Multi-Robot Decentralized Belief Space Planning in Unknown Environments

Tal Regev

Under the supervision of Assistant Prof. Vadim Indelman

TECHNION
Israel Institute of Technology

International Conference on Intelligent Robots and Systems (IROS 2016), Accepted

Graduate Seminar, July 2016

Introduction – Application

- Navigation and mapping in unknown environment with **multiple robots (MR)**

Navigation and mapping with MR:
Why is it interesting?

Autonomous cars

[google.com]

Space exploration

[nasa.gov]

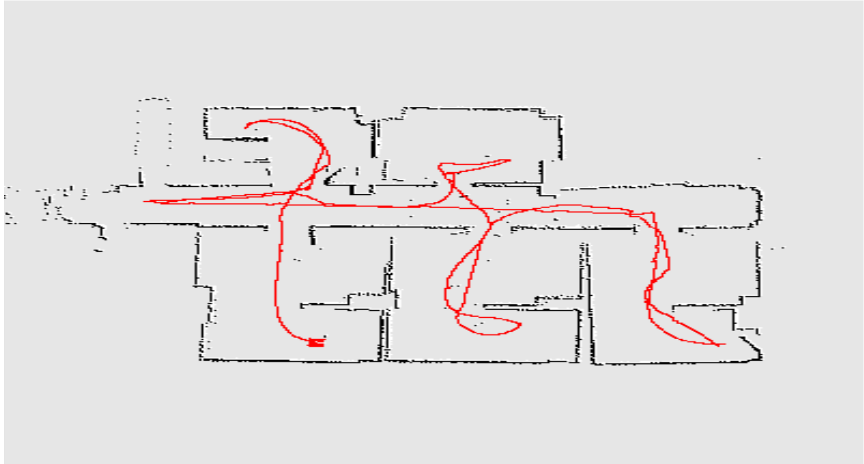
Search & rescue operations

[spectrum.ieee.org]

Introduction - SLAM

- Navigation and mapping in **unknown environment** without GPS
- Simultaneous localization and mapping (**SLAM**)

Estimation and Perception:
Where am I? What is the surrounding environment?



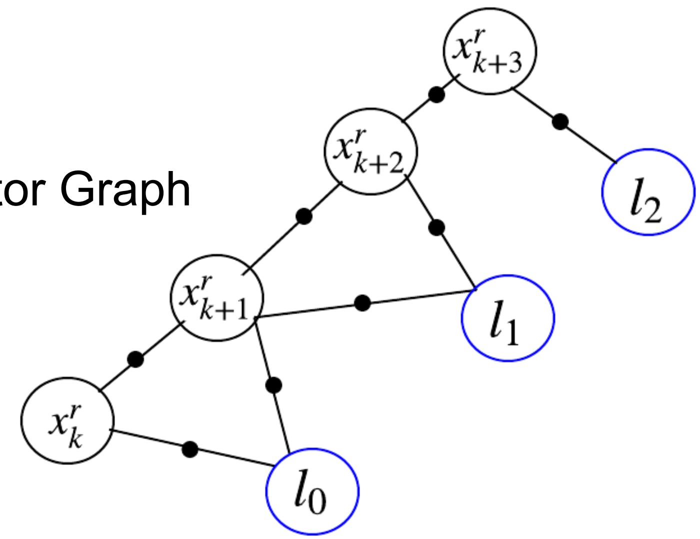
[societyofrobots.com]

Estimation ⇄ **Perception**

Introduction - SLAM

- Represent robot knowledge in a graph model
  - Vertices represent the variables. For example location of robot
  - Edges represent constrains between variables, also known as **factors**

Factor Graph

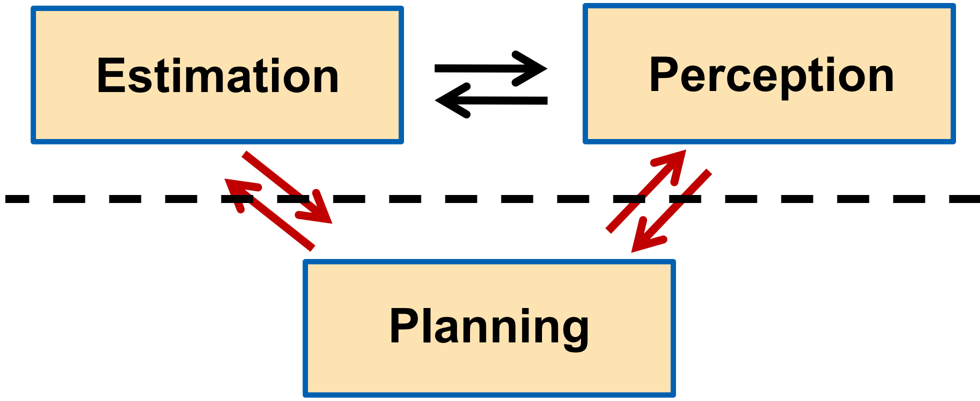


Estimation ⇄ **Perception**

- Allows computationally efficient probabilistic inference, given data
- For example, pose estimation given sensor data (e.g. images)

Introduction - Planning

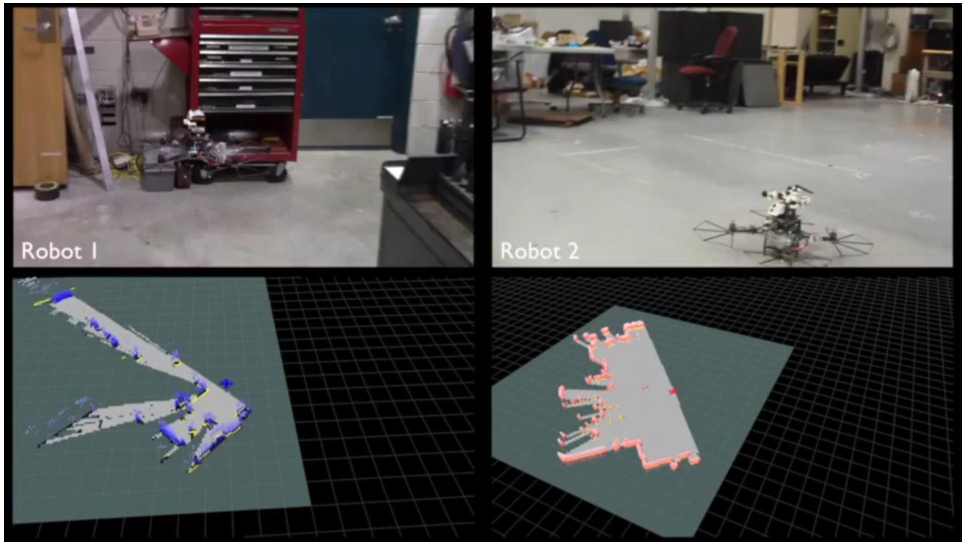
- Navigation and mapping in **unknown environment** without GPS
- Key components for autonomous operation include
  - Estimation and Perception:
    Where am I? What is the surrounding environment?
  - Planning: What to do next?



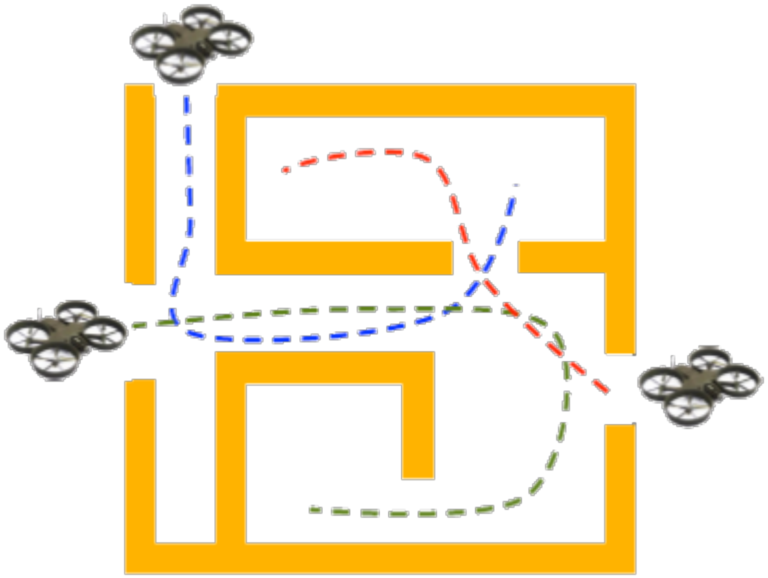
- Belief space planning (BSP) - fundamental problem in robotics

Introduction – Multi Robot SLAM

- Navigation and mapping in unknown environment with **multiple robots (MR)**
  - Robust and faster exploration/mapping
  - Higher accuracy in a multi robot collaborative framework



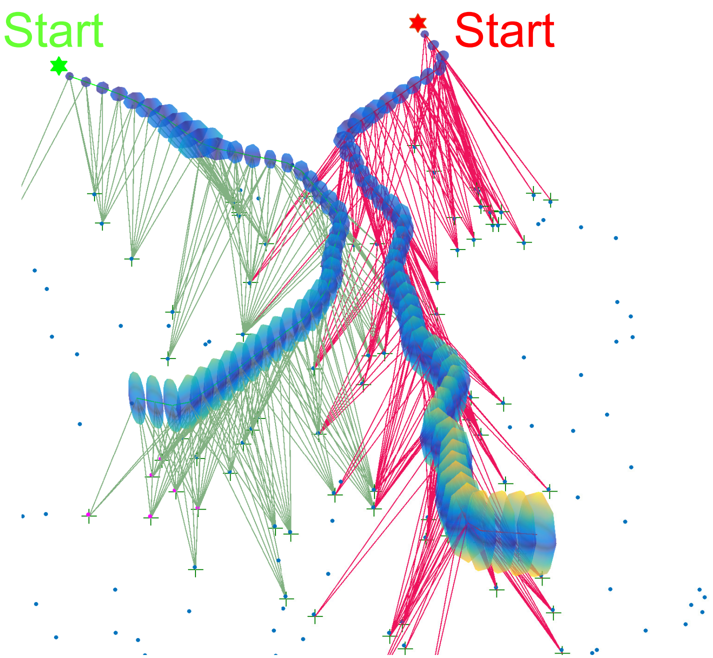
[J. Dong et al. '15 ICRA]



[Indelman '16 CSM, '14 ICRA]

