Sub-optimal Heuristic Search and its Application to Planning for Manipulation

April 26, 2014

Lecturer: Michael Phillips

Scribes: Hongyun Huang, Qing Liu, Chi Liu, Jia Zeng

Content

1. Suboptimal heuristic searches can be fast enough for high-D motion planning and return consistent good quality solution
   1.1 Planning with optimal heuristic search-based approaches
   1.2 Planning with sampling based approaches
   2. Example of planning for 20D arm in 2D

3. Suboptimal heuristic searches for High D Motion planning
4. Suboptimal heuristic searches for High D Motion planning
5. Anytime Search based on weighted A*
6. ARA*
   6.1 ARA* - based Planning for manipulation
   6.2 Plan with Experience Graphs
   6.3 Plan with E-Graphs for Mobile Manipulation

7. Conclusions
1. Suboptimal heuristic searches can be fast enough for high-D motion planning and return consistent good quality solution

1.1 Planning with optimal heuristic search-based approaches (e.g. A*, D*)
   ○ Advantages:
     ■ Completeness/optimality guarantees
     ■ Excellent cost minimization
     ■ Consistent solution to similar motion queries
   ○ Disadvantages:
     ■ Slow and Memory intensive
     ■ Used mostly for 2D planning

1.2 Planning with sampling based approaches (e.g. RRT, PRM)
   ○ Advantage:
     ■ Typically very fast and low memory
     ■ Used for high D motion planning
   ○ Disadvantage:
     ■ Completeness in the limit
     ■ Often poor cost minimization
     ■ Hard to provide solution consistency

2. Example of planning for 20D arm in 2D

![Figure 1: A 20D arm represented on 2D to illustrate the number of states available for planning](image)

- Each state is defined by 20 discretized joint angle (q1, q2...q20)
- Each action is changing one angle at a time.
- States are generated on-the-fly as search expands states
3. Suboptimal heuristic searches for High D Motion planning

- ARA* (Anytime version of A*) graph search: Effective use of solutions to relax motion planning problems
- Experience graphs: Heuristic search that learn from its planning experience

4. Suboptimal heuristic searches for High D Motion planning

![Diagram of A* search](image)

Figure 2: A* search. g(s) is the coming cost, and h(s) is the heuristic cost to the goal.

- A* Search: expand states in the order of $f=g+h$ values
- In A*, when $s$ is expanded g(s) is optimal
- h(s) is the simplified version of the problem
5. Anytime Search based on weighted A*

![Diagram showing different expansions and solutions for various ε values]

- **Constructing anytime search based on weighted A***:
  a. Find the best path possible given some amount of time for planning
  b. Do it by running a series of weighted search with decreasing ε

- **Inefficient because**
  c. Many state values remain the same between search iterations
  d. We should be able to reuse the results of previous searches

6. ARA*

- Efficient series of weighted A* searches with decreasing ε

```plaintext
Compute Path with Reuse function
while (f(s_goal) > minimum f-value in OPEN)
remove s with the smallest [g(s) + εh(s)] from OPEN:
insert s into CLOSED.
for every successors’ of s
  if g(s’) > g(s) + c(s,s’)
    g(s’) = g(s) + c(s,s’);
    if s’ not in CLOSED then insert s’ into OPEN;
  otherwise insert s’ into INCONS
set ε to large value;
g(s_start) = 0; OPEN = {s_start};
```

Figure 3: Weighted A* searches with decreasing ε
CLOSED = {}; INCONS = {}; 
ComputePathwithReuse(); 
publish current $\epsilon$ suboptimal solutions; 
decrease $\epsilon$; 
Inirialize OPEN = OPEN $\cup$ INCONS

When using ARA*, the research is in finding a graph representation $G$ and a corresponding heuristic function $h$ that lead to shallow local minima for the search.

### 6.1 ARA* - based Planning for manipulation

- **Pros**
  - Explicit cost minimization
  - Consistent plans (similar inputs generate similar outputs)
  - Can handle arbitrary goal sets
  - Many path constraints are easy to implement
- **Cons**
  - Requires good path and heuristic design to plan fast
  - Number of motions from a state affects speed and path quality

### 6.2 Plan with Experience Graphs

- Many planning tasks are repetitive
  - Loading a dishwasher
  - Opening doors
  - Moving objects around a warehouse
- Can we reuse prior experience to accelerate planning, in the context of search-based planning?
- Would be especially useful for high-dimensional problems such as mobile manipulation!

### 6.3 Plan with E-Graphs for Mobile Manipulation

- **Pros**
  - Can lead the search around local minima (such as obstacles)
  - Provided paths can have arbitrary quality
  - Can reuse many parts of multiple experiences
○ User can control bound on solution quality

• Cons
○ Local minima can be created getting on to the E-Graph
○ The E-Graph can grow arbitrarily large

7. Conclusions

• Heuristic Search within sub-optimally bounds
○ Can be much more suitable for high-dimensional planning for manipulation heuristic search
○ Still provides
  ■ Good cost minimization
  ■ Consistent behavior
  ■ Rigorous theoretical guarantees

• Planning for manipulation solves similar tasks over and over again
○ Great opportunity for combining planning with learning