On Parameters Settings in Multi-Robot Exploration Strategies

Jan Faigl, Petr Vaněk

Laboratory for Computational Robotics

Agent Technology Center / Department of Computer Science

Czech Technical University in Prague

August 18, 2014

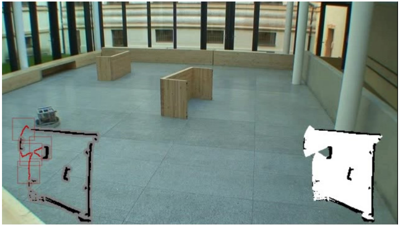
Outline

- Multi-Robot Exploration - problem formulation
- Overview of the goal assignment methods
- Local decision-making towards global optimization
  - Distance cost vs time to travel
  - Task-allocation with return to depot
  - Goal assignment with small number of goal candidates
- Influence of replanning period
- Evaluation methodology and results

Multi-Robot Exploration of Unknown Environment

A problem to create a map of the given unknown environment

- Frontier-based approach
  Yamauchi (1997)
- Occupancy grid
  Moravec and Elfes (1985)
- Laser scanner sensor
- Next-best-view approach
  Select the next robots' goals



Performance metric:

Time to create the map of the whole environment

search and rescue mission

Multi-Robot Exploration as Task-Allocation Problem

At the time t of the local decision-making we have:

- A current map of the environment $\mathcal{M}$
- A set of m robots at positions $\boldsymbol{R}(t) = \{r_1, r_2, \ldots, r_m\}$
- A set of n goal candidates be $\boldsymbol{G}(t) = \{g_1, \ldots, g_n\}$

e.g., frontiers

We need to assign a goal $g \in \boldsymbol{G}(t)$ for each robot $r \in \boldsymbol{R}(t)$ that will minimize the required time to explore the environment.

The problem is formulated as the task-allocation problem

$$(\langle r_1, g_{r_1}\rangle, \ldots, \langle r_m, g_{r_m}\rangle) = \mathrm{assign}(\boldsymbol{R}(t), \boldsymbol{G}(t), \mathcal{M}),$$

*We call the determination of **G** and assignment procedure the **exploration strategy**.*

We consider only the distance cost for the assignment

Multi-Robot Exploration as Task-Allocation Problem

At the time t of the local decision-making we have:

- A current map of the environment $\mathcal{M}$
- A set of m robots at positions $\boldsymbol{R}(t) = \{r_1, r_2, \ldots, r_m\}$
- A set of n goal candidates be $\boldsymbol{G}(t) = \{g_1, \ldots, g_n\}$

e.g., frontiers

We need to assign a goal $g \in \boldsymbol{G}(t)$ for each robot $r \in \boldsymbol{R}(t)$ that will minimize the required time to explore the environment.

The problem is formulated as the task-allocation problem

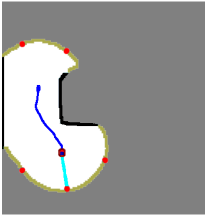
$$(\langle r_1, g_{r_1}\rangle, \ldots, \langle r_m, g_{r_m}\rangle) = \mathrm{assign}(\boldsymbol{R}(t), \boldsymbol{G}(t), \mathcal{M}),$$

We call the determination of ***G*** *and assignment procedure the* ***exploration strategy****.*

We consider only the distance cost for the assignment

Distance Cost Variants

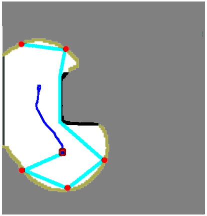
Simple robot–goal distance



- A length of the robot-goal candidate path
- Greedy goal selection

Select the closest goal candidate

TSP distance cost



Kulich M., Faigl J., Přeučil L.,
On Distance Utility in the Exploration Task,
ICRA, 2011, 4455–4460.