Introduction - Multi Robot Belief Space Planning

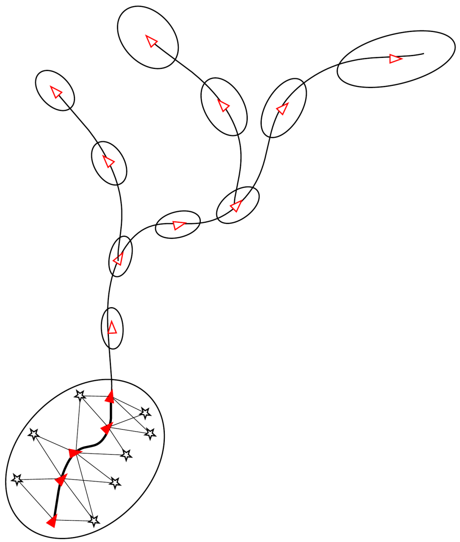
- Belief space planning in unknown environment with multiple robots (MR)
- Reason about uncertainty within planning and consider collaboration between robots



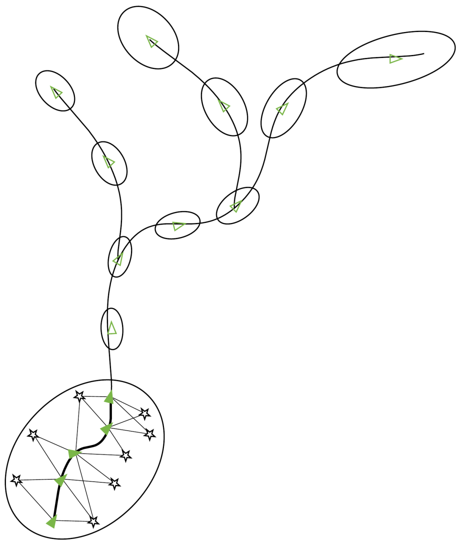
[Indelman '15 ISRR]

Related Work – Belief Space Planning (BSP)

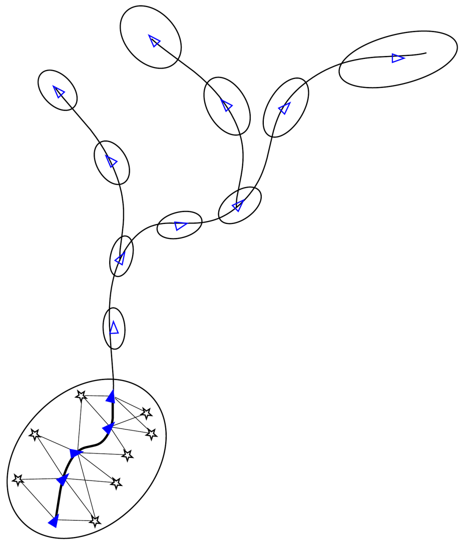
- Solving **multi-robot BSP** is in particular challenging
  - Involves considering **all combinations** of candidate paths of **all** robots
  - Existing approaches typically assume **environment/map is known**



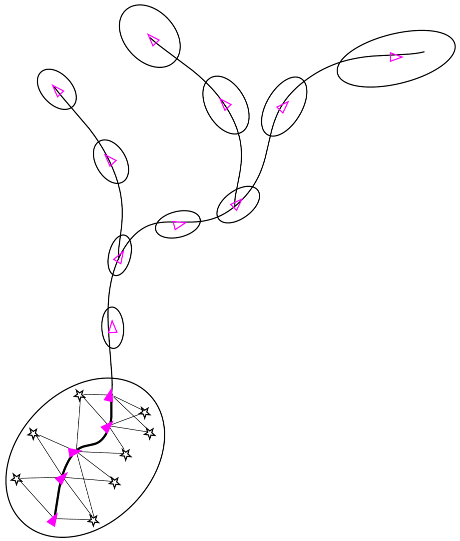
Red Robot



Green Robot



Blue Robot

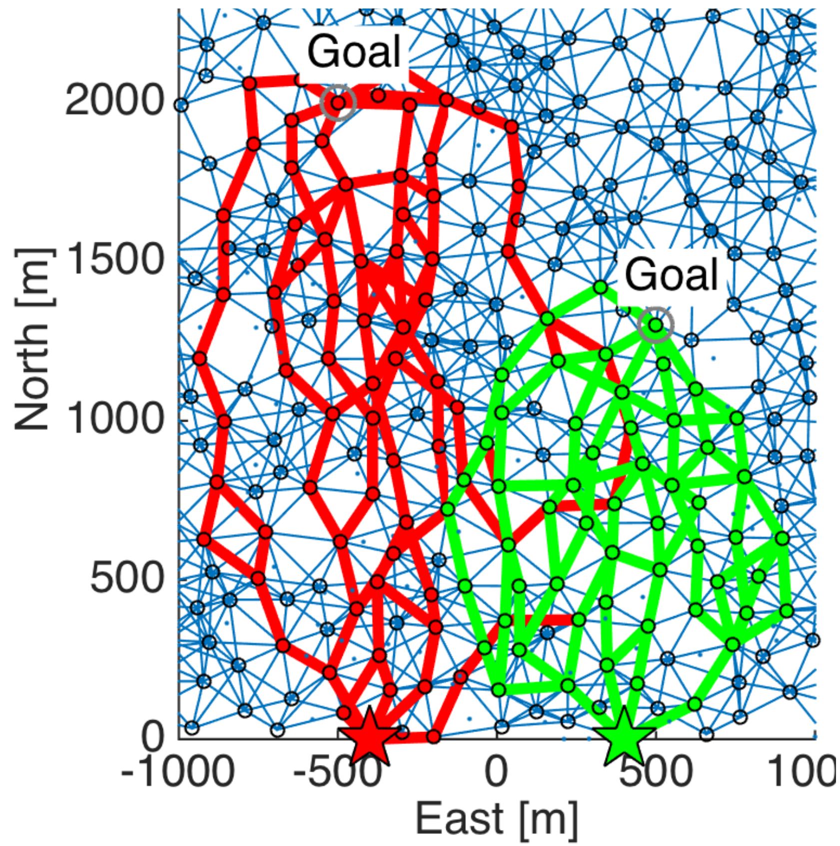


Purple Robot

Related Work – Sampling Methods

- Sampling Methods
  - Discretize the environment
  - Candidate trajectories
- Existing approaches
  - Rapidly exploring random trees (RRT)
  - Rapidly exploring random graph (RRG)
  - Probabilistic road map (PRM)

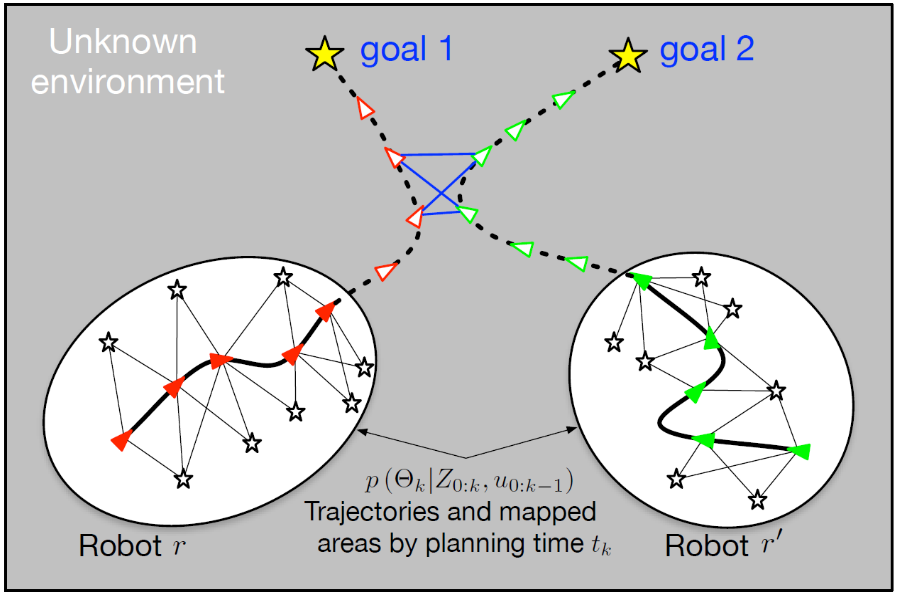
PRM



Related Work – Belief Space Planning (BSP)

– Existing approaches typically assume **environment/map is known**

- Recent works enable operation in unknown environments **[Kim et al. '14 IJRR] [Indelman et al. '15 IJRR] [Indelman '15 ISRR]**
- In particular, reason **about future observations of unknown environments within multi-robot belief space planning** **[Indelman '15 ISRR]**

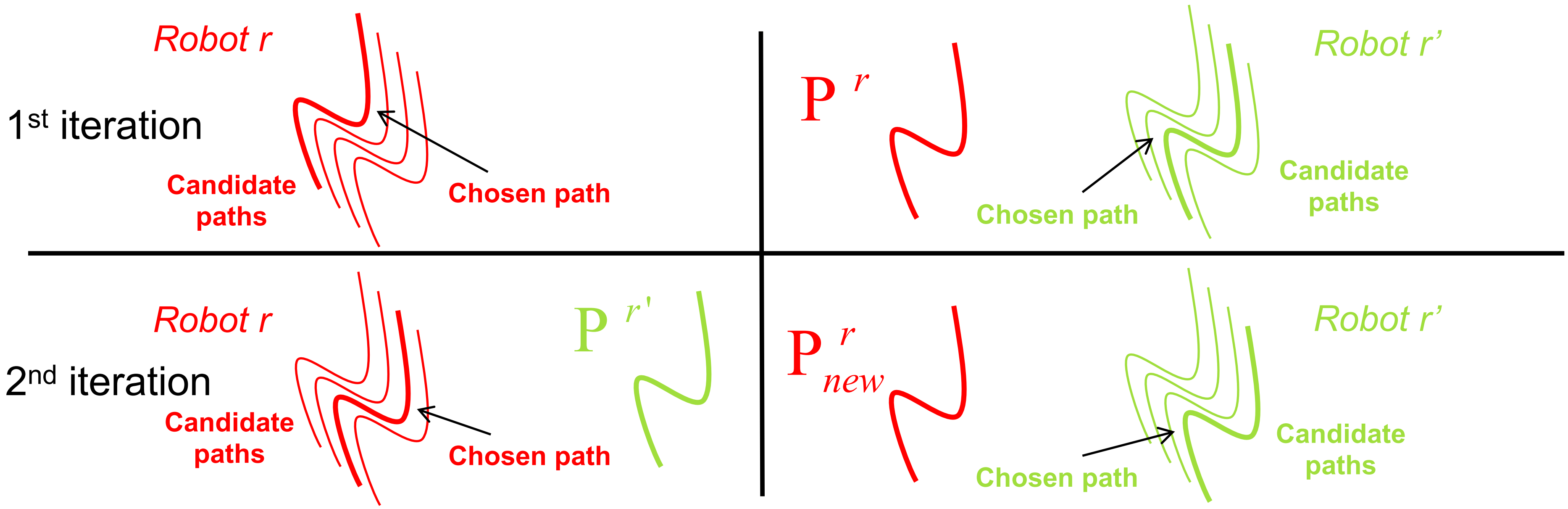


Related Work – Announced Path Approach

– Involves considering **all combinations** of candidate paths of **all** robots

- A common (sub-optimal) iterative approach to reduce computational effort: [Atanasov '14 TRO] [Levine '13 JAIS]

