FUEL Learning Poster

Tags: Motion Planning, Flying Robots

(Reference: Zhou, Boyu, et al. "Fuel: Fast uav exploration using incremental frontier structure and hierarchical planning." IEEE Robotics and Automation Letters 6.2 (2021): 779-786.)

Problems

1. Greedy exploration leads to <u>local optimality</u>, and most methods are <u>conservative</u>. → Solved by <u>hierarchical planning</u>.
2. <u>Low-speed exploration</u>, disallows fully exploit dynamic capability to fulfill the mission.
3. Due to <u>high computational overhead</u>, quadrotor can not respond quickly and frequently to environmental changes.

Solve

Solve

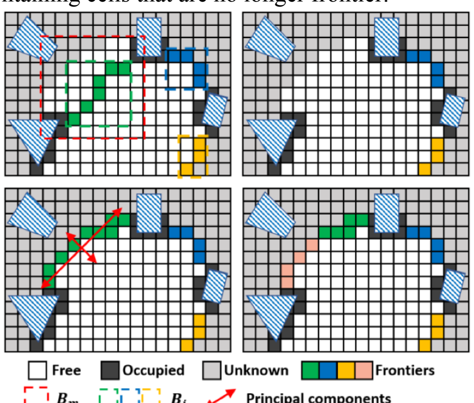
Solve

Eliminate unaffected clusters in a <u>fast</u> way, significantly <u>reduce</u> number of <u>expensive precise checks</u>.

A. Remove <u>outdated frontier clusters</u>:

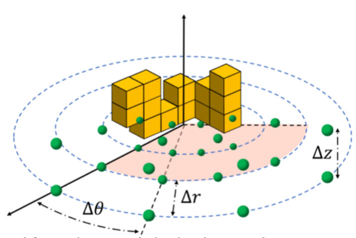
a. Record the AABB B_m of the updated region, select clusters whose AABBs (B_i) intersect with B_m.

b. Precise check on selected clusters, remove clusters containing cells that are no longer frontier.



B. Frontier Detection:
Search and cluster new frontier clusters by region growing algorithm, ignore small ones (typically from noise).

C. Clustering:
Recursively conduct PCA until <u>all large clusters are divided into small ones</u>.



A. Uniformly sampled viewpoints set $VP_i = \{x_{i,1}, x_{i,2}, \ldots, x_{i,n_i}\}$, $x_{i,j} = (p_{i,j}, \xi_{i,j})$

B. Filter (at most reserve N_{view} viewpoints) and sort VP_i in descending order by coverage.

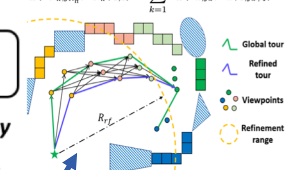
Updating when new frontiers are detected, leads to <u>more frequent</u> and <u>faster speed</u> exploration.

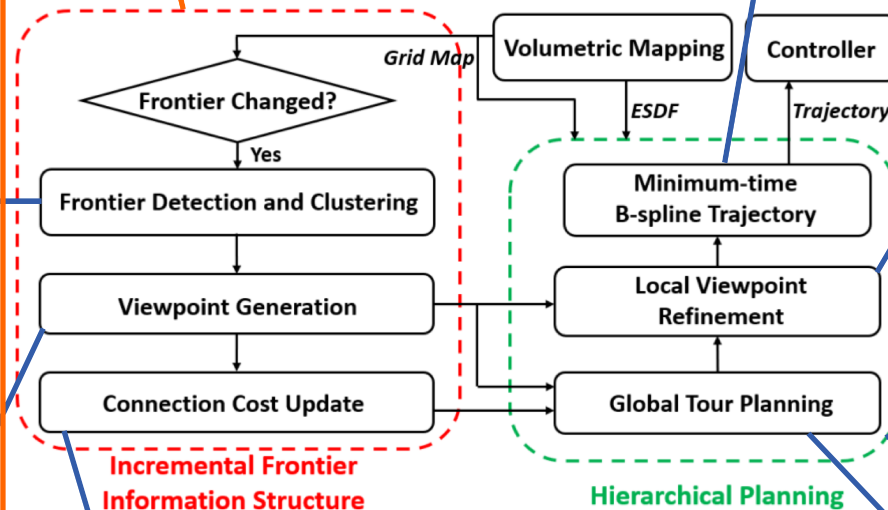
Minimum-time B-Spline Trajectory:
Generate <u>smooth</u>, <u>safe</u> and <u>dynamically feasible</u> B-spline trajectories

