ways to present this example, of course.

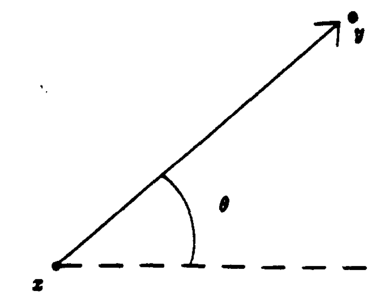

Figure 1: A mobile robot at configuration x is pointing its range finder in orientation θ to interrogate a point y in the environment.

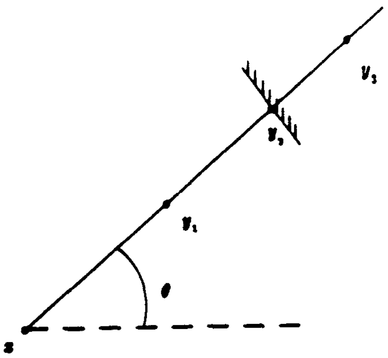

Figure 2: Consider points y_i in the world, which are in direction θ from x. We imagine the robot "interrogating" y_i by pointing the sensor in direction θ. In this example, there is an obstacle at y_2, which is distance d from x. We have $p_x(\theta, y_1) = \emptyset$, $p_x(\theta, y_2) = \lfloor d \rfloor$, $p_x(\theta, y_3) = \infty$.

2. The value $\emptyset$ if there are no obstacles on the line segment $[x, y]$.
3. The value ∞ if there is another obstacle on the line segment $[x, y)$.

If Z^+ denotes the non-negative integers, we can view the space B of sensor values as

$$B = Z^+ \cup \{\emptyset, \infty\}. \qquad (1)$$

Hence we can define our sensor as a mapping $p_x(\theta, y) = b$ for b in B. It is clear that this mapping induces a partition of the world into equivalence classes of points that will appear the same under the sensor. These equivalence classes are conceptually sketched in fig. 3. If we disregard θ, two obstacles at roughly distance d will appear the same, even if they are at different angular locations. (The reader should not be alarmed; we will take θ into account later and use it to discriminate among the equivalence classes in fig. 3.) In this case, the perceptual equivalence classes are the unit annuli centered on x with integral inner and outer radii. (We view the unit disk centered on x as a kind of degenerate annulus).

Now, it is clear how the equivalence classes in fig. 3 arise. The equivalence classes form a partition of the world (points y in M). The equivalence relation is that the points yield the same sensor reading when viewed or "projected" through p_x; compare [C]. Finally, it is clear that p_x is a

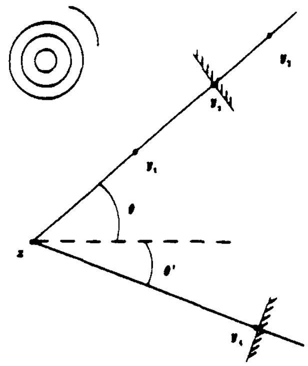

Figure 3: If we disregard θ, two obstacles at roughly distance d will appear the same, even if they are at different angular locations. Hence, since $p_x(y_2,\theta) = p_x(y_4,\theta')$, y_2 and y_4 are in the same perceptual equivalence class.

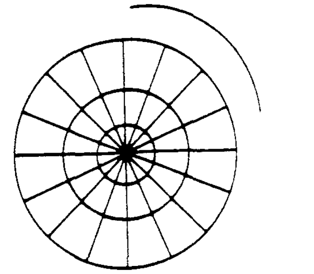

Figure 4: Suppose the robot uses θ, the direction in which the sensor is pointing, to distinguish between obstacles at the same distance, but at a different relative orientation. Then the world (around x) is divided into a finer partition than in fig. 3. Each equivalence class lies on a line through x, has unit length, and starts and ends at an integral distance from x.

kind of quotient map, and that the equivalence classes are induced by p_x. They are generated by the inverse images $\{ p_x^{-1}(b) \}$ for sensor values b in B.

We now elaborate on this point. We view the mapping p_x as follows. Let S^1 denote the unit circle, so we may say $\theta \in S^1$. We view points y in the world as living in a space we denote as M. Here, M is the plane $\Re^2$.

$$\begin{array}{rcccl} p_x & : & M \times S^1 & \rightarrow & B \\ & & (y,\theta) & \mapsto & b. \end{array} \qquad (2)$$

Let π_M be the natural projection of $S^1 \times M$ onto M, so $\pi_M(y,\theta) = y$. We now see the following

Definition 2.1 *At a fixed configuration x, the perceptual equivalence classes are classified by the sets of the form*

$$\pi_M p_x^{-1}(b) \qquad (3)$$

for a sensor value b in B.

