CHAPTER 2

PLANNING A PLAN

As the saying goes, “You have to plan your work, and work your plan.” A planning algorithm would likely heed this advice. Planning covers a variety of topics, from motion planning, which is used in this dissertation, to investment strategy planning. In either case, an intelligent agent makes decisions on what the best action is to drive a system from some initial state to some goal state. The amount of freedom the agent has depends on the development of the entire system. If a person is hungry, the initial and goal states might be how to get from his current location to some restaurant. For investment planning, the initial state might be a current net worth of one dollar with a goal of $1 billion. Traditionally, in the case of robotics, planning contains two parts: motion planning and trajectory planning [36]. The motion planner converts high-level task specifications into low-level movement descriptions. Next, the trajectory planner determines how to carry out these movements while accounting for physical constraints of the robot. As shown later, the planning algorithm used here actually combines the two steps. The high-level task specification is given in the form of a nominal desired path, and a motion plan is generated which adheres to the kinematic constraints on the system.

The term motion plan is a misnomer because it is not so much of a motion plan, which implies specifications beyond a path, as it is a path plan. Then, trajectory planning implies how to move along the path given the system constraints. The resultant of the combination of *path* planning and *trajectory* planning is a *motion* plan. The issue is perhaps exacerbated in SUPCI since the desired path is referred to as a nominal *trajectory*. This is most likely due to the fact that the nominal trajectory is time-dependent, thus presenting both a desired path and a desired trajectory to the planner which is referred to here as a *motion* planner. In the case of SUPCI, the algorithm determines in what order and for how long to apply system inputs to achieve the desired motion.

When discussing locomotion or steering a vehicle, the idea of motion planning seems like a natural concept. However, how does one plan a motion for the task of manipulating an object? Several factors must be considered. First, the object must be held tightly enough so it is not dropped. Second, while the object must be held firmly, it must not be held so firmly that it is broken or crushed. Third, the motion of the fingers is correlated with the desired motion. For example, to rotate an object, the fingers must move in a plane perpendicular to the axis of rotation. Fourth, it must be determined when the joint limits of the manipulator have been reached so the fingers can be repositioned to continue the task, for example, screwing in a light bulb. Finally, there must be a way to indicate that the task has been completed. But these are high-level criteria. Within any one task, achieving other performance criteria may also be important.

Obviously, the main goal is to drive a system from an initial state to a goal state with high accuracy. However, several other factors must be considered. These may include obstacle avoidance, energy consumption, real-time error corrections, or time required to achieve a goal. Certain methods become intractable if it is necessary to prepare a contingency for every issue that may arise. However, since humans are able to cope and even excel in an ever-changing world, an aside on intelligence is in order.

How humans go about accomplishing tasks is a complicated study. The breadth is such that the researchers are social, cognitive, and neural psychologists, linguists, activity theorists, computer scientists, and engineers. Only a brief review on the quest for artificial intelligence (AI) is presented in an attempt to relate it back to the idea of biological motivation for robotic manipulation. When all is said and done, the bases for an intelligent design, in the absence of an intelligent agent, are sensory feedback and natural language processing. These provide motivation for the use of tactile sensors and of fuzzy logic in this work.

2.1 The Search for Artificial Intelligence

The traditional AI view held that an executor was in the world to control it by carrying out instructions of a plan [1]. However, recognizing that humans operate in a vast array of infinite possibilities, Suchman [65] suggested situated actions. These are actions that make sense only when taken in context. Thus, the executor evolved into an agent. An agent interacts with its environment and its planner to improvise when necessary.

To separate the two approaches, Agre and Chapman [1] present two views on planning. The first is plan-as-program where the plan is simply a set of instructions to be carried out sequentially by an executor. The first autonomous vehicle, Shakey, worked in this way [7]. It viewed the world as static. Therefore, it was oblivious to changes in its environment while its path was being planned. The second view is that of plan-as-communication where the plan is more of a suggestion. It is up to the agent to modify the plan as necessary to fit current circumstances or to scrap the current plan and move to a new one. For example, a student has a plan to walk to school this morning. As a communication, it is a very high-level task with no constraints except that the only choice of transportation is to walk, and that the destination is school. What happens if it is raining? Does the student modify his plan and bring along an umbrella? Does he scrap the plan and drive to school instead? Could he have executed these modifications by a plan-as-program approach? This would require a set of rules along the lines of "if it is raining, then bring an umbrella." How many branches of rules would a planning program need to account for all the uncertainties possibly encountered during the trip?