Calculations are performed from scratch at each iteration

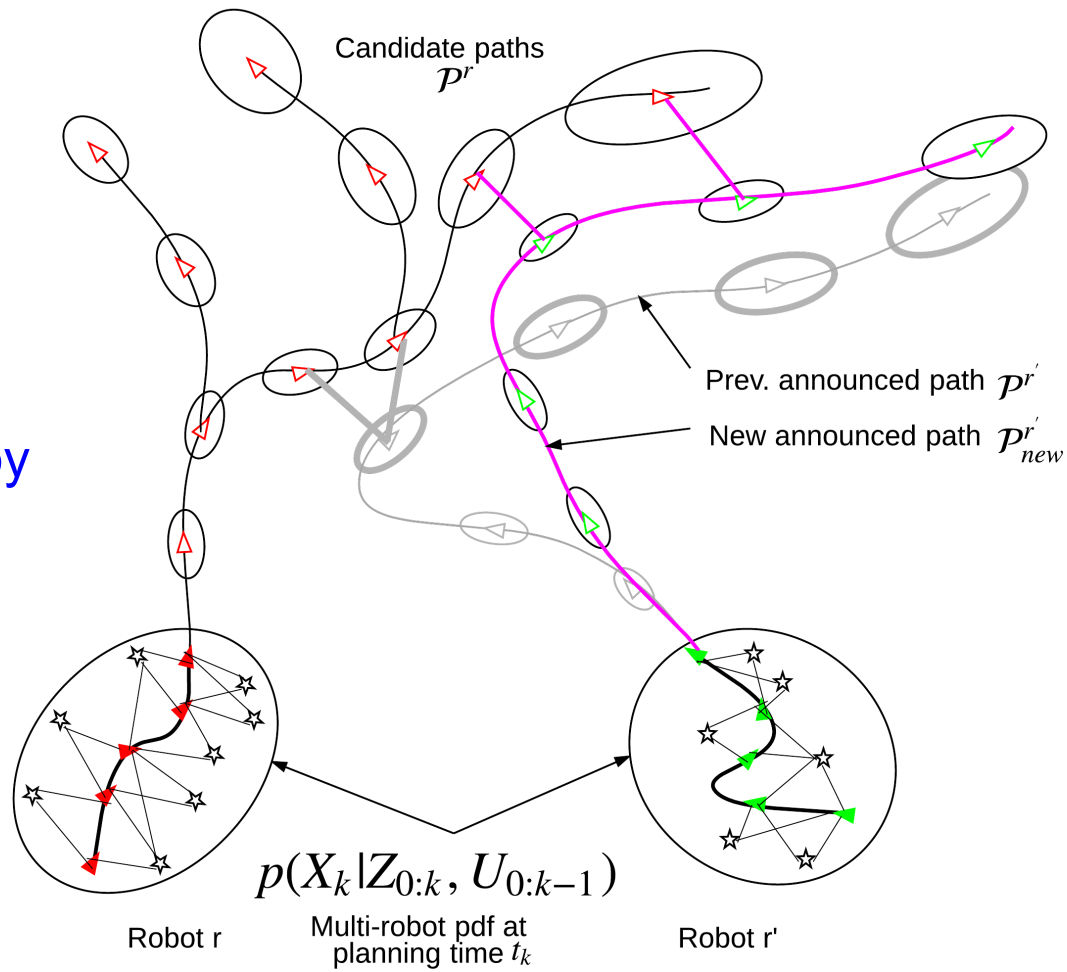


TECHNION Israel Institute of Technology

Contribution

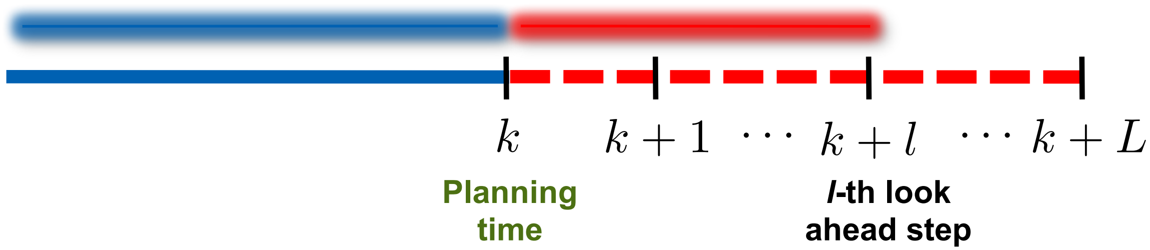
- Each time the announced path from some robot r' changes, each robot r has to re-evaluate its candidate paths

- **Key idea**
  - Not all paths are impacted due to change in the announced paths
  - Impacted paths can be efficiently re-evaluated by reusing calculations



Formulation - Multi-robot Belief Space Planning

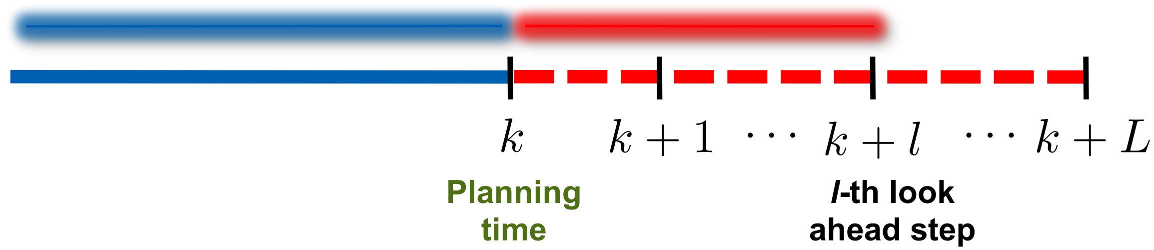
- Belief for robot r at a future time t_{k+l}: $b\left[X_{k+l}^{r}\right]$ B $p\left(X_{k+l}^{r} \mid Z_{0:k+l}^{r}, u_{0:k+l-1}^{r}\right)$



- X_{k+l}^{r} - Poses and landmarks estimated by robot r
- $Z_{0:k+l}^{r}$ - Observations available to robot r
- $u_{0:k+l-1}^{r}$ - Controls of robot r

Formulation - Multi-robot Belief Space Planning

- Belief for robot r at a future time t_{k+l}: $b\left[X^r_{k+l}\right] \triangleq p\left(X^r_{k+l} \mid Z^r_{0:k+l}, u^r_{0:k+l-1}\right)$



- Multi-robot objective function:

$$J(\mathrm{U}) = \mathrm{E}\left[\sum_{l=1}^{L}\sum_{r=1}^{R} c^r_l\left(b[X^r_{k+l}], u^r_{k+l}\right)\right]$$

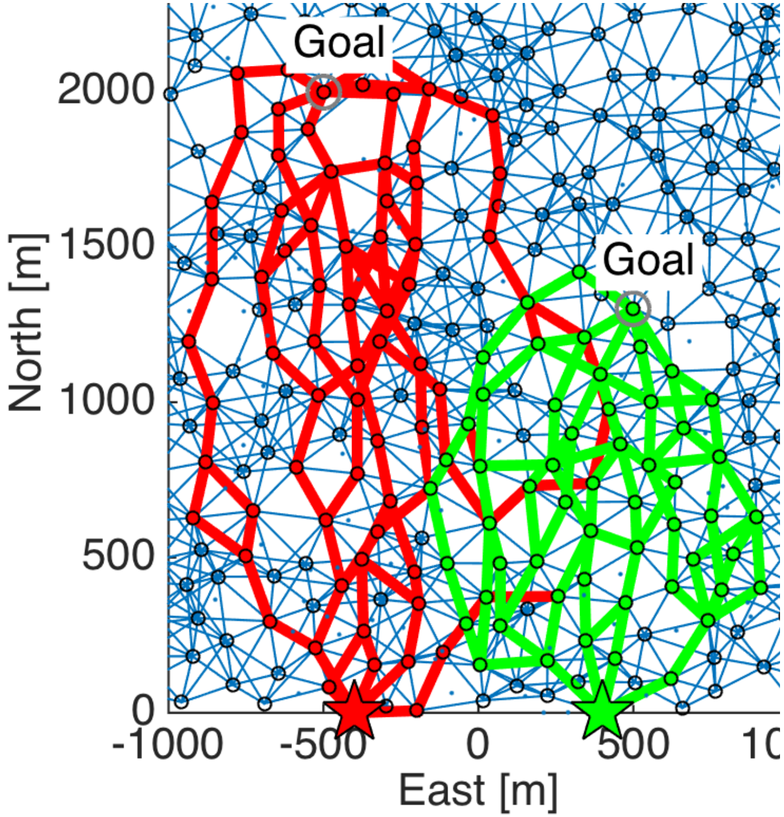
- Optimal controls for all R robots: $\mathrm{U}^{\star} = \underset{\mathrm{U}}{\operatorname{argmin}}\; J(\mathrm{U})$

Formulation - Multi-robot Belief Space Planning

- Probability distribution function (pdf) of multi robot at planning time t_k

$$b[\mathrm{P}] = p(X_k \mid Z_{0:k}, U_{0:k-1}) \prod_{r=1}^{R} \left[\prod_{l=1}^{L(\mathrm{P}^r)} p(x_{v_l}^r \mid x_{v_{l-1}}^r, u_{v_{l-1}}^r) \cdot p(Z_{v_l}^r \mid X_{k+l}^r) \prod_{\{i,j\}} p(z_{i,j}^{r,r'} \mid x_{v_i}^r, x_{v_j}^{r'}) \right]$$