Now, it is easy to extend our example to take θ, the direction of the sensor, into account. By remembering θ, the robot may distinguish points within one annulus. In this case the perceptual equivalence classes will be finer, and are conceptually sketched in fig. 4. Here partition of the world (around x) is finer than in fig. 3, and the perceptual equivalence classes are isomorphic to radial line-segments of unit length at orientation θ and integral distance from x.

Now, we note that the partition of M in fig. 4 is *finer* than the partition in fig. 3. Corresponding to this finer partition is a "finer" map[4] p_x^* which takes θ into account in the natural way. Intuitively, p_x^* should be be a quotient map $p_x \times \mathrm{id}_{S^1}$ that induces the partition into perceptual equivalence classes depicted in fig. 4, by using information about θ to disambiguate objects at the same distance but at a different relative orientation.

2.2 Lattice Structure

A fundamental question in characterizing the utility of a robot strategy is: what information is gained (or lost) by this strategy? More generally, one wishes to represent the knowledge of the robot's state before and after an action, and to measure, or compare the knowledge states in order to determine whether uncertainty as increased or decreased.

[4]Throughout, we use *map* to mean a mathematical function, not a representation for the robot's spatial environment.

The situation we encountered in sec. 2.1 is an example of a more general phenomenon. Let us introduce this phenomenon by contrasting our problem of robot strategy synthesis under large uncertainties with another, drastically easier problem: the gross motion planning problem with no uncertainty (perfect control, sensing, and models). In this problem, the state of the robot may be represented as a point in a configuration space. Thus moving from a start to a goal point may be viewed as finding an arc in free space connecting the two points. Since the robot is assumed to have perfect control and sensing, any such arc may be reliably executed once it is found. In particular, given a candidate arc, it may be tested. That is, motion along the arc may be simulated to see whether it is collision free. For example, an algebraic curve may be intersected with semi-algebraic sets defining the configuration space obstacles. In the presence of uncertainty, however, we cannot simply simulate a motion strategy to verify it. Instead, we need some technique for simulating *all* possible orbits, or evolutions of the robot system, under any possible choice of the uncertain parameters.[5] With sensing and control uncertainty, the state of the robot must be viewed as a subset of the configuration space. Motions, then, can be viewed as mappings between these subsets.

As an example, consider fig. 5, a sketch of the area in front of my office. Three corridors, labeled Left (L), Up (U), and Right (R) connect at an area near my office door (D). The top figure denotes the idealized model of the world, M. This neighborhood may be segmented into perceptually distinguishable equivalence classes as in sec. 2.1. Depending on the resolution and accuracy of the sensing, the equivalence classes might look like $\{ L, U, R, D \}$ (see fig. 5). So one motion might move the robot from L to D and another might move from D to U. Hence, together with the partition of M into recognizable classes, we obtain a "reachability graph" whose vertices are perceptual equivalence classes (what we call the *recognizable* sets), and whose edges are labeled with motions or strategies. The edges may be directed if the motion is not reversible. The connectivity of this graph might be (for example):

[5]That is, we need some method that can bound all possible behaviors of the system.

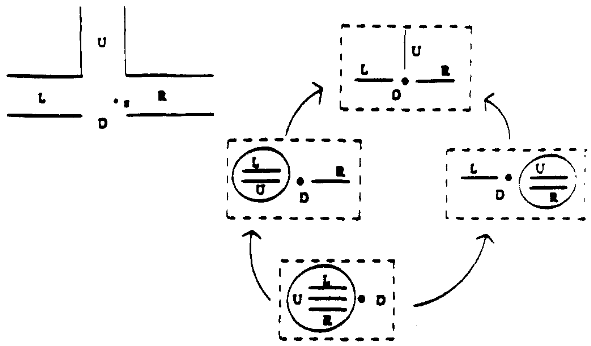

Figure 5: A sketch of the area in front of my office door, and part of the lattice of knowledge states. This lattice is (6).

$$\begin{array}{ccccc} & & U & & \\ & & \updownarrow & & \\ L & \longleftrightarrow & D & \longleftrightarrow & R. \end{array} \qquad (4)$$

2.2.1 Reachability Graphs of Recognizable Sets

The reachability relationship is the natural analogue of path-connectivity. We call a graph such as in (4) an *RR-graph*, which stands for a *reachability graph of the recognizable sets*. Hence it is clear that motion of the robot induces a motion in the RR-graph. Viewing the motion in the RR-graph as a "mapping", we call it a *reachability map*. Each reachability map corresponds to a connected sequence of edges in the RR-graph. An example of a reachability map is: a strategy (or motion) v takes the robot from class D to class L. We denote this by $D \xrightarrow{v} L$.