Advances in imaging technology have led to a new interpretation of cognition [2]. It starts with the structure of the brain. Researchers now see the brain as blocks of interconnected modules for processing various information [2, 4]. Although it is an oversimplification, this may not be unlike the traditional control structure represented in Figure 1.1. In addition, through functional magnetic resonance imaging (fMRI), researchers are now able to observe the brain in action. Much of the research goes into seeing what parts of the brain are stimulated during logic games-playing such as Towers of Hanoi (See, for example, [3, 17, 18, 50]). Huettel *et al.* [27] have shown that long and short term memory processing occur in different parts of the brain, and that this complements the brain's attempts to reduce uncertainty in decision making based on experience.

The confluence of advances in imaging and other fields led to the *computational theory of mind* [2]. This theory provides a framework for scientific analysis, a way to cast abstract concepts such as perception, motivation, and emotion in a scientific light. In doing so, mental states become amenable to scientific inquiry. The theory posits that the brain is a system of organs to perform computations, and that performing these computations effects intelligence [44]. All one needs to do then to create an intelligent system is to build a system that looks, structurally, like the brain.

In the absence of an intelligent agent, the focus is switched to two concepts the referenced researchers agree on as requirements for building intelligent systems. One is interaction with and perception of the environment. The second is semantics of action [24]. The first is accomplished through sensory input and processing of that information. Based on results of new or updated information, it is assumed the agent will make good decisions because it has a better estimate of the consequences [27]. The latter is accomplished through natural language processing. By understanding the plan's intent given the current situation, the agent is empowered to overlook low-level commands, and to modify actions in an attempt to still achieve the goal. It is as if the student decided to take the bus to school rather than walk in the rain.

The group of opinions above provides motivation for two approaches followed in this work, namely fuzzy logic, a method for computing with words, and haptic feedback, a method for a robot to interact with its environment. With this motivation complete, the attention turns back to steering robotic systems.

2.2 Motion Planning

Motion planning must account for physical constraints of the system. For manipulation tasks, constraints arise from contact between a manipulator and an object. During contact, this limits certain directions a manipulator can move. In addition, intermittent contact gives rise to nonsmooth equations of motion. Similarly, objects with corners or edges give rise to nonsmooth equations of motion.

2.2.1 Holonomic Motion Planning

Early work in motion planning was done with continuous, holonomic systems [22, 45, 70]. However, this represents only a small class of systems with engineering utility. Many interesting systems also include nonholonomic constraints. The task of parallel parking a conventional road vehicle is one such system. Without accounting for drive constraints, a motion planning scheme may generate a path contrary to the physical ability of the system, for example, requiring a sideways motion of the vehicle, which constitutes sliding [48]. Sliding constraints are known to be holonomic in nature. Since sliding control introduces dynamical relationships between objects, it is not considered in this work except to elucidate the interpretation of contact kinematics developed in Chapter 3. Finally, many practical systems are discontinuous in nature. One example is a task requiring intermittent contact such as walking. It is necessary to develop schemes to account for such diversity.

2.2.2 Nonholonomic Motion Planning

The nonholonomic motion planning method from [34] is used in this work. This section presents an overview of the approach. It is described in mathematical detail in Chapter 3. The motivating example throughout will be parallel parking a vehicle, and Chapter 4 presents a complete solution to a parallel parking problem.