- P - Some specific candidate paths for all robots $\mathrm{P} = \left[\mathrm{P}^r, \mathrm{P}^{r'}, \ldots \right]$



- Maximum a posteriori (MAP) inference:

$$b[\mathrm{P}] = N\left(\hat{X}(\mathrm{P}), \Lambda^{-1}(\mathrm{P}) \right)$$

- State transition and observation models:

$$x_{i+1}^r = f(x_i^r, u_i^r, w_i^r), \quad z_{i,j}^r = h(x_i^r, x_j^r, v_{i,j}^r)$$

Formulation - Multi-robot Belief Space Planning

- Probability distribution function (pdf) of multi robot at planning time t_k:

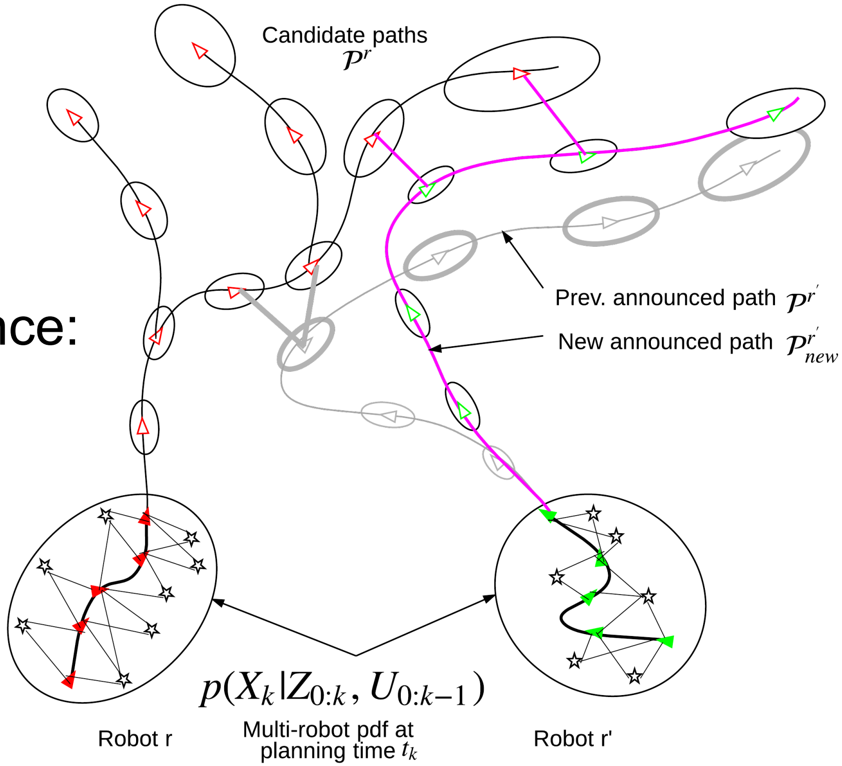
$$b[\mathrm{P}] = p(X_k \mid \mathrm{Z}_{0:k}, \mathrm{U}_{0:k-1}) \prod_{r=1}^{R} \left[\prod_{l=1}^{L(\mathrm{P}^r)} p(x^r_{v_l} \mid x^r_{v_{l-1}}, u^r_{v_{l-1}}) \cdot p(Z^r_{v_l} \mid X^r_{k+l}) \prod_{\{i,j\}} p(z^{r,r'}_{i,j} \mid x^r_{v_i}, x^{r'}_{v_j}) \right]$$

Prior

$$\Lambda(\mathrm{P}) = \Lambda_k + \sum_{r=1}^{R} \left[\sum_{l=1}^{L(\mathrm{P}^r)} \Lambda^{r,local}_l + \sum_{\{i,j\}} \Lambda^{r,r'}_{i,j} \right]$$

- Maximum a posteriori (MAP) inference:

$$b[\mathrm{P}] = N\left(\hat{X}(\mathrm{P}), \Lambda^{-1}(\mathrm{P})\right)$$



Formulation - Multi-robot Belief Space Planning

- Probability distribution function (pdf) of multi robot at planning time t_k:

$$b[\mathrm{P}] = p(X_k \mid \mathrm{Z}_{0:k}, \mathrm{U}_{0:k-1}) \prod_{r=1}^{R} \left[\prod_{l=1}^{L(\mathrm{P}^r)} p(x_{v_l}^r \mid x_{v_{l-1}}^r, u_{v_{l-1}}^r) \cdot p(Z_{v_l}^r \mid X_{k+l}^r) \prod_{\{i,j\}} p(z_{i,j}^{r,r'} \mid x_{v_i}^r, x_{v_j}^{r'}) \right]$$

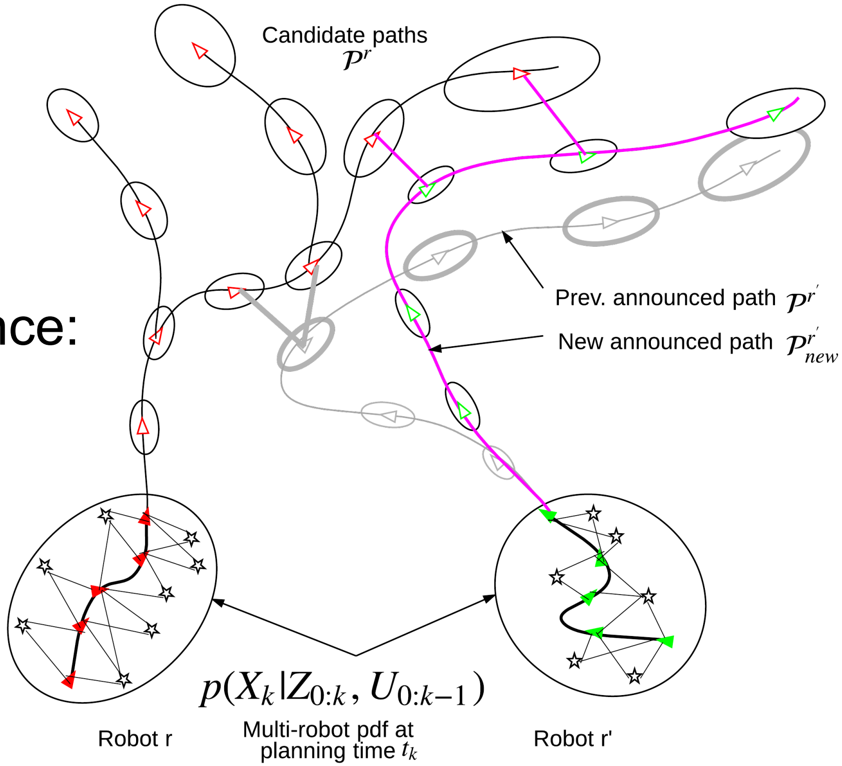
Prior

Local information

$$\Lambda(\mathrm{P}) = \Lambda_k + \sum_{r=1}^{R} \left[\sum_{l=1}^{L(\mathrm{P}^r)} \Lambda_l^{r,local} + \sum_{\{i,j\}} \Lambda_{i,j}^{r,r'} \right]$$

- Maximum a posteriori (MAP) inference:

$$b[\mathrm{P}] = N\left(\hat{X}(\mathrm{P}), \Lambda^{-1}(\mathrm{P})\right)$$



Formulation - Multi-robot Belief Space Planning

- Probability distribution function (pdf) of multi robot at planning time t_k:

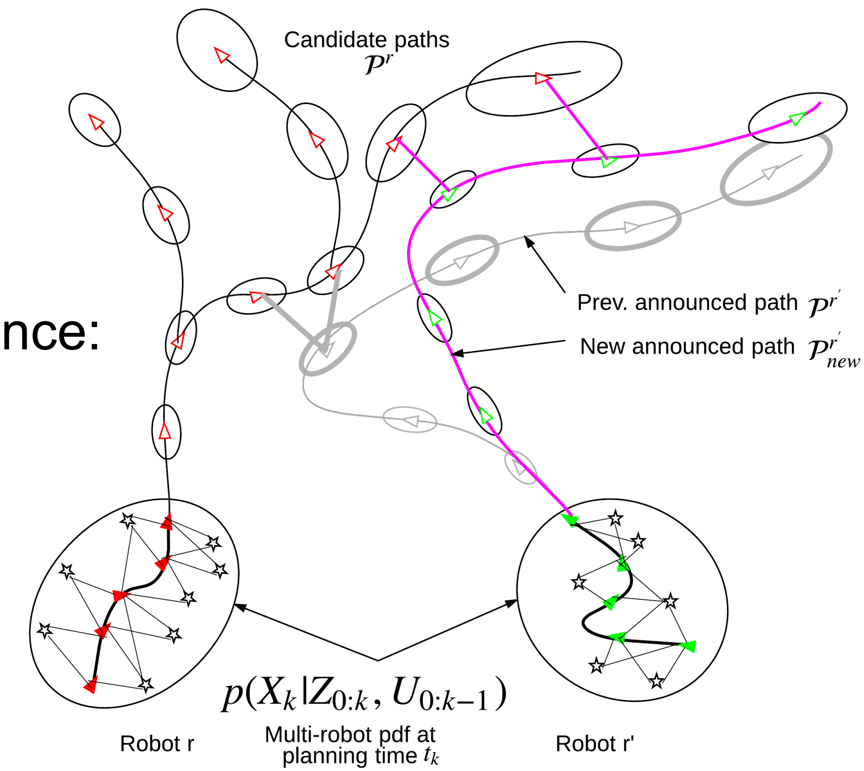
$$b[\mathrm{P}] = p(X_k \mid \mathrm{Z}_{0:k}, \mathrm{U}_{0:k-1}) \prod_{r=1}^{R} \left[\prod_{l=1}^{L(\mathrm{P}^r)} p(x^r_{v_l} \mid x^r_{v_{l-1}}, u^r_{v_{l-1}}) \cdot p(Z^r_{v_l} \mid X^r_{k+l}) {\color{red}\prod_{\{i,j\}} p(z^{r,r'}_{i,j} \mid x^r_{v_i}, x^{r'}_{v_j})} \right]$$

