Spatial Reasoning and Robotics

Mikhail Soutchanski
Department of Computer Science
University of Toronto

mes@cs.toronto.edu

Slides for the depth oral examination
Friday May 31, 1996
1:30 PM, LP266

Motivating Example

There is the difficult theoretical problem: how to integrate task level planning and motion planning.

One of the major aspects of this problem is how to check that the task level specification is feasible at the motion level ?

Example.

Suppose that robot is given the task level specification

$$pickup(book, p_1); move(p_1, p_2); puton(book, p_2)$$

where p_1 and p_2 are the initial and the final positions, respectively. The robot job is to execute this sequence of actions. But before robot starts moving, it must check whether there is a path in the free space from p_1 to p_2.

To find a path in the free space, certain geometric information about the environment should be available. Information about shapes of obstacles, clearances amidst them can be available on several levels of details.

Talk Outline

Imagine that information varies along the scale characterized,
• from the one hand, by situations when numerically precise geometric information is given beforehand,
• and from the other hand, by situations when no information is given.
All informatic and motion situations can be categorized as follows.

1. Exact information and accurate control. In this category we can distinguish complete algorithms for path planning and incomplete or approximate algorithms for path planning.
2. Path planning under uncertainty in motion or geometrical model. Algorithms for planning rely on sensing during the motion to compensate uncertainties in motion control.
3. Qualitative (topological) spatial information. Sometimes, given qualitative description one can significantly simplify reasoning about the evolution of qualitatively significant properties.
4. Absence of prior information. The main slogan here is "Learn more to know".

Exact information and control

Assume we have precise geometric description of the work-space and all motion commands can be implemented without errors. If we don't consider kinematic constraints on motion, then navigation problem for a body amidst stationary obstacles reduces to the search for a path in the free space from the initial to the final position.

It is convenient to consider this problem not in the original work-space, but in a space where obstacles are enlarged by the shape of the moving body. Axes of this new *configuration space* correspond to degrees of freedom of the moving body, and the body itself is represented by a point.

Example. Consider motion of a circle in the work-space occupied by polygonal obstacles. Let the origin of moving frame be the center of this circle. Because circle is invariant with respect to rotations around its center, the configuration space is two-dimensional.

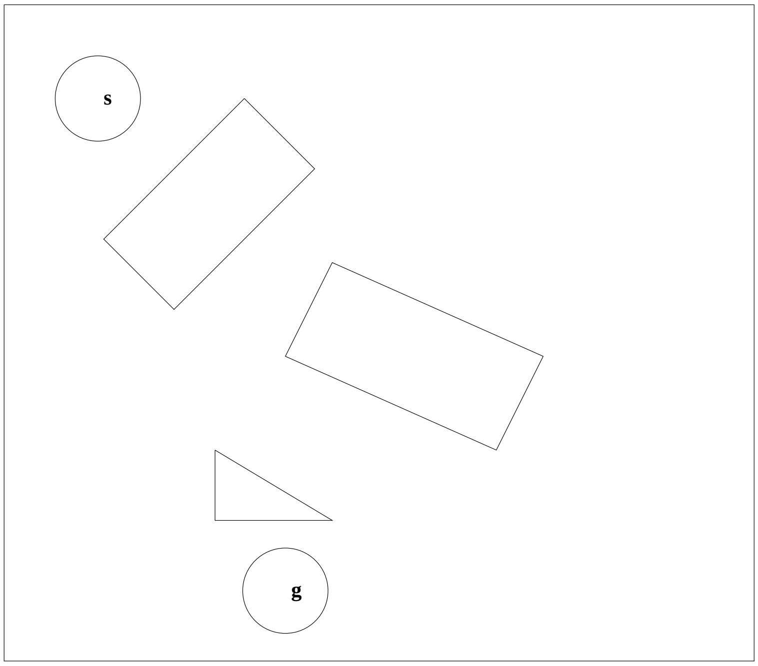

Figure 1: The `Findpath` problem for a circle.

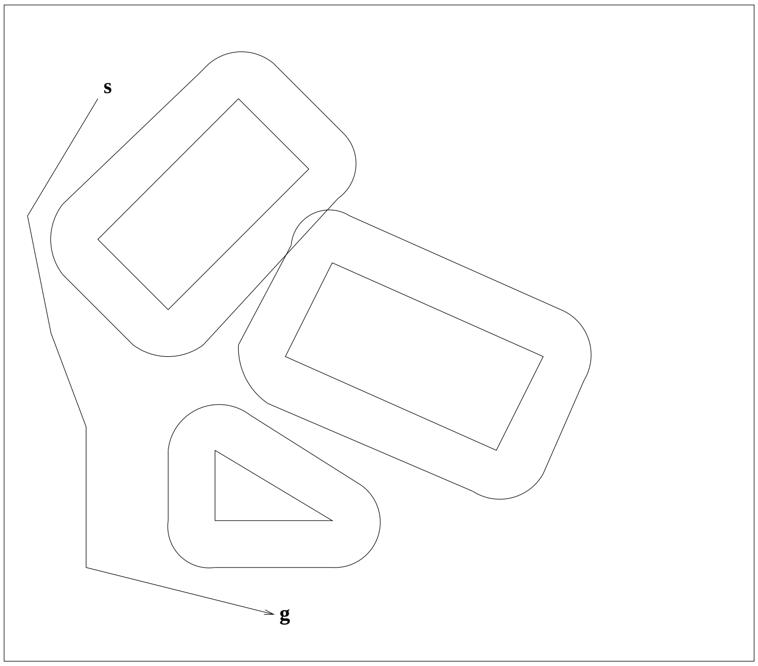

Figure 2: The corresponding problem in the configuration space.

