Online Planning for Sensing Objects

Feng Wu

Willow Garage, Inc.

September 9, 2010
Outline

1 Motivation

2 Online Planning

3 Demo on PR2

4 Conclusion
Various Sensing Abilities of PR2

Sensors Everywhere:

- Stereo cameras in the head.
- Tilting laser range finder.
- Base laser range finder.
Wide Sensing vs. Narrow Sensing

**Wide Sense**
- Long distance
- Wide range
- Less time
- Noisy

**Narrow Sense**
- Short distance
- Narrow range
- More time
- Accurate

Our motivated goal:
- Combine information from different sensors.
- Actively plan when to sense and act.
- Choose the most useful sensing actions.

Feng Wu (Summer Intern)
September 9, 2010
Partially Observable Markov Decision Process (POMDP) is a useful mathematical framework for modeling different type of sensing actions:

- States $S$, Actions $A$, Transition $T(s'|s,a)$, and Reward $R(s,a)$.
- Observations $\Omega$, and Observation Probability $O(o|s',a)$.
- Belief: $b \in \Delta(S)$, a probability distribution over states given the initial state distribution and the history of actions and observations.

We are going to plan in the belief space. This will allow us to evaluate the tradeoffs between different sensing, navigation, and manipulation actions.

The planner is efficient and runs online on PR2.
Finding Objects: A Case Study

There are several objects randomly located in an environment. PR2 needs to find them as quickly as possible. PR2 can use different types of sensing actions (wide or narrow sensing) to figure out where they are.

- **Locations:** \( n \) discretized search places in the map.
- **Actions:** move to any locations and do wide or narrow sensing.
  - Narrow Sense: `tabletop_object Detector`.
  - Wide Sense: `table_clutter_detector`.
  - Pickup: `object_manipulator`.
  - Move: `move_base`
- **Observations:** returns of sensing, see objects in some locations.
- **States:** objects exist in some locations. The number of objects and their locations are **unknown**. The size of the state space is **unbounded**.

We wrote `table_clutter_detector` based on pcl and using tilt scan.
Online Planning

1. Motivation

2. Online Planning

3. Demo on PR2

4. Conclusion
We designed and implemented the model, sampler and planner.
Sample-Based Belief Monitoring

Monte-Carlo Belief Sampling:
1. Draw a state $s$ from the current belief.
2. Draw next state $s'$ from transition.
3. Compute the weight of $s'$ based on $o$.
4. Repeat 1-3 until have enough samples.
5. Compute belief set from the samples.

Belief for empty area: keep track the visited locations without objects.
Belief for object clusters: probability distribution over possible object locations.
The AND/OR Tree Search:

1. Expand the OR nodes for each action.
2. Choose the $K$-best AND nodes.
3. Sample beliefs for the $K$-best nodes.
4. Repeat 1-3 until the end of time.
5. Backtrace and update the AND values.
6. Run the best action for belief $b_0$.

Heuristic: belief entropy — minimize the uncertainty of objects’ possible locations.

Reward: huge reward for finding an object and cost for each sensing action.

It is an anytime algorithm and gives reasonable plans when we run it for 3 seconds.
Example of Planning for Sensing

Demonstration on Stage, when there is no object in the environment.

We wrote `stage_extensions` that supports additional sensors, gripper.
Demo on PR2

1 Motivation

2 Online Planning

3 Demo on PR2

4 Conclusion
Demonstration 1: One Table
Demonstration 2: Two Tables
Conclusion

1 Motivation

2 Online Planning

3 Demo on PR2

4 Conclusion
Summary

Probabilistic reasoning and planning framework for PR2:
- A decision-theoretical planning model for finding objects.
- A Monte-Carlo sampling algorithm for belief monitoring.
- An online planning algorithm using bounded resources.

All the source codes are available on:

```bash
$ svn co https://code.ros.org/svn/wg-ros-pkg/branches/trunk_cturtle/stacks/find_object
```

```bash
$ ls find_object
find_object_actions find_object_executive find_object_planner
table_boundary_detector table_clutter_detector
stage_extensions
```
Towards more intelligent personal robots:

- Incorporate camera data for long-distance object detection.
- Reason the relationship between objects, e.g. when PR2 sees a kitchen, it knows there will be more chance to find food.
- Generate the model dynamically and do the planning, e.g. PR2 scans the whole room and finds the possible locations for searching objects. Adam’s multi_table_detector node.
- Use the action execution time to plan due to the anytime property.
Thank you!

- The manipulation and navigation team for their code.
- Radu Bogdan Rusu for a lot of help on pcl.
- Mentor: Bhaskara Marthi.
- All the other Interns.
- WillowGarage, Inc.