Prior — Local information — Mutual observations

$$\Lambda(\mathrm{P}) = \Lambda_k + \sum_{r=1}^{R} \left[\sum_{l=1}^{L(\mathrm{P}^r)} \Lambda_l^{r,local} + {\color{red}\sum_{\{i,j\}} \Lambda^{r,r'}_{i,j}} \right]$$

- Maximum a posteriori (MAP) inference:

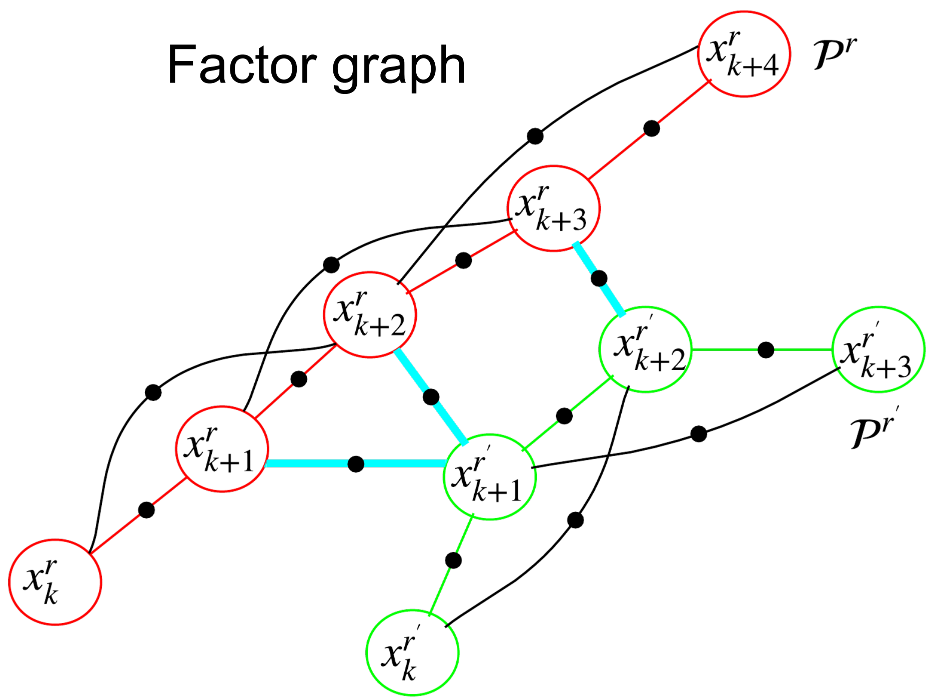
$$b[\mathrm{P}] = N\left(\hat{X}(\mathrm{P}), \Lambda^{-1}(\mathrm{P})\right)$$

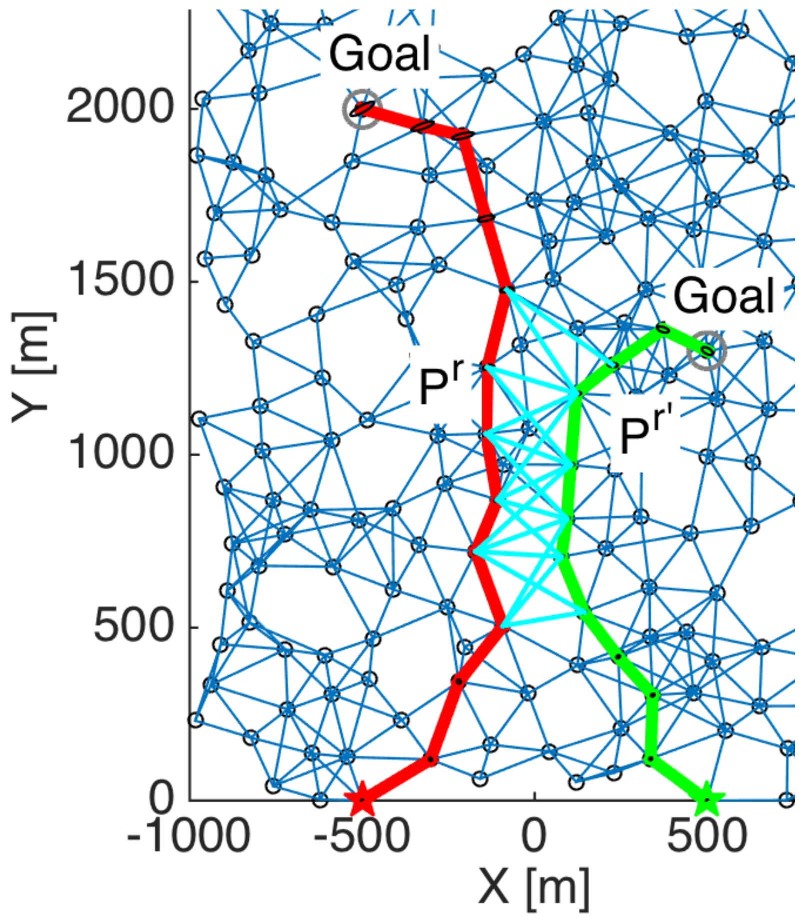


Formulation - Multi-robot Belief Space Planning

- Probability distribution function (pdf) of multi robot at planning time t_k:

$$b[\mathrm{P}] = p(X_k \mid \mathrm{Z}_{0:k}, \mathrm{U}_{0:k-1}) \prod_{r=1}^{R} \left[\prod_{l=1}^{L(\mathrm{P}^r)} p(x_{v_l}^r \mid x_{v_{l-1}}^r, u_{v_{l-1}}^r) \cdot p(Z_{v_l}^r \mid X_{k+l}^r) \prod_{\{i,j\}} p(z_{i,j}^{r,r'} \mid x_{v_i}^r, x_{v_j}^{r'}) \right]$$



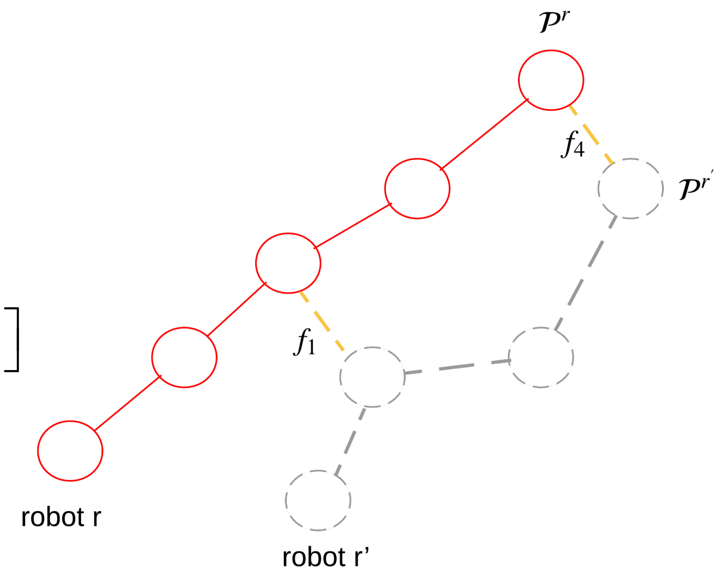


Factor Graph Inference

- Joint state $\mathrm{P}^{r}, \mathrm{P}^{r'}$:

$$b[\mathrm{P}^{r}, \mathrm{P}^{r'}] = p(X_k \mid \mathrm{Z}_{0:k}, \mathrm{U}_{0:k-1}) \prod_{l=1}^{L(\mathrm{P}^{r})} \left[p(x^{r}_{v_l} \mid x^{r}_{v_{l-1}}, u^{r}_{v_{l-1}}) \cdot p(Z^{r}_{v_l} \mid X^{r}_{k+l}) \right]$$

$$\prod_{l=1}^{L(\mathrm{P}^{r'})} \left[p(x^{r'}_{v_l} \mid x^{r'}_{v_{l-1}}, u^{r'}_{v_{l-1}}) \cdot p(Z^{r'}_{v_l} \mid X^{r'}_{k+l}) \right] \prod_{r=1}^{R} \left[\prod_{\{i,j\}} p(z^{r,r'}_{i,j} \mid x^{r}_{v_i}, x^{r'}_{v_j}) \right]$$



- Joint state $\mathrm{P}^{r}, \mathrm{P}^{r'}_{new}$:

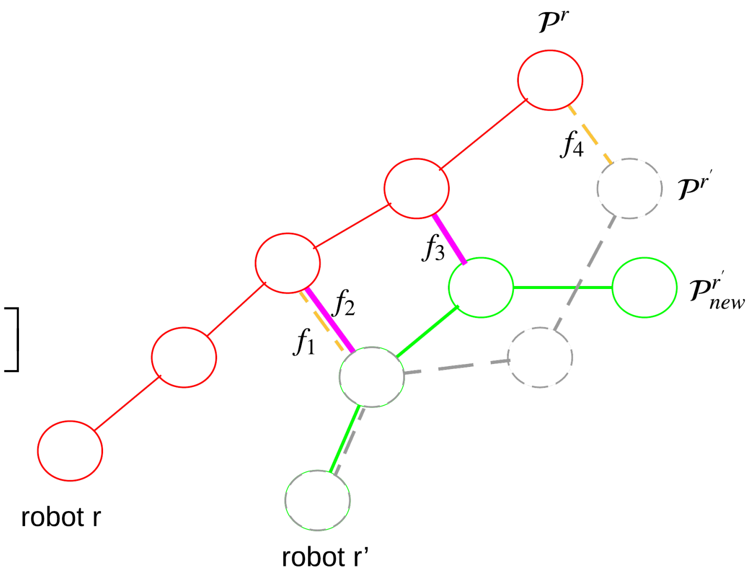
TECHNION Israel Institute of Technology

Factor Graph Inference

- Joint state $\mathrm{P}^{r}, \mathrm{P}^{r'}$:

$$b[\mathrm{P}^{r}, \mathrm{P}^{r'}] = p(X_k \mid \mathrm{Z}_{0:k}, \mathrm{U}_{0:k-1}) \prod_{l=1}^{L(\mathrm{P}^{r})} \left[p(x^{r}_{v_l} \mid x^{r}_{v_{l-1}}, u^{r}_{v_{l-1}}) \cdot p(Z^{r}_{v_l} \mid X^{r}_{k+l}) \right]$$

$$\prod_{l=1}^{L(\mathrm{P}^{r'})} \left[p(x^{r'}_{v_l} \mid x^{r'}_{v_{l-1}}, u^{r'}_{v_{l-1}}) \cdot p(Z^{r'}_{v_l} \mid X^{r'}_{k+l}) \right] \prod_{r=1}^{R} \left[\prod_{\{i,j\}} p(z^{r,r'}_{i,j} \mid x^{r}_{v_i}, x^{r'}_{v_j}) \right]$$