However, not all motions will be as "well-behaved" as a motion such as (for example) $D \xrightarrow{v} L$. Suppose that due to uncertainty in control and sensing, we cannot guarantee that we will move to L, but that it is possible that we remain in D, or even move to U "by mistake." Suppose further that after executing the motion (call it v), we can tell (by sensing) whether or not we are still in D, but that L and U will be indistinguishable.

In this case we have the transition $D \xrightarrow{v} \{D, LU\}$, where LU denotes the union of sets L and U. In essence, this means that after executing the strategy v, we know the robot is in D, or it is in L or U (but we don't know *which* of L or U).

Now, clearly the motion v descends to a state of knowledge that is less certain. We have the following idea for how to measure this change in information of the knowledge state. We imagine another partition of M into the perceptual equivalence classes $\{LU, R, D\}$. This means that L and U are perceptually indistinguishable in the new partition. It should be clear that such a partition, in which there is *less information* about the world and the perceptual equivalence classes are larger could arise naturally due to less accurate sensing, and its existence is independent of strategies and motions like v.

First, we remark that each new partition such as $\{LU, R, D\}$ comes equipped with its own RR-graph. Hence, we can think of the transition $D \xrightarrow{v} \{D, LU\}$ as being the composition of two mappings: a *forgetful* map and a reachability map:

1. The first map descends *in the lattice* shown in (6). It is a map from one partition to a less fine partition. This map is specified by:

$$\begin{array}{ccc} \{L, U, R, D\} & \to & \{LU, R, D\} \\ L & \mapsto & LU \\ U & \mapsto & LU \\ R & \mapsto & R \\ D & \mapsto & D. \end{array} \qquad (5)$$

   We call the map (5) the *forgetful map*, since it loses information (and "forgets" whether we were in L or U). It is clear that for any two partitions $\mathcal{U}$ and $\mathcal{V}$, if $\mathcal{U}$ is finer than $\mathcal{V}$, then there is a well-defined forgetful map from $\mathcal{U}$ to $\mathcal{V}$.

2. The second map is simply a reachability map in the less fine partition $\{LU, R, D\}$. That is, it is a self-map of $\{LU, R, D\}$. In this case it takes (non-deterministically) D to D or D to LU.

Hence, in the same way that sensing operations that provide more information will "ascend" to a finer partition, it is clear that motion strategies can "descend" to a less fine partition (with less information). Of course, motion strategies can also gain information; for example, they can use compliant motion or wall-following to reduce uncertainty [Don].

2.2.2 The Lattice of Perceptual Equivalence Classes

One might think that strategies that lose information are to be avoided; however, sometimes they are necessary since attaining some goal may require descending to a more ambiguous state that contains the goal as one possibility (e.g. as a subset). To investigate this situation, we must better understand the relationship between partitions.

Consider the new partition $\{LU, R, D\}$. Clearly, the new partition is "less fine" than $\{L, U, R, D\}$ in the same sense that the partition in fig. 3 is less fine than fig. 4. We denote this relationship of "finer partition" by placing $\{LU, R, D\}$ lower than $\{L, U, R, D\}$ in the lattice shown in fig. 5. There are other partitions we could imagine, such as $\{L, UR, D\}$ that are incomparable in fineness to $\{LU, R, D\}$ but are both finer than (say) $\{LUR, D\}$ and less fine than $\{L, U, R, D\}$:

$$\begin{array}{ccccc} & & M & & \\ & & \uparrow & & \\ & & \vdots & & \\ & & \uparrow & & \\ & \nearrow & \{L, U, R, D\} & \nwarrow & \\ \{LU, R, D\} & & & & \{L, UR, D\} \\ & \nwarrow & \{LUR, D\} & \nearrow & \\ & & \uparrow & & \\ & & \{LURD\}. & & \end{array} \qquad (6)$$

We think of a partition $\mathcal{V}$ as "including" in a finer one $\mathcal{U}$ and since we think of the arrow relationships in (6) as

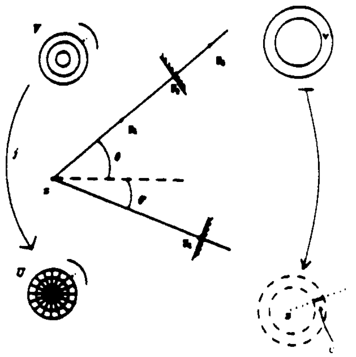

Figure 6: Illustration of the inclusion map. $\mathcal{V}$ is the partition obtained by a sensing strategy that merely interrogates the distance of objects. $\jmath$ models a "better" sensing strategy that interrogates the relative orientation θ as well. The image of equivalence class $V \in \mathcal{V}$ will be a set $U \in \mathcal{U}$ of the form $U = V \cap r(x, \theta)$, where $r(x, \theta)$ is the ray from x in direction θ.