- Consider visitations of all goal candidates

  Solve the associated traveling salesman problem (TSP)
- A length of the tour visiting all goals
- Goal representatives

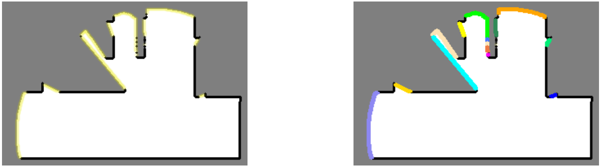
TSP distance cost improves performance about 10-30%

Goal Candidates Determination

RFE – Representative of Free Edges

Kulich & Faigl & Preucil, ICRA (2011)

a) Organize frontier cells into free edges $\boldsymbol{F_i}$
b) Select representatives of $\boldsymbol{F_i}$ from which $\boldsymbol{F_i}$ would be covered
- Means of n_r clusters found by K-means algorithm



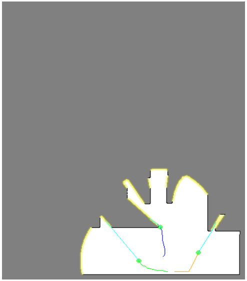
RFE is necessary for the TSP distance cost, but it also improves performance of other task-allocation methods.

Faigl J., Kulich M.
On determination of goal candidates in frontier-based multi-robot exploration,
ECMR, 2013, 210–215.

Multi-Robot Exploration Procedure

1. Initialize occupancy grid $\mathcal{O}cc$ and robot plans $\mathcal{P} = (P_1, \ldots, P_m)$
2. `Repeat`
   2.1 Navigate robots regarding $\mathcal{P}$
   2.2 Update $\mathcal{O}cc$ with new sensor measurements (with range ρ)

   `Until` **replanning condition.**



3. Update the navigation map $\mathcal{M}$ from the current $\mathcal{O}cc$.
4. Detect all frontiers $\mathcal{F}$ in the current map $\mathcal{M}$.
5. **Determine goal candidates** $\boldsymbol{G}(t)$ from the frontiers $\mathcal{F}$.
6. `If` $|\boldsymbol{G}(t)| > 0$ **assign goals to the robot**
   - $(\langle r_1, g_{r_1}\rangle, \ldots, \langle r_m, g_{r_m}\rangle) = \text{assign}(\boldsymbol{R}(t), \boldsymbol{G}(t), \mathcal{M})$
   - Plan paths $\mathcal{P} = \text{plan}(\langle r_1, g_{r_1}\rangle, \ldots, \langle r_m, g_{r_m}\rangle, \mathcal{M})$
   - Go to Step 2.
7. Stop all robots

Goal Assignment Methods

1. **GA** – Greedy assignment

   Yamauchi B, Robotics and Autonomous Systems 29, 1999

   - Randomized greedy selection of the closest goal candidate
2. **IA** – Iterative assignment

   Werger B, Mataric M, Distributed Autonomous Robotic Systems 4, 2001

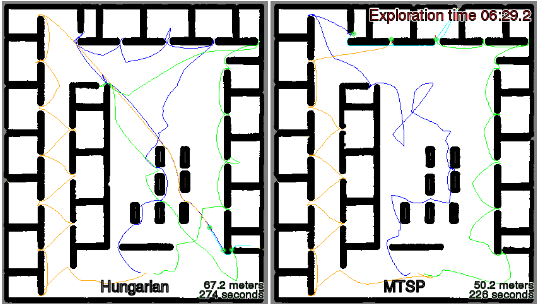
   - Centralized variant of the broadcast of local eligibility (BLE)
   - Ordering of all robot-goal pairs $\langle r, g \rangle$
3. **HA** – Hungarian assignment
   - Optimal solution of the task-allocation problem

   Stachniss C, C implementation of the Hungarian method, 2004
4. **MA** – Multiple Traveling Salesman Problem assignment
   - Based on TSP distance cost
   - ⟨*cluster-first, route-second*⟩ heuristic based on K-means

Faigl J., Kulich M., Přeučil L.
Goal Assignment using Distance Cost in Multi-Robot Exploration,
IROS, 2012, 3741–3746.