- Joint state $\mathrm{P}^{r}, \mathrm{P}^{r'}_{new}$:

Does not change

$$b[\mathrm{P}^{r}, \mathrm{P}^{r'}_{new}] = p(X_k \mid \mathrm{Z}_{0:k}, \mathrm{U}_{0:k-1}) \prod_{l=1}^{L(\mathrm{P}^{r})} \left[p(x^{r}_{v_l} \mid x^{r}_{v_{l-1}}, u^{r}_{v_{l-1}}) \cdot p(Z^{r}_{v_l} \mid X^{r}_{k+l}) \right]$$

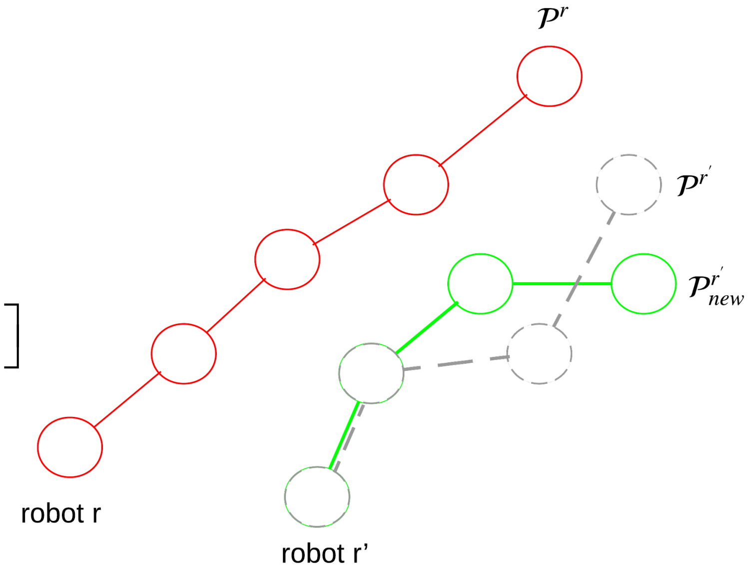
$$\prod_{l=1}^{L(\mathrm{P}^{r'})} \left[p(x^{r'}_{v_l} \mid x^{r'}_{v_{l-1}}, u^{r'}_{v_{l-1}}) \cdot p(Z^{r'}_{v_l} \mid X^{r'}_{k+l}) \right] \prod_{r=1}^{R} \left[\prod_{\{i,j\}} p(z^{r,r'}_{i,j} \mid x^{r}_{v_i}, x^{r'}_{v_j}) \right]$$

Factor Graph Inference

- Joint state $\mathrm{P}^{r}, \mathrm{P}^{r'}$:

$$b[\mathrm{P}^{r}, \mathrm{P}^{r'}] = p(X_k \mid \mathrm{Z}_{0:k}, \mathrm{U}_{0:k-1}) \prod_{l=1}^{L(\mathrm{P}^{r})} \left[p(x^{r}_{v_l} \mid x^{r}_{v_{l-1}}, u^{r}_{v_{l-1}}) \cdot p(Z^{r}_{v_l} \mid X^{r}_{k+l}) \right]$$

$$\prod_{l=1}^{L(\mathrm{P}^{r'})} \left[p(x^{r'}_{v_l} \mid x^{r'}_{v_{l-1}}, u^{r'}_{v_{l-1}}) \cdot p(Z^{r'}_{v_l} \mid X^{r'}_{k+l}) \right] \prod_{r=1}^{R} \left[\prod_{\{i,j\}} p(z^{r,r'}_{i,j} \mid x^{r}_{v_i}, x^{r'}_{v_j}) \right]$$



- Joint state $\mathrm{P}^{r}, \mathrm{P}^{r'}_{new}$:

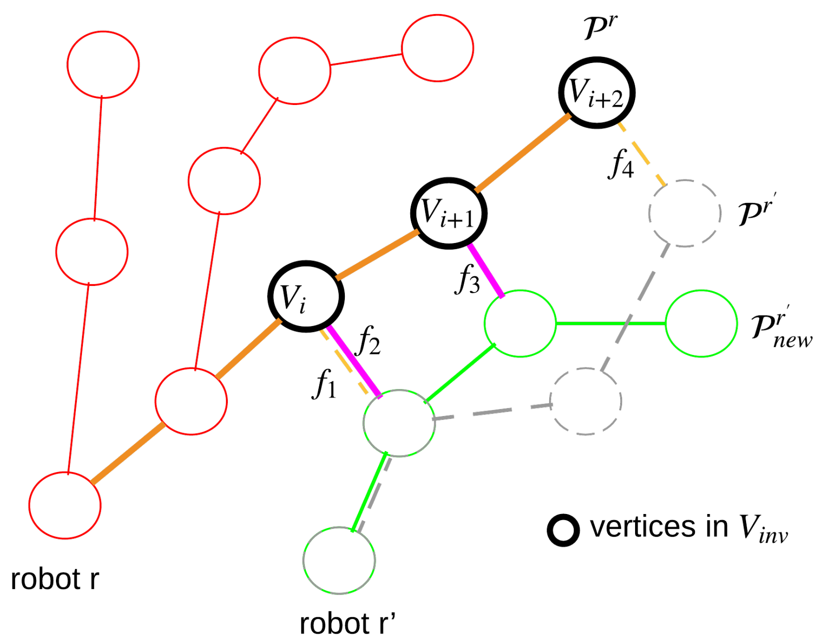
Does not change

$$b[\mathrm{P}^{r}, \mathrm{P}^{r'}_{new}] = p(X_k \mid \mathrm{Z}_{0:k}, \mathrm{U}_{0:k-1}) \prod_{l=1}^{L(\mathrm{P}^{r})} \left[p(x^{r}_{v_l} \mid x^{r}_{v_{l-1}}, u^{r}_{v_{l-1}}) \cdot p(Z^{r}_{v_l} \mid X^{r}_{k+l}) \right]$$

$$\prod_{l=1}^{L(\mathrm{P}^{r'})} \left[p(x^{r'}_{v_l} \mid x^{r'}_{v_{l-1}}, u^{r'}_{v_{l-1}}) \cdot p(Z^{r'}_{v_l} \mid X^{r'}_{k+l}) \right]$$

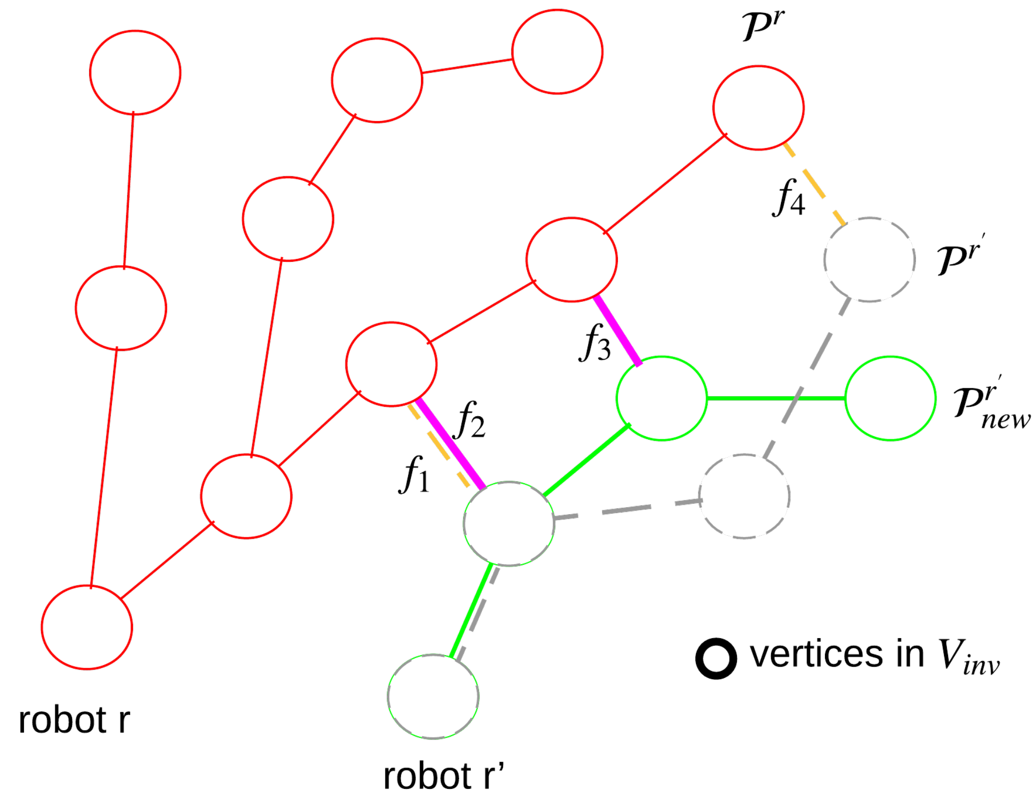
Algorithm Overview – High Level

- Key Idea
  - Not all paths are impacted due to change in the announced paths
  - Impacted paths can be efficiently re-evaluated by reusing calculations
- Iterate **over vertices** in previous and new announced paths
- Identify **involved** vertices in multi-robot factors – collect into set V_{inv}
- Identify and **mark** involved paths from all candidate paths