Given a control system described by a set of generally nonlinear ordinary differential equations, it is possible to generate new, desirable directions along which the system can move by applying available inputs in a specified manner. The algorithm's name derives from the fact that each input, when it is applied, is held constant for a specific amount of time, turned off, and then another input is applied. The composition of inputs may yield motion in a previously unrealized direction. Such systems are called *underactuated* since direct control inputs are not available for any direction in which to steer the system. Underactuation arises in the vehicle problem due to a physical constraint, namely that the wheels are restricted from sliding perpendicular to their orientation. For the parallel parking problem shown in Figure 2.1, ordered combinations of forward and reverse along with rotating the wheel allow the vehicle to be positioned between the other two vehicles. To an outside observer witnessing only the initial and final locations of the vehicle, it may appear as if the system has an additional, albeit fictitious input, allowing the vehicle to move sideways. It is interesting to note that humans typically parallel park without using discrete control inputs. Rather, the accelerator is controlled simultaneously with the steering wheel.

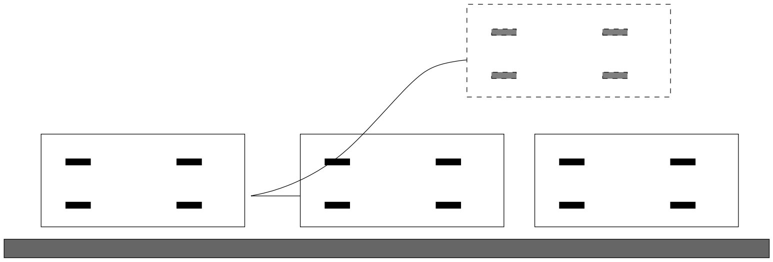

Figure 2.1. Path for Parallel Parking a Vehicle

For intermittent contact, the equations of motion are discontinuous. The method above is now extended to such systems.

2.2.3 Stratified Motion Planning

Discontinuous systems are generally characterized by the presence of intermittent physical constraints. However, for many systems, this may be their most salient feature. For example, to manipulate an object, fingers may have intermittent contact with the object. Likewise, ambulation is characterized by feet having intermittent contact with the ground during a gait cycle. Systems with such constraints pose difficulties from the control-theoretic viewpoint because they have discontinuous equations of motion for which typical control algorithms are not applicable. While position control of robot manipulators can be achieved using several techniques, such as computed torque or linear control laws designed for linear versions of the system, discontinuous systems have no such controller designs.

Goodwine [19] defines *stratified* as a configuration manifold containing submanifolds upon which the system is subject to (additional) constraints. The mathematical concept of a manifold will be presented in Section 3.2. Obviously, systems characterized by intermittent contact or engagement belong to this class. Stratified motion planning is fundamentally based upon the nonlinear geometric properties of such systems and the extension of geometric nonlinear control techniques from [34].

The differential geometric basis for the control theory, however, requires nearly exact *a priori* knowledge of the system and knowledge of the environment in which it operates. For example, if a legged robot walks on a smooth floor, previous work provides means for determining controllability and for providing motion planning algorithms [19]. However, if the surface geometry is unknown, or if it contains jagged terrain, the previous work is inapplicable because the stratified structure cannot be explicitly determined. In addition, unmodeled dynamics or physical degradation can affect the accuracy of the model upon which the stratified structure is based, consequently hindering performance.

2.3 Planning for Nonsmooth Object Manipulation

Work done by Wei [76] has extended [19] to include nonsmooth object manipulation by a set of coordinating robots. First, Wei extends the stratified approach to nonsmooth systems by identifying multiple, lowest-dimensional submanifolds associated with a nonsmooth object. Second, Wei constructs an approach for closed loop experiments using a vision-based concept known as Camera Space Manipulation (CSM) to provide visual feedback to each of the manipulators on the orientation and location of both the object and of the end-effectors in a common frame of reference. CSM requires visual cues to be placed on the manipulator and object. Based on the visual information received, the method forms a map between the joint configuration of the robot and the appearance of cues on the end-effector and/or object being manipulated [12]. Processing requirements to detect the cues, however, must remain tractable or the method would not be pragmatic in many applications [6]. Moreover, the CSM's requirements of the surroundings are very structured and would not currently be portable. In an effort to avoid these limitations, this research assumes vision is not a necessary attribute for effecting manipulation.