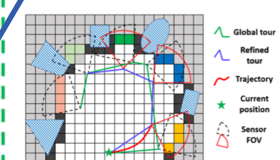
$$\underset{\mathbf{X}_{c,b},\Delta t_b}{\arg\min}\ f_s + w_t T + \lambda_c f_c + \lambda_d (f_v + f_a) + \lambda_{bs} f_{bs}$$

Local Viewpoint Refinement:
Only a single viewpoint of each cluster -> <u>a richer set of viewpoints</u> on a <u>truncated segment</u> of the global tour.
Use Dijkstra to find optimal local tour $\Xi = \{\mathbf{x}_{1,j_1}, \mathbf{x}_{2,j_2}, \cdots, \mathbf{x}_{N_{rf},j_{N_{rf}}}\}$ by min:

$$c_{rf}(\Xi) = t_{lb}(\mathbf{x}_0, \mathbf{x}_{1,j_1}) + w_c \cdot c_c(\mathbf{x}_{1,j_1}) + t_{lb}(\mathbf{x}_{N_{rf},j_{N_{rf}}}, \mathbf{x}_{N_{rf}+1,1}) + \sum_{k=1}^{N_{rf}-1} t_{lb}(\mathbf{x}_{k,j_k}, \mathbf{x}_{k+1,j_{k+1}})$$







coarse-to-fine steps

Select viewpoints with <u>highest coverage</u> to estimate connection cost as:

$$t_{lb}(x_{k_1,1}, x_{k_2,1}) = \max\left\{\frac{\mathrm{len}\left(P(p_{k_1,1}, P_{k_2,1})\right)}{v_{max}}, \frac{\min\left(|\xi_{k_1,1}-\xi_{k_2,1}|, 2\pi-|\xi_{k_1,1}-\xi_{k_2,1}|\right)}{\xi_{max}}\right\}$$

$P(p_{k_1,1}, P_{k_2,1})$ is searched using A*.

Connection cost computed in an <u>incremental</u> manner:
a. Above update scheme takes $O(k_{new} \cdot N_{cls})$ time.
b. k_{new} can be regarded as a constant factor, resulting in a <u>linear time</u> update.

Global Exploration Tour Planning:
Compute an open-loop tour starting from current and passing all clusters by <u>ATSP</u>.
For cost matrix M_{tsp}:

$$\mathbf{M}_{tsp}(k_1, k_2) = \mathbf{M}_{tsp}(k_2, k_1) = t_{lb}(\mathbf{x}_{k_1,1}, \mathbf{x}_{k_2,1}),\ k_1, k_2 \in \{1, 2, \cdots, N_{cls}\}$$

$$\mathbf{M}_{tsp}(0, k) = t_{lb}(\mathbf{x}_0, \mathbf{x}_{k,1}) + w_c \cdot c_c(\mathbf{x}_{k,1}),\quad k \in \{1, 2, \cdots, N_{cls}\}$$

$$c_c(\mathbf{x}_{k,j}) = \cos^{-1}\frac{(\mathbf{p}_{k,j} - \mathbf{p}_0)\cdot \mathbf{v}_0}{\|\mathbf{p}_{k,j} - \mathbf{p}_0\|\,\|\mathbf{v}_0\|}$$

$$\mathbf{M}_{tsp}(k, 0) = 0,\ k \in \{0, 1, 2, \cdots, N_{cls}\}$$

Summarization and Personal Thinking

Summarization:

1. Innovation:
   a. Maintained FIS (Frontier Incremental Structure) for hierarchical planning:
      Due to this <u>incremental</u> updated structure, quadrotor <u>makes decisions at high frequency</u> to <u>respond quickly</u> to environmental changes.
   b. Coarse-to-fine hierarchical planner for <u>global coverage</u> and <u>reducing conservation</u>:
      b.1 For global optimality, it firstly finds efficient global tours (coarse).
      b.2 Then selecting a local set of optimal viewpoints for refinement.
      b.3 Minimum-time B-spline local trajectories are generated at last to guarantee <u>safety</u> and <u>dynamical feasibility</u>.
2. Limitation:
   Error in state estimation should not be ignored.

Future Work and Personal thinking:

1. Consider state estimation uncertainty, and evaluate under pose drifts (mentioned in paper).
2. Consider about methods which model planner as learned dynamics (using reinforcement learning or neural-network with mathematical constraints), FUEL can be combined with these "learning" related methods.
3. Expert demonstration learning and intervention learning could be applied in this scenario, and "trial and error" methods are also suitable in fast UAV motion planning.