being like inclusion maps, we will call them *inclusions*.[6]

2.3 Inclusion Maps Model Sensing Strategies

Now, consider sec. 2.1, and figs. 3 and 4 in particular. Suppose that the partition shown in fig. 4 is denoted $\mathcal{U}$, and fig. 3 is denoted $\mathcal{V}$. Then by the observation above, there is an inclusion $\jmath : \mathcal{V} \hookrightarrow \mathcal{U}$. What is the nature of this inclusion $\jmath$? The map $\jmath$ takes a coarse perceptual equivalence class V (in this case, an annulus) to a "fine" equivalence class U (a line segment, as shown in fig. 6).

Now, which line segment U is the image of V? Clearly, we must have $U \subset V$. In physical terms, the answer depends on θ. Here is the key point: passing from the coarser partition to the finer one corresponds to gaining information. The information is gained by using another sensor, or a better sensor, or by interpreting the information in the sensor better. We name this process, and call it *the application of sensing operation* $\jmath$. The image U of V under $\jmath$ is the result of applying operation $\jmath$ to the known state V. That is, the image will depend on the "better" sensing strategy's outcome. The manner in which V is refined into some (sub)set U depends on the exact state of the world and the result of the sensor operation.

However, even though we cannot always predict, ahead of time, what $\jmath(V)$ will be, we may be able to characterize this set in some way. For example, suppose V is the unit annulus with integral inner radius d. U will depend on θ (which $\jmath$ "senses"). V and θ completely characterize U, and yet θ is unknown before applying $\jmath$. Let $r(x, \theta)$ be the ray from x in direction θ. Then we know that

$$\jmath(V) = U = V \cap r(x, \theta).$$

Hence, we have a complete classification of the kind of set that can result from applying operation $\jmath$. Facts deduced from this classification–e.g. size, shape, dimensions, bounds–can be used in deciding whether performing the operation and increasing the information about the state is useful.

We note that these inclusions naturally model the application of a sensing strategy that gains information. An inclusion maps from a partition to a finer one. The image of a particular knowledge state under the inclusion depends on the result of the operation and cannot necessarily be predicted in advance of trying the sensing strategy. (In this sense they are "dual" to motion strategies under uncertainty). However, using knowledge of the sensors, often the space of recognizable sets (in this case, $\mathcal{U}$) can be characterized, and this characterization can be used in planning and choosing actions.

[6] In general, given two distinct partitions $\mathcal{U}$ and $\mathcal{V}$ of M into perceptual equivalence classes $\mathcal{U} = \{U_\alpha\}$ and $\mathcal{V} = \{V_\alpha\}$, either $\mathcal{U}$ or $\mathcal{V}$ will be finer, or they will be incomparable.

2.4 The Local Quotient is a Bundle

A key phenomenon of sensing is that the robot's perception of the environment changes when the robot has moved. For example, suppose x and z are configurations of the robot, and z is "far" from x. Then from x, objects in the world "near" x appear finely distinguished, while objects in the world near z all blur together. When the world is viewed from z, the situation is reversed: objects near z appear finely interdifferentiated, while the environment near x is identified to a single perceptual equivalence class. We consider varying a configuration ϕ from x to z along a path connecting the two points. This defines a family of local sensing maps $\{p_\phi\}$. Each map locally defines a partition of the world into perceptual equivalence classes (and an RR-graph); this "sequence" of RR-graphs "converges" to the RR-graph "at z" as $\phi \to z$ and $p_\phi \to p_z$. We wish to paste together these local maps $\{p_\phi\}$ to get a "global" sensing map, which will be a bundle. The bundle, when suitably restricted, will yield the local quotient maps like p_x.

For a fixed configuration ϕ, we have a sensing map such as p_ϕ described in some detail above (for example, p_ϕ is like p_x or p_x^* in (2)). We "globalize" p_ϕ as follows:

Definition 2.2 *The* world sensing bundle p *is given by*

$$p \; : \; M \times C \to B, \quad (y, \phi) \mapsto p_\phi(y). \tag{7}$$

Now, let π_C be the natural projection of $M \times C$ onto C, so $\pi_C(y, \phi) = \phi$. Hence both p and π_C are fibrations. We construct the diagram

$$C \overset{\pi_C}{\swarrow} M \times C \overset{p}{\searrow} B. \tag{8}$$

It is not hard to see the following property, which yields the desired local quotient structure we promised:

Proposition 2.3 *The map p is locally a quotient map; i.e., it is a quotient on each fibre $\pi_C^{-1}(\phi)$ of π_C.*

When we have a robot fixed at one configuration, the definition of perceptual equivalence classes we adopted in 2.1 is sensible. Now, we wish somehow to generalize the kind of definition in 2.1 to the case of the bundle p. We can do this by passing to the configuration space C of the robot, and define an equivalence relation on this space that defines what it means for two places to "look the same." This essentially means that to construct global representations of perceptual equivalence classes, we pass to "robo-centric" coordinates (configuration space).