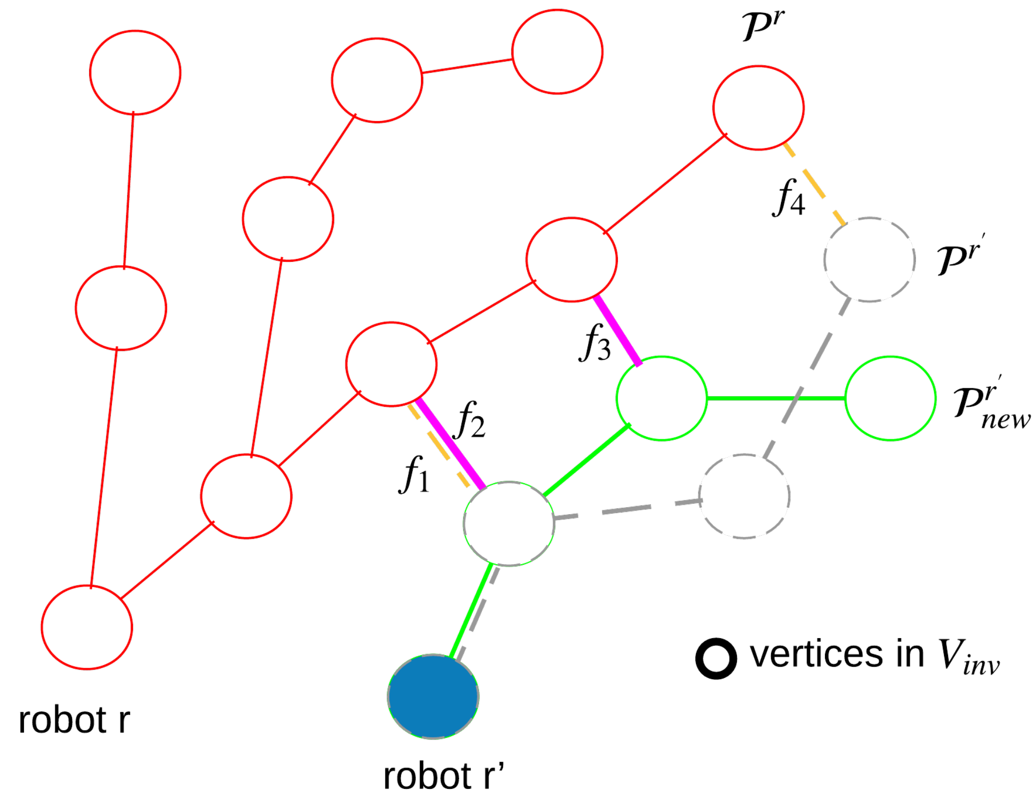
Algorithm Overview – Details

- Key Idea
  - Iterate vertices $v^{r'}$ that belong to $\mathcal{P}^{r'}$ or $\mathcal{P}^{r'}_{new}$
    - Find all nearby vertices $\{v^r\} \subseteq V^r$ to $v^{r'}$
    - Add v_i to V_{inv}
  - Identify multi-robot factors to be added or removed
  - Mark all candidate paths that go through vertex



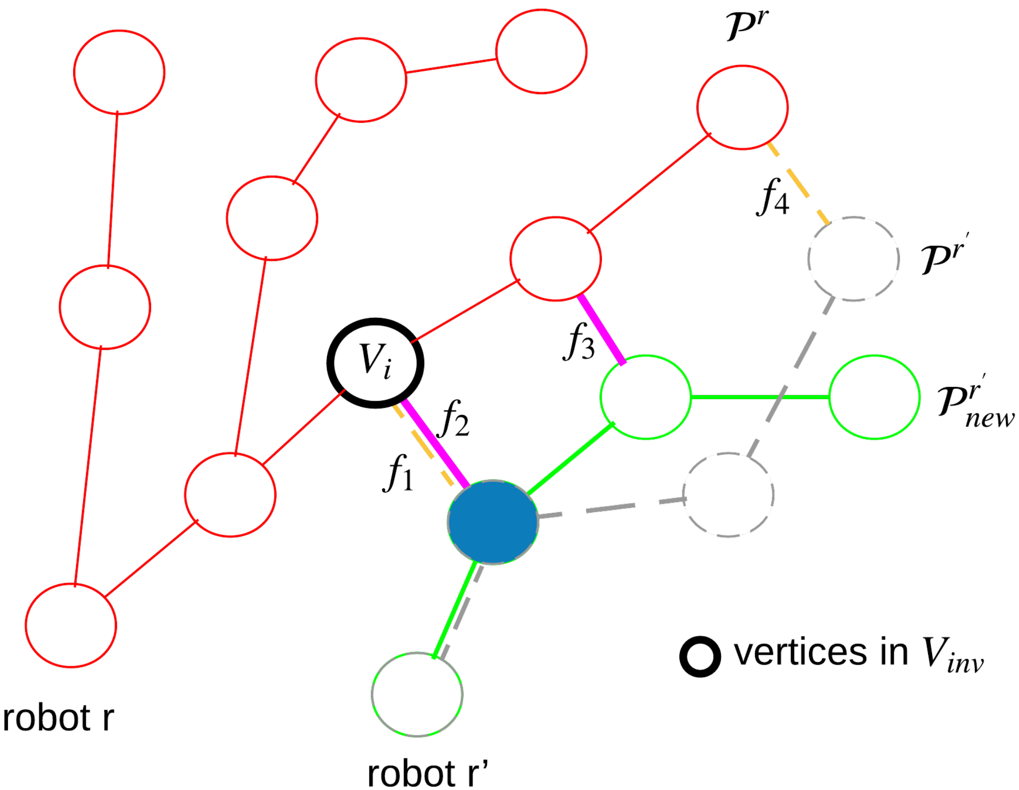
Algorithm Overview – Details

- Key Idea
  - Iterate vertices $v^{r'}$ that belong to $\mathcal{P}^{r'}$ or $\mathcal{P}^{r'}_{new}$
    - Find all nearby vertices $\{v^r\} \subseteq V^r$ to $v^{r'}$
    - Add v_i to V_{inv}
  - Identify multi-robot factors to be added or removed
  - Mark all candidate paths that go through vertex



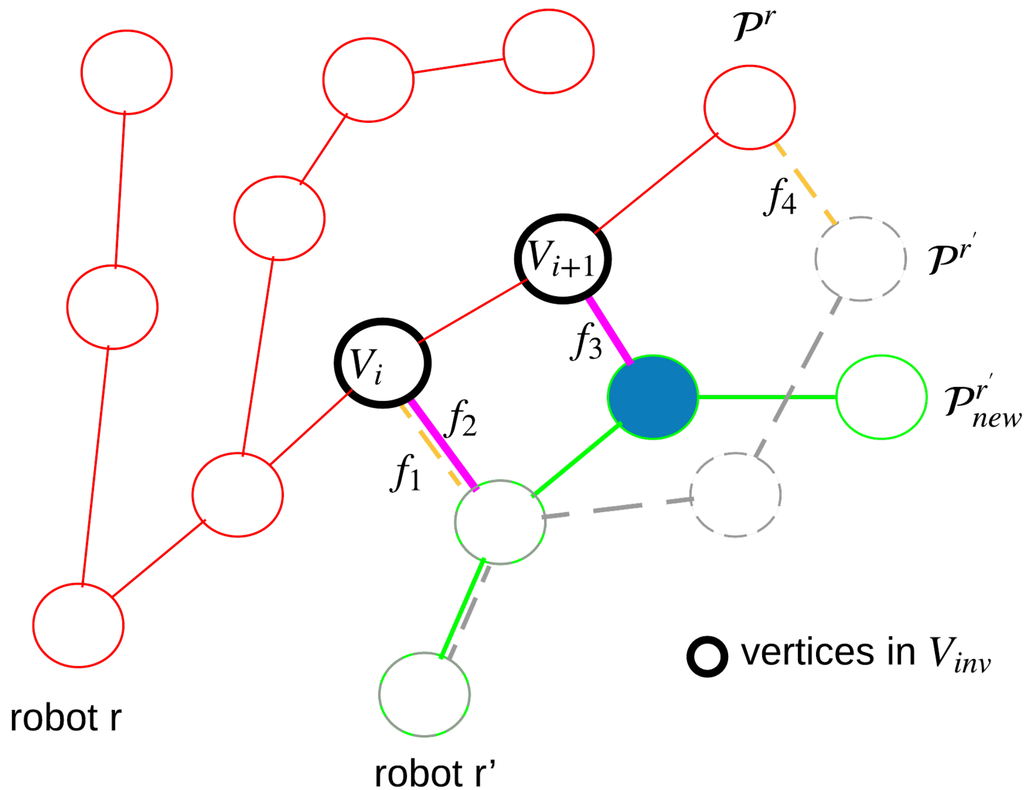
Algorithm Overview – Details

- Key Idea
  - Iterate vertices $v^{r'}$ that belong to $\mathcal{P}^{r'}$ or $\mathcal{P}^{r'}_{new}$
    - Find all nearby vertices $\{v^r\} \subseteq V^r$ to $v^{r'}$
    - Add v_i to V_{inv}
  - Identify multi-robot factors to be added or removed
  - Mark all candidate paths that go through vertex



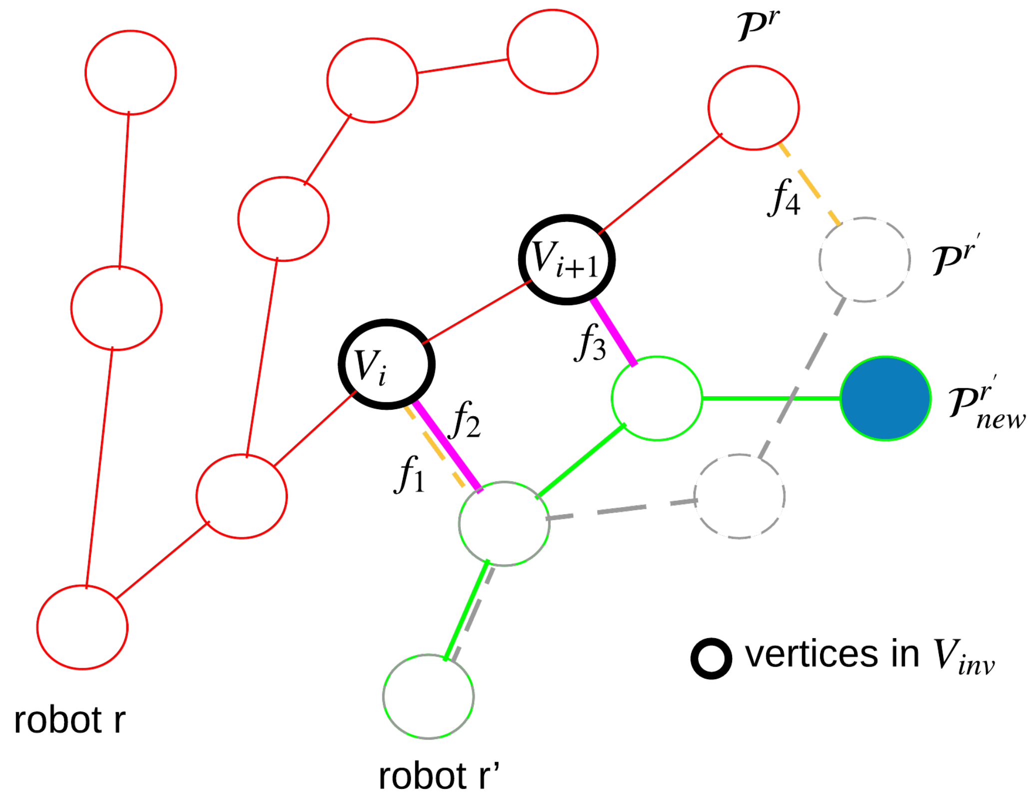
Algorithm Overview – Details

- Key Idea
  - Iterate vertices $v^{r'}$ that belong to $\mathcal{P}^{r'}$ or $\mathcal{P}^{r'}_{new}$
    - Find all nearby vertices $\{v^r\} \subseteq V^r$ to $v^{r'}$
    - Add v_i to V_{inv}
  - Identify multi-robot factors to be added or removed
  - Mark all candidate paths that go through vertex