Example of Multi-Robot Exploration

Objective function: time to complete the exploration



Faigl, Kulich, Přeučil, IROS'12

- Local planner based on Smooth Nearness-Diagram (SND)

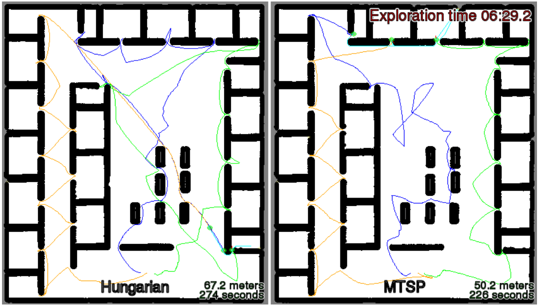
Durham, Bullo, IROS'08

- Notice, robots are slow in doors

Goals assignment is based on robot-goal distances while we are minimizing the required time to complete exploration.

Example of Multi-Robot Exploration

Objective function: time to complete the exploration



Faigl, Kulich, Přeučil, IROS'12

- Local planner based on Smooth Nearness-Diagram (SND)

Durham, Bullo, IROS'08

- Notice, robots are slow in doors

Goals assignment is based on robot-goal distances while we are minimizing the required time to complete exploration.

Local Decision-Making and Global Optimality

- We aim to minimize a global objective function using local decision-making

  It's because we collect new information during the mission

- The objective function is optimized indirectly based on the **next-best-view** approach
- A longer planning horizon in the TSP distance cost improves the exploration performance

What are other options to improve the performance by "better" local decisions?

Local Decision-Making and Global Optimality

- We aim to minimize a global objective function using local decision-making

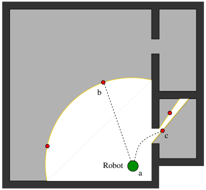
  It's because we collect new information during the mission

- The objective function is optimized indirectly based on the **next-best-view** approach
- A longer planning horizon in the TSP distance cost improves the exploration performance

What are other options to improve the performance by "better" local decisions?

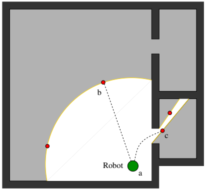
Distance Cost vs Time to Travel in Goal Assignment

- Does time to travel provide better exploration performance?
- Not necessarily, it depends on the motion model
  - Path to *b* is longer, but faster
  - Path to *c* is shorter, but slower
- Distance cost with assumption of some average speed of the robots
- Distance cost for task-allocation and time as the objective function



Distance Cost vs Time to Travel in Goal Assignment

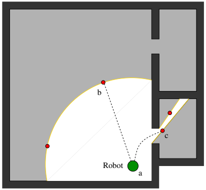
- Does time to travel provide better exploration performance?
- Not necessarily, it depends on the motion model
    - Path to *b* is longer, but faster
    - Path to *c* is shorter, but slower



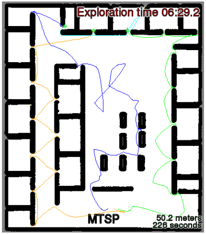
- Distance cost with assumption of some average speed of the robots
- Distance cost for task-allocation and time as the objective function

Distance Cost vs Time to Travel in Goal Assignment

- Does time to travel provide better exploration performance?
- Not necessarily, it depends on the motion model
  - Path to *b* is longer, but faster
  - Path to *c* is shorter, but slower



- Distance cost with assumption of some average speed of the robots
- Distance cost for task-allocation and time as the objective function



Exploration with the Return to Depot

- The robots collect a large dataset during the exploration

  e.g., high resolution images

- The dataset cannot be transmitted due to a limited communication bandwidth
- The robots are requested to return to the depot at the end of the mission

Bautin et al., IROS'13

- We propose modifications of the local decision-making:
  - **HAd** – the cost for the goal assignment is computed as

    $$c(r, g) = d(r, g) + d(g, depot) + \text{traveled\_distance}(r).$$

  - **MAd** – the TSP distance cost include a return trip from the last goal of the route to the depot

Exploration with the Return to Depot

- The robots collect a large dataset during the exploration

  e.g., high resolution images
- The dataset cannot be transmitted due to a limited communication bandwidth
- The robots are requested to return to the depot at the end of the mission