To make sense of this definition, it helps to bear in mind the following:

1. The configuration space C includes the control parameters of the sensors (e.g. aim of the rangefinder), and knowledge of these parameter values is encoded in the sensor space B.

2. Sensor noise is modelled by passing to set-valued functions. That is, we consider $p_\phi(y)$ to be a subset of B, representing a bound on all possible sensor values. In general, this can destroy the transitivity of, for example, def. 2.4 (below), but the reflexivity and symmetry are retained. Hence a definition like 2.4 becomes analogous to the "adjacency" relation, and its transitive closure analogous to "connectedness". This issue is subtle. Even though noise *can* destroy transitivity, we still use the term *perceptual equivalence class* to illuminate connections with previous work [MC, BW, S, CMM], which uses the term *sensory equivalence*. While these ideas cover the basic approach to dealing with noise and uncertainty, there is much more to be said; see [DJ].

Definition 2.4 *Fix a sensor value $b \in B$. We call a subset S of configuration space C a* configuration space perceptual equivalence[7] class *(or simply* C-Perceptual Equivalence Class*) when for each configuration ϕ in S, there exists some point y_ϕ in the world M such that $p_\phi(y_\phi) = b$.*

If we imagine the robot living in configuration space C, the places relevant to the robot will be places in that configuration space. This definition tells us what it means for two places to appear the same; it defines what it means to be a perceptual equivalence class in configuration space.

2.5 C-Space Perceptual Equiv. Classes

We wish to formalize the construction of the C-perceptual equivalence classes, to obtain a way to generate and classify them in general.

Lemma 2.5 *The maximal C-Perceptual Equivalence Classes under definition 2.4 are the cells in the arrangement generated by the sets*

$$\{ \pi_C p^{-1}(b) \}$$

for $b \in B$.

We will later discuss how this classification lemma leads to an algorithm for computing the recognizable sets in configuration space.

2.6 History is Important

We use the term *sensor history* to denote a mapping from time to sensor values, or any "subset" of this function, that the robot remembers. Similarly, we use the term *control history* for the mapping from time to commanded controls. Both can be immensely useful in robot strategies [E]. Consider an robot that must follow walls, in a typical office building. Suppose it can distinguish corners from straight hallways, but cannot tell, say, a north from a west wall. If the robot starts near wall L (e.g. as in fig. 5) and makes 1 or 2 turns (following walls), it can know that it is has not reëncountered L, even though the nearest wall now (say, U) is *locally* perceptually indistinguishable from L. This reasoning is accomplished using sensing and control history, represented by a reachability map (2.2.1). Effectively, history permits us to ascend to a finer partition, in which U and L are distinguishable.

As another example, consider fig. 6, illustrating a partition $\mathcal{V}$ and a finer partition $\mathcal{U}$ related by an inclusion map $\jmath$, which models the *a priori* availability of better sensing information (namely, orientation). Assume that sensing θ is impossible. This places us in the coarse partition $\mathcal{V}$. Now suppose control history is available. For example, coarse control history $v^*(t)$ might approximately "remember" the commanded motion direction $v(t) \pm 45°$. Our inability to sense θ makes the inclusion map $\jmath$ unavailable directly; hence, the finer partition $\mathcal{U}$ is apparently unimaginable. However, our *a priori* knowledge of $\mathcal{V}$ plus control history $v^*(t)$ permits us to "lift" to the partition $\mathcal{U}$ "manually"—that is to construct an inclusion map like $\jmath$ from the evidence $(\mathcal{V}, v^*(t))$. See [DJ] for details of the construction. Hence, we can pass to a finer partition $\mathcal{U}$ and reason about sensing and action there. This example illustrates a quite general phenomenon. There is much more to be said about the "lifting" ability history admits; for a mathematical formalization see [DJ].

[7] But see point 2, above.

3 Computing C-Perceptual Equivalence Classes

One may investigate our theory by picking a particular world model, and then asking what are the C-perceptual equivalence classes given some representation of the world sensing bundle p). This does not mean that our approach is limited to cases where a world model is available (see sec. 4 below). Instead, we merely suggest that the construction of the C-perceptual equivalence classes should be studied in the case where a model is known, and that these "maps" provide invaluable data on the structure of the recognizable sets. Finally, the ability to construct the C-perceptual equivalence classes and RR-graphs from a world model will indeed be useful in those structured situations where a model is available.