Uncertainty

There are several aspects of uncertainty:
- uncertainty about the initial position;
- control uncertainty;
- perceptual uncertainty;
- model uncertainty.

There are two main sub-classes.

First, when geometric information and motion commands are distorted by an additive random noise and robot is able to perform several measurements to approximate values of uncertain parameters.

Second, when measurements and commands are corrupted by systematic disturbances and nothing is known about the nature of these disturbances besides that errors belong to bounded regions.

Important ideas from this area of research:

- preimage backchaining - it is geometric counterpart to the idea of regression in abstract planning;
- the planner must reason about what information might be available during the execution because perceptive actions allow to select correct branches of conditional plans and thus compensate the model and initial uncertainties.
- distinction between goal reachability and goal recognizability.

The most well-developed area of research are algorithms for *compliant motion*. They use obstacles as landmarks. If map is given, planning algorithm anticipates that if motion stick to certain obstacles and follow along their boundaries, then robot is guaranteed to reach the goal.

Qualitative information

Qualitative (topological) information can be metrically grounded, then it may support mixed types of reasoning.

The most well-established and comprehensive research area in this category is qualitative kinematics.

The most important idea is that configuration space can be divided into certain *places* (regions) which correspond to qualitatively indistinguishable kinematic states of a mechanism. Boundaries between places corresponds to qualitatively important changes.

Adjacent places of the configuration space form, respectively, adjacent vertices of the graph. Any real motion of bodies corresponds to a traverse along edges of this graph. Thus, this graph can be employed for prediction of the long-term behavior of a mechanical system.

The main limitation is that the graph can be monstrously large. The methods developed in qualitative kinematics are very sensitive to geometric details in shapes. For similar ideas be applicable to navigation planning in relatively uncluttered environments, original shapes have to be approximated.

If qualitative information is not abstracted out of numerical data, reasoning about space and time can proceed only on the purely qualitative level. Usually, purely qualitative reasoning is not sufficient for navigation planning.

There are several approaches to formalization of notions related to motion in the first-order logic.

In this area, researchers usually consider temporal evolution of relational descriptions which have some 'spatial flavor'. Typical example is the theory of space based on the primitive relation "connected" between objects named regions.

Lack of information

Different types of information can be absent. We can distinguish two sub-classes.

The first case is when there is no prior information about the environment; in particular, environment can contain unknown obstacles. Several algorithms have been proposed for learning a map from sensor data in a stationary environment. Some algorithms extract only topological information about routes in the environment, while other learn an occupancy map.

Most of proposed algorithms do not rely upon general knowledge about spatial properties in the environment. Meanwhile, in all indoor environments there are certain points of topological significance (corridor junctions, doors) where sensor data change abruptly. Representation of knowledge about these points may facilitate assimilation of sensor data and extraction important high-level features.

The second case is when there is no prior information about uncertainties in motion commands or when preliminary motion skills are inadequate for successful completion of a task.

Skill acquisition algorithms learn how to do an action quickly and correctly through training examples.

Future Work

- Representation. What representation of space is useful and sufficient both for reasoning about motion and for navigation planning ?
- Closely related problem: how represent in a version of the situation calculus interaction between high-level specifications for action and low-level motion planning procedures.
- Solution of the previous problem is necessary to solve the 'intelligent' version of the `Findpath` problem: how reactive and planning aspects should be represented in the high-level language ?
- Actions with probabilistic effects. Any serious research dealing with real equipment should address issues of noise and uncertainty.

Why we need logic?

- Logic is the perfect language for high-level specifications because it has well-understood *semantics*.
- The high-level language is more *convenient* for humans than a low-level language which allows programmers to write only commands like $move(1.7, 5.3, 8.2)$. This and similar commands are difficult to program because exact numerical values of locations, orientations and forces have to be specified.
- Logic provides the mathematically precise framework for description of (general) properties for some environments, e.g., in-door office-type environments. This knowledge representation framework is available even if description of the particular environment is *incomplete*. This framework may simplify the process of assimilating the sensor information. For example, one can anticipate abrupt changes in sensor readings because of changes in viewpoint when a robot moves from a corridor inside a room or pass through a junction of corridors. Consequently, representation of knowledge about these topologically significant places facilitates map construction and extraction of high-level features from sensor data.
- Logic provides natural tools for *verification* of task level plan given the correct model of the world. It also provide the guidelines for the verification process of the low level routines and interaction between them.

- Logic facilitates *‘intelligent’ obstacle avoidance*. If an execution of a high-level plan fail due to unexpected obstacles, logical methods of inference together with search control techniques provides the basis for making the appropriate decisions. These decisions may involve partial re-planning or invoking problem-specific handlers from the pre-compiled library of correct routines. In some cases, real-time partial re-planning and/or routines for dealing with specific problems may help to recover from the failure.
- The text-book example: monkey-banana problem.