Bautin et al., IROS'13

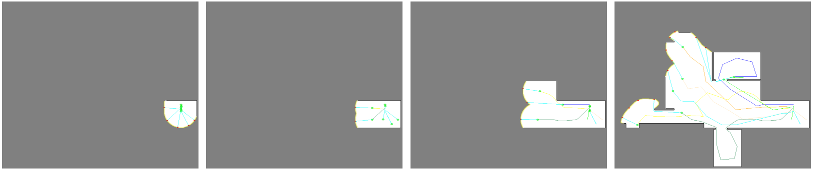
- We propose modifications of the local decision-making:
  - **HAd** – the cost for the goal assignment is computed as

$$c(r, g) = d(r, g) + d(g, depot) + \text{traveled\_distance}(r).$$

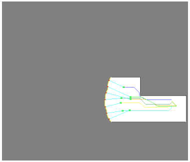
  - **MAd** – the TSP distance cost include a return trip from the last goal of the route to the depot

Too Few Goals

- Robots without assigned goal move back to the depot
- RFE does not consider number of robots in the team
- The robots may move back to the depot too soon



- A modification of the RFE with the adaptive number of representatives (**ANR**) that increase number of goal candidates of the largest free edges

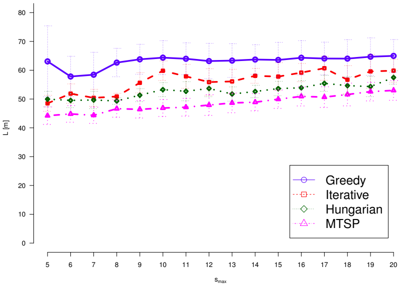


Time reduction from 1744 to 1556 discrete time steps

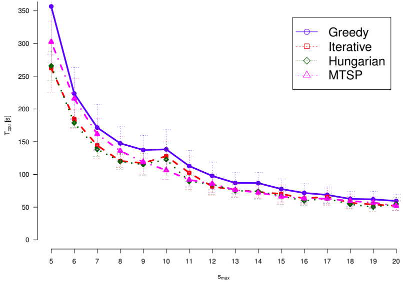
Influence of the Replanning Period

s_{max} – the replanning period in the number of motion primitives

(move forward / turn)



Average traveled distance L



Required computational time

3.2 GHz CPU

Performance of the exploration mission depends on the computational resources and also speed of the robots.

Replanning Conditions

Two replanning conditions are studied:

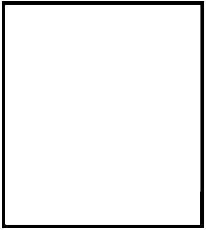
- **GR** – Goal Replanning
  - Replanning when a robot reaches its goal
  - Computationally inexpensive
- **IR** – Immediate Replanning
  - Replanning whenever a goal is no longer frontier
  - Fastest replanning possible
  - Computationally very demanding

Evaluation Methodology

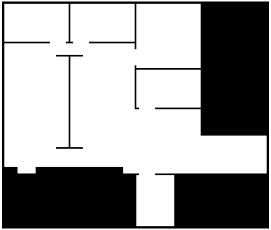
- A discrete event based simulator allowing evaluation of the computationally demanding strategies without influence of the limited computational resources
- Perturbations of the initial robot positions (20 variants)
- 4 environments: *em* - empty, *autolab*, *jh* – structured, *pot-holes* – unstructured
- No. of robots $m \in \{3, 5, 7\}$, visibility range $\rho \in \{3m, 5m, 7m\}$
- 6 assignment strategies: **GA**, **IA**, **HA**, **MA**, **HAd**, **MAd**
- 2 goal candidates determination methods: **RFE**, **ANR**
- 2 types of missions: open paths (**OP**) and return depot (**RD**)
- 2 variants of replanning: goal replanning (**GR**) and immediate replanning (**IR**)
- Statistical evaluation of 362 880 trials