Our basic result is: when (1) the sensor space B is finite, (2) the geometry of the world can be encoded as a solid model, and (3) we can "write down" the world sensing bundle p so that an algorithm can manipulate it, then the C-perceptual equivalence classes are efficiently computable. Note that the finite sensor space B could be obtained by breaking up a continuous sensor space into a finite number of equivalence classes. A comment on (3): we will assume that we can write down the geometric world model and the map p using the language of semi-algebraic sets. In [DJ] we argue that this is reasonable, and not very restrictive. We can show:

Lemma 3.1 [DJ] *Suppose that B is finite and contains n values, and that M and C are algebraic and each has dimension d. Then the C-perceptual equivalence classes can be computed in time polynomial in n and exponential in d.*

As a corollary, we see that "sensor fusion" (adding new sensors and sensor values) is hence polynomial-time, since adding sensor values only increases n. Finally, we note that if our modelling is adequate to predict (simulate) the outcome of motion strategies (under uncertainty), then, having constructed the C-Perceptual equivalence classes using the algorithm above, then we can construct an RR-graph whose edges denote *guaranteed* or *strong* (instead of *possible* or *weak*) reachability. Algorithms for this problem are sometimes called *forward projection* (under uncertainty), and have been developed, for example, in [Erd,Buc,Don,Can]. The assumption that we can compute accurate forward projections is rather strong (in the mobot domain) but may be possible in semi-structured environments, such as mechanical parts assembly.

4 Incremental Computation

Suppose the robot has no model of the world, but it wishes to build up a reachability graph of the recognizable places. Incremental computation of RR-Graphs and C-Perceptual Equivalence Classes proceeds in the obvious way: the C-perceptual equivalence classes are completely classified by the sets $\{ \pi_C p^{-1}(b) \}$, for $b \in B$ (lemma 2.5). Hence they are completely determined by the sensor values b. Let us imagine that B is discrete, or at least that it is the quotient of the true "continuous" sensing space—e.g. that we have broken the sensing space into regions that are close enough to "look the same." Suppose the robot starts out sensing some value b_0. It moves, performing action v. The sensors now read b_1. Hence the robot can build the graph $b_0 \xrightarrow{v} b_1$, which of course is shorthand for $\pi_C p^{-1}(b_0) \xrightarrow{v} \pi_C p^{-1}(b_1)$. The latter is a (weak) reachability edge in the RR-graph. At any rate, subsequent moves add more edges to the RR-graph, which is the robot's representation of the recognizable places and the reachability relationships between them. We call the resulting structure a **B-encoded RR-graph.** "Planning" then reduces to a search of the RR-graph—without reconstruction (definition 4.1 below) or a "built-in" world model, this is the only type of planning possible. See, for example, [Mat]. Note that in general, we may consider the RR-graph to be only partially known.

There is much more to be said about our framework in the absence of world models. For example: Given a bundle $p : E \to B$ we call a map $s : B \to E$ a *section* of p if $p \circ s$ is the identity on B. A section is called "local" if it is only defined on a neighborhood U of B. Suppose we have a partial or incomplete model of the geometry of the world, and a partial or incomplete characterization of the sensor map p. In our view, a "small number" of remembered sensor values and control values defines a kind of local section of the world sensing bundle $p : C \times M \to B$. That is, this history and a partial model permits us to hypothesize a correspondence between current sensor values (and also "nearby" sensor values) and objects in the world. Hence history in general generates a "family" of these sections $\{ s_t \}$, parameterized by time. This allows us to define a computational problem that is related to what others have called "map-making" in mobile robotics:

Definition 4.1 *The* reconstruction problem *is as follows. Given a partial world model of geometric and physical obstacles in M, that "projects" to sensor values by a world sensing bundle $p : C \times M \to B$, construct an approximation to the inverse map p^{-1} from a family of sections $\{ s_t \}$.*

We view the value of *reconstruction* as a key topic of debate in analyzing "reactive" systems. In a way, reconstruction is a hybrid of "classical" planning (planning with world models) and reactive planning (planning with minimal models). Need a robot need reconstruct in order to be successful? Can we characterize which tasks and domains require reconstruction, and to what degree reconstruction can be ignored? These issues (since they concern the importance and utility of the map p^{-1}) are "dual" to questions of the importance of world models. That is, one may independently ask (for a task or domain): *(1) Are world models required? How much of a model is necessary?* and *(2) Is reconstruction required? How good an approximation of p^{-1} is necessary?*. Finally, we conjecture the following: *(3) Reactive systems do not reconstruct, or reconstruct minimally. A measure of "reactivity" may be how little the map p^{-1} is approximated, constructed, or used.*

5 Conclusions

We considered how a robot may interpret its sensors and direct its actions so as to gain more information about the world, and to accomplish manipulation tasks. We observed that sensing forces us to view the world through a kind of projection, and that the information loss in this projection divides the world into perceptual equivalence classes.

