CSE634: Algorithms in Robot Planning
Assignment-3

Akanksha Singal (2021008)

December 5, 2024

Question-1

The Rapidly-exploring Random Tree (RRT) algorithm constructs a tree via an incremental sampling-based approach. It is possible to create a graph instead of a tree while keeping the concept of the algorithm intact.

1. Extend the algorithm for RRT for creating a graph instead of a tree.
2. Which important feature does the new algorithm add to the existing properties of RRT algorithm?

Objective:

To extend the algorithm for RRT for creating a graph instead of a tree.

Solution:

To modify the Rapidly-exploring Random Tree (RRT) algorithm to construct a graph instead of a tree, we can introduce the concept of connecting new nodes to multiple nearby nodes rather than just the nearest one. The paper [2] talks about how we can construct rapidly exploring random graphs.

0.1 Algorithm:

To construct a RRG G = (V, E) we follow the steps below:

- **Sampling:** Randomly sample a configuration q_{rand} in the configuration space.
- **Nearest Nodes Selection:** Instead of finding just the nearest node q_{near}, identify all nodes within a certain radius r of q_{rand}. This set is denoted as Q_{near}.
- **Steering:** For each node q in Q_{near}, attempt to steer from q towards q_{rand} to generate a new node q_{new}. Ensure that the path from q to q_{new} is collision-free.
- **Graph Expansion:**
  - **Add Node:** Add q_{new} to the graph.
  - **Add Edges:** For each node q in Q_{near}, if the path between q and q_{new} is collision-free, add an edge between them.
  - **Bidirectional Edges:** Since it's a graph, edges can be bidirectional, allowing movement from q_{new} to q and vice versa.
- **Repeat:** Continue sampling and expanding the graph until the goal is reached or a stopping criterion is met.

Algorithm 1 Rapidly-exploring Random Graph

Input: Obstacle free configuration space Q_{free}, start configuration q_{start}, goal $\mathcal{X}_{\text{goal}}$, step size d, radius r
Output: A graph $G = (V, E)$ connecting q_{start} to the goal
Initialize $G = (V, E)$ with $V = \{q_{\text{start}}\}$ and $E = \emptyset$
while True **do**
  $q_{\text{rand}} \leftarrow \text{Sample}(Q_{\text{free}})$
  $Q_{\text{near}} \leftarrow \{q \in V : \|q - q_{\text{rand}}\| \leq r\}$
  for $q \in Q_{\text{near}}$ **do**
    $q_{\text{new}} \leftarrow \text{Steer}(q, q_{\text{rand}}, \text{d})$
    if (q, q_{new}) path is collision free **then**
      $V \leftarrow V \cup \{q_{\text{new}}\}$
      $E \leftarrow E \cup \{(q, q_{\text{new}}), (q_{\text{new}}, q)\}$
  if $q_{\text{new}} \in \mathcal{X}_{\text{goal}}$ **then**
    break
return $G = (V, E)$ =0

Important additions:

1. The paper [2] proves advantages compared to RRT regarding run duration and traveled path length.
2. By connecting nodes to multiple neighbors, the algorithm can explore various paths between the start and goal configurations.
3. The graph captures the connectivity of the free space more thoroughly than a tree and can be beneficial in dynamic configuration space.

Question 2

We discussed multi-robot coordination with inconsistent beliefs of the robots. In algorithm **VerifyAC**, the action selection policy defines the rank-1 action as the "most preferred" action both in steps 2 and 3 of the algorithm. However, it is a rigid criteria to satisfy rank-1. It is possible to generalize the notion of most preferred action by selecting some rank i and j actions, where $1 \leq i, j \leq n$ and $i, j \in \mathbb{Z}^+$, in steps 2 and 3 respectively (instead of restricting it to rank-1, i.e., $i = j = 1$).

(a) What are the pros and cons of the above generalization?

(b) Modify the mathematical expression (8) in the paper *Kundu2024iros* to capture the generalized notion of most preferred action.

Solution:

(a) What are the pros and cons of the above generalization? subsection*Pros

- Selecting actions based on rank i and j allows the algorithm to adapt to varying priorities or constraints, accommodating broader use cases.
- By considering a broader range of preferences, the system can avoid being overly rigid, which might lead to suboptimal outcomes when rank-1 actions are not globally optimal.
- It facilitates exploring diverse strategies, which can be beneficial in dynamic environments where robots may need to alternate priorities.
- If rank-1 consistency is too strict and leads to frequent communication, relaxing the rank constraint might reduce unnecessary triggers.

Cons

- A looser definition of "most preferred" increases the complexity of ensuring that actions selected by different robots are consistent.
- May result in choices that are not optimal.
- Might introduce ambiguity if the preference rankings of the robots do not align.
- and comparing actions increases computational effort, particularly in decentralized settings.

(b) Modify the mathematical expression in the paper [1]

Original Expression:

$$\forall b \in B_Z,\, a \succeq a' \iff J(b,a) \geq J(b,a').$$

Generalized Notion: To generalize this expression for rank-i and rank-j actions, let A_i and A_j denote the sets of actions corresponding to ranks i and j, respectively. The generalized expression can be written as:

$$\forall b \in B_Z,\, a_i \in A_i,\, a_j \in A_j,\, a_i \succeq a_j \iff J(b,a_i) \geq J(b,a_j).$$

Here:

- A_i and A_j are subsets of the joint action space A, representing actions of rank i and j, respectively.
- The inequality $J(b,a_i) \geq J(b,a_j)$ reflects the generalized preference ordering based on the modified rank constraints.

This generalization ensures that actions are evaluated and compared across flexible ranks while retaining the structure of preference evaluation.

References

[1] Tanmoy Kundu, Moshe Rafaeli, and Vadim Indelman. Multi-robot communication-aware cooperative belief space planning with inconsistent beliefs: An action-consistent approach, 2024.

[2] Marco Steinbrink, Philipp Koch, Bernhard Jung, and Stefan May. Rapidly-exploring random graph next-best view exploration for ground vehicles. In *2021 European Conference on Mobile Robots (ECMR)*, pages 1–7, 2021.