20 trials per problem for stochastic strategies

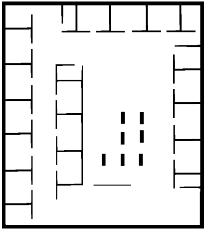
Testing Environments



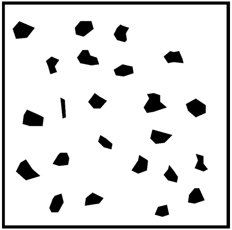
em



autolab



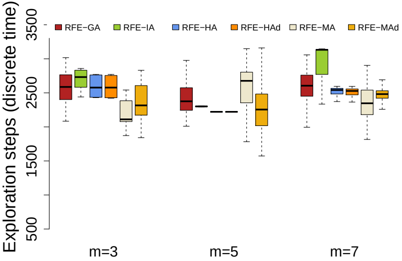
jh



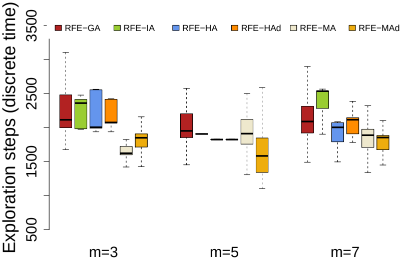
potholes

Results – Return to Depot

- Selected results for *jh* environment, visibility range ρ=7 m,
- Goal replanning (GR), and RFE goal candidates determination



return depot (**RD**) missions



open paths (**OP**) missions

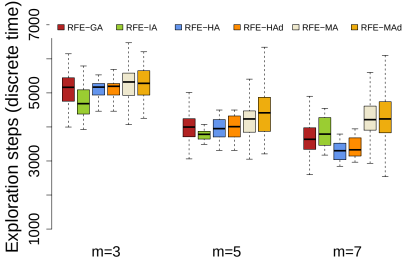
- HAd and MAd strategies do not provide expected benefits

in general

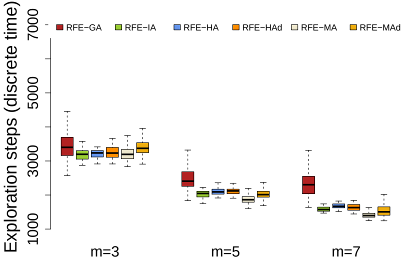
- Modifications may occasionally improve the performance

Results – Replanning Period

- Selected results for *potholes* environment, visibility range ρ=3 m,
- Open paths (OP), and RFE goal candidates determination



Goal replanning (**GR**)

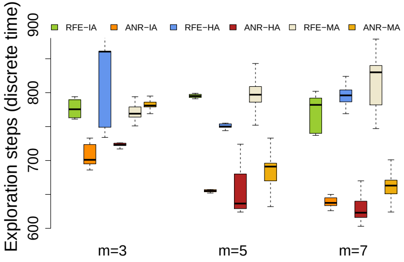


Immediate replanning (**IR**)

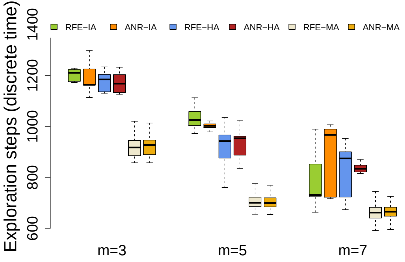
- Faster replanning significantly improves the performance
- A results of strategies comparison depends on replanning

Results – Adaptive Number of Representatives (ANR)

- Selected results for open paths (OP), and immediate replanning (IR)



autolab, ρ= 7 m



jh, ρ= 5 m

- Better in some cases, but mostly similar to RFE
- Provides a more stable the solution quality

(smaller standard deviations)

Conclusion

- There are hidden issues of the local decision-making and global optimality (*distance vs time*, *return to depot*)

  Approximations based on average expected performance provides better results than tailored decisions for local situations.

- For large teams do not forget to generate sufficient number of goal candidates.
- Although simple centralized approaches have been evaluated, the evaluation and comparison of strategies can be still tricky.

Q1: Can we learn somehow the overall approximations that can be used for local decision-making?

Q2: Is there a way to compare and evaluate exploration strategies that will provide a generalizable results?

Q3: What evaluation methodology we should use?

Conclusion

- There are hidden issues of the local decision-making and global optimality (*distance vs time*, *return to depot*)

  Approximations based on average expected performance provides better results than tailored decisions for local situations.

- For large teams do not forget to generate sufficient number of goal candidates.
- Although simple centralized approaches have been evaluated, the evaluation and comparison of strategies can be still tricky.

Q1: Can we learn somehow the overall approximations that can be used for local decision-making?

Q2: Is there a way to compare and evaluate exploration strategies that will provide a generalizable results?

Q3: What evaluation methodology we should use?

Thank You!

faiglj@fel.cvut.cz