We identify global landmarks, which are the recognizable sets in configuration space, the C-perceptual equivalence classes. We can compute them, given a world model, and we developed efficient algorithms for doing so. In the absence of a model, these "landmarks" can be acquired incrementally, through exploration. A graph of these explorations, the RR-graph, encodes the reachability relationships between the recognizable sets. Again, it may be computed from a model, but in the absence of a model we can "encode" it using the sensor values and history alone. For a sensor space B, we call this a "B-encoding" of an RR-graph. This representation seems ideal for mobots with minimal models.

RR-graphs and C-perceptual equivalence classes are simple concepts; these recognizable sets form the basic building blocks for our theory. These landmarks, or recognizable sets—that is, the places in the world that the robot can recognize and distinguish between—have structure we can exploit in developing sensor-based planning algorithms. In particular, the interaction of the lattice and history is subtle and can be used to plan or program sophisticated strategies for achieving goals and accomplishing tasks.

Our goal is to develop a framework for reasoning and planning in the presence of uncertainty, and, in particular, with minimal models. We believe our ideas can help a robot to determine what information is required to solve a task, and what actions help acquire that information to solve it. It is also our hope that this framework is useful in a broad class of domains, and can be thought of as a programming and planning paradigm for reasoning about about uncertain and unknown environments. We are currently implementing our ideas on a mobile autonomous platform, and developing the capability to construct, compute, and recognize perceptual equivalence classes, to plan and reason using them, and in particular, to exploit the lattice, bundle, RR-graph, and history structures and algorithms described here. For a description of our architecture and hardware, see [OdD, BSD].

6 Previous Work

Many of the issues in this paper have been encountered or have precursors in earlier work. Due to space limitations, we can mention only a small number of these.

Strategies for coping with uncertainty, and reasoning about reachable and recognizable sets has been a theme in robotics research for many years. See, for example: [LMT, E89, Don, Brost, Bri, DW]. Reactive systems, behavior-based robots, and the subsumption architecture are discussed in [Brooks 1985], [Mat], [MC]. The last describes the use of reinforcement learning in the mobile robot domain. [Dufay and Latombe] implemented a system which addresses learning in the domain of robot motion planning with uncertainty. In work on learning manipulation strategies, [CMM] identifies perceptual equivalence classes as consequences of imperfect sensing, which may be viewed as a projection [C]. [WB] and [S] discuss perceptual and model aliasing. Our lattice is reminiscent of Mitchell's *version space* [Mit]. Finally, we believe the ideas in [R] to be of considerable relevance.

7 Acknowledgements

Many key ideas in this paper arose in discussions with Tomás Lozano-Pérez, Mike Erdmann, and Dan Huttenlocher, and we are very grateful to them for discussions and suggestions, and for their generosity with their time and ideas. We would like to thank Peter Kahn and Allen Back for discussions and advice. Thanks to Maja Mataric for discussing her work and robotics in general with us. Thanks also to Devika Subramanian, David Chapman, and Phil Agre for insight and comments on this work.

8 References

[BSD] **Blondell, B., Shollenberger, S. and Donald, B.** *Design of a microprocessor driven generic controller for a mobile robot base,* Cornell CS Dept., Ithaca, NY CU-CS-TR-90-1140 (1990).

[Bri] **Briggs, Amy.** (1989) *An Efficient Algorithm for One-Step Compliant Motion Planning with Uncertainty,* 5^{th} ACM Symp. Comp. Geom., Saarbrucken, BRD

Brooks, R. A. *Symbolic Error Analysis and Robot Planning,* IJRR Vol 1, no. 4, Dec., 29–68 (1982).

Brooks, R. A *A Robust Layered Control System for a Mobile Robot,* IEEE JRA **RA-2** (1): 14–23. (1985).

Brost, R., "Automatic Grasp Planning in the Presence of Uncertainty", *IEEE ICRA*, San Francisco, 1986.

[Buc] **Buckley, S. J.** *Planning and Teaching Compliant Motion Strategies,* Ph.D. Thesis. MIT, Dept EECS, 1987. Also MIT-AI-TR-936 (1987).

Caine, M., "Chamferless Assembly of Rectangular Parts in Two and Three Dimensions", S.M. thesis, MIT Dept. of Mech. Eng. 1985.

[Can] **Canny, J.F.** *On The Computability of Fine-Motion Plans,* Proc. IEEE ICRA Scottsdale, AZ (1989).

Chapman, D. *Planning for Conjunctive Goals,* MIT AI-TR 802 (1985).

[C] **Christiansen, A.** *Manipulation Planning from Empirical Backprojections,* CMU Dept. C.S. (1991).

[CMM] **Christiansen, A., Mason, M., and Mitchell, T.** *Learning Reliable Manipulation Strategies without Initial Physical Models,* IEEE ICRA (1990).

[D] **Donald, B. R.** *A Geometric Approach to Error Detection and Recovery for Robot Motion Planning with Uncertainty,* Artificial Intelligence, **37** (1-3), Dec. pp. 223-271. (1988).

[Don90] **Donald, B. R.**, "Planning Multi-Step Error Detection and Recovery Strategies", *IJRR*, **9** (1), Feb. 1990. pp. 3-60.

[Don] **Donald, B. R.** *Error Detection and Recovery in Robotics,* Springer-Verlag (New York): Lecture Notes in CS, vol. 336 (1989).

[DJ] **Donald, B. and Jennings, J.** *Sensor Interpretation and Task-Directed Planning Using Perceptual Equivalence Classes,* Cornell CS Dept, Ithaca, NY CU-CS-TR (1991).

Dufay, B., and J. Latombe, "An Approach to Automatic Robot Programming Based on Inductive Learning", in Brady, M., and R. Paul, *Robotics Research: ISRR-1,* MIT Press, 1984.

[DW] **Durrant-Whyte, H.** *Concerning Uncertain Geometry in Robotics,* Artificial Intelligence, **37** (1-3), Dec. (1986).

[E] **Erdmann, M.** *Using Backprojections for Fine Motion Planning with Uncertainty,* IJRR Vol. 5 no. 1 (1986).

[E89] **Erdmann, M.** *Probabilistic Strategies for Robot Tasks,* MIT AI Lab PhD Thesis, Dept. EECS (1989).

[FHS] **Friedman, J., J. Hershberger and J. Snoeyink** *Compliant Motion in a Simple Polygon,* 5^{th} ACM Symp. Comp. Geom., Saarbrucken, BRD. (1989).

[GM89] **Goldberg, K., and Mason, M.** *Bayesian Grasping,* IEEE ICRA, 1990 (1989).

[LLS] **Latombe, J.C., A. Lazanas, and S. Shekhar** *Motion Planning with Uncertainty: Practical Computation of Non-Maximal Preimages,* IEEE/RSJ Int'l. Workshop on Intelligent Robots and Systems, Sept. 4-6 (1989).

[LoP] **Lozano–Pérez, T.** *Spatial Planning: A Configuration Space Approach,* IEEE Trans. on Computers (C–32):108–120 (1983a).

[LMT] **Lozano-Pérez, T., Mason, M. T., and Taylor, R. H.** *Automatic Synthesis of Fine-Motion Strategies for Robots,* IJRR, Vol 3, no. 1 (1984).

[MC] **Mahadevan, S. and Connell, J.** *Automatic Programming of Behavior-based Robots using Reinforcement Learning.,* IBM T.J.Watson Res. Rept. (1990).

Mason, M. T. *Automatic Planning of Fine Motions: Correctness and Completeness,* 1984 IEEE ICRA, Atlanta Ga. (1984).

[Mat] **Mataric, M.** *A model for distributed mobile robot environment learning and navigation,* S.M. Thesis, MIT AI Lab, Dept. EECS (1990).

[Mit] **Mitchell, T. M.** *An Approach to Concept Learning,* Stanford Univ., Palo Alto, CA TR STAN-CS-78-711 (1978).

[OdD] **O'Donnell, M. and Donald, B.** *On Large Scale Planar Manipulation,* Cornell CS Dept., Ithaca, NY CU-CS-TR 90-1127 (1990).

[R] **Rosenschein, S.J.** *Synthesizing Information-Tracking Automata from Environment Descriptions,* Teleos Research TR No. 2 (1989).

[S] **Shafer, S.** *Why Can't We Model the Physical World?,* for the CMU CS Dept. 25^{th} Anniversary (1990).

Simunovic, S. N., "An Information Approach to Parts Mating", Ph.D. thesis, Dept. EE, MIT, 1979.

Taylor, R. H. *The Synthesis of Manipulator Control Programs from Task-level Specifications,* Stanford A.I. Lab., AIM-282, July (1976).

[WB] **Whitehead, S. and Ballard, D.** *Active Perception and Reinforcement Learning,* 7^{th} Int'l. Conf. on Machine Learning (1990).