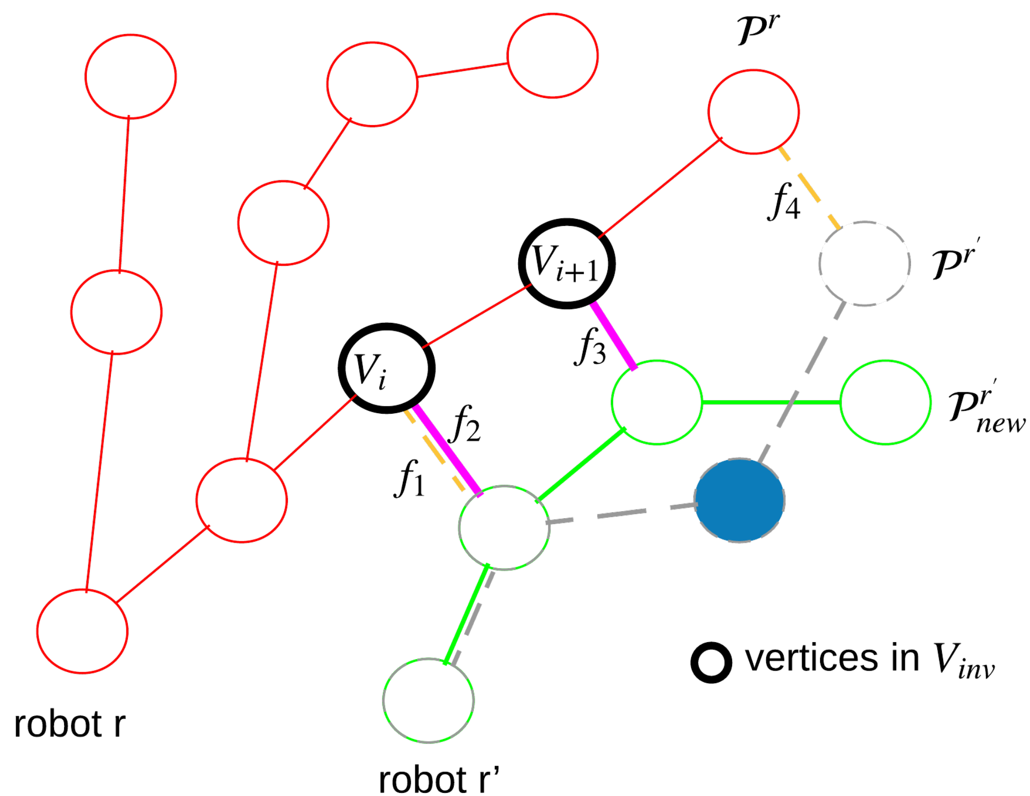
Algorithm Overview – Details

- Key Idea
  - Iterate vertices $v^{r'}$ that belong to $\mathcal{P}^{r'}$ or $\mathcal{P}^{r'}_{new}$
    - Find all nearby vertices $\{v^r\} \subseteq V^r$ to $v^{r'}$
    - Add v_i to V_{inv}
  - Identify multi-robot factors to be added or removed
  - Mark all candidate paths that go through vertex



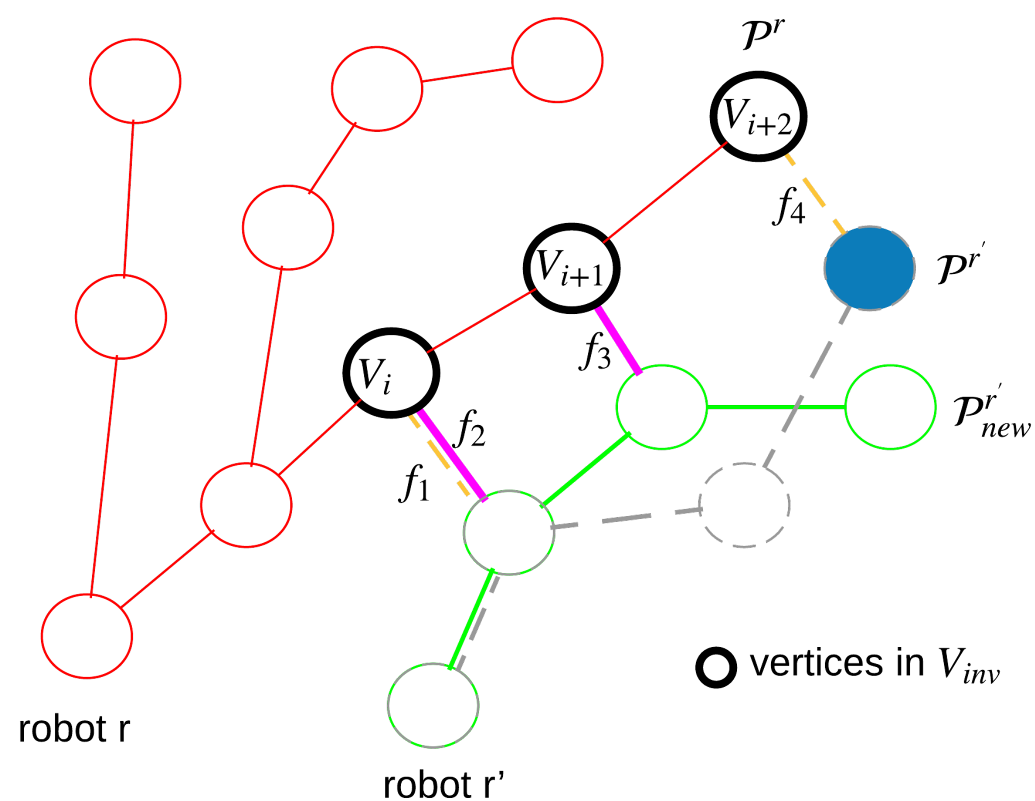
Algorithm Overview – Details

- Key Idea
  - Iterate vertices $v^{r'}$ that belong to $\mathcal{P}^{r'}$ or $\mathcal{P}^{r'}_{new}$
    - Find all nearby vertices $\{v^r\} \subseteq V^r$ to $v^{r'}$
    - Add v_i to V_{inv}
  - Identify multi-robot factors to be added or removed
  - Mark all candidate paths that go through vertex



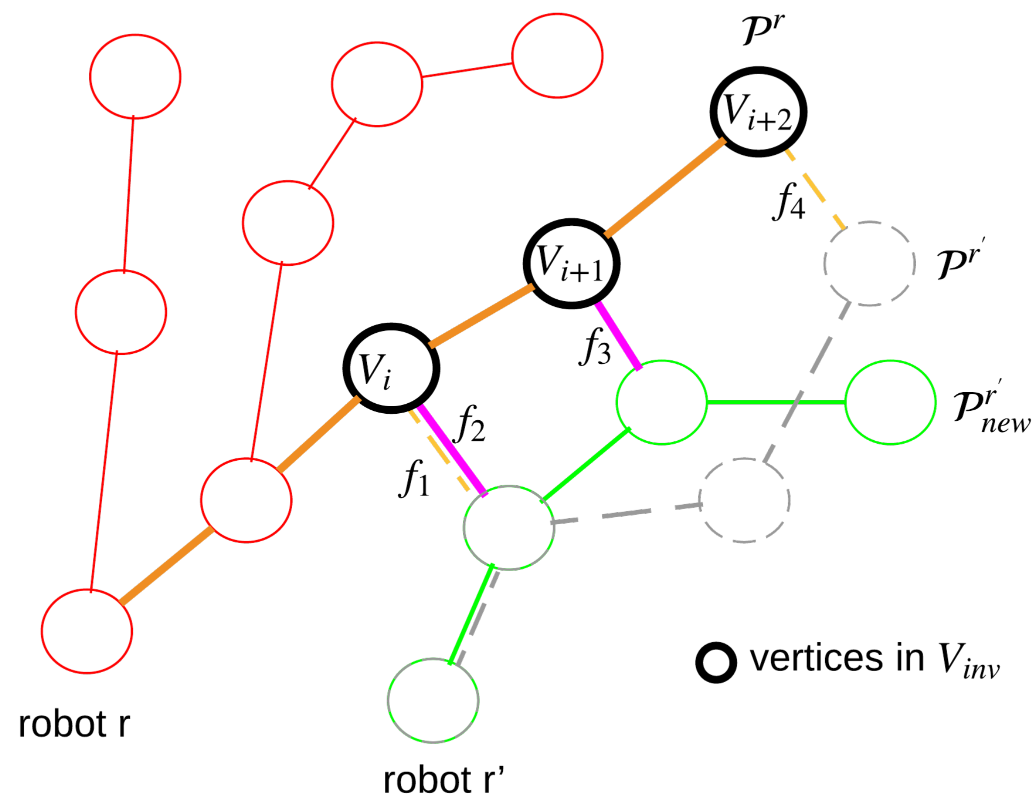
Algorithm Overview – Details

- Key Idea
  - Iterate vertices $v^{r'}$ that belong to $\mathcal{P}^{r'}$ or $\mathcal{P}^{r'}_{new}$
    - Find all nearby vertices $\{v^r\} \subseteq V^r$ to $v^{r'}$
    - Add v_i to V_{inv}
  - Identify multi-robot factors to be added or removed
  - Mark all candidate paths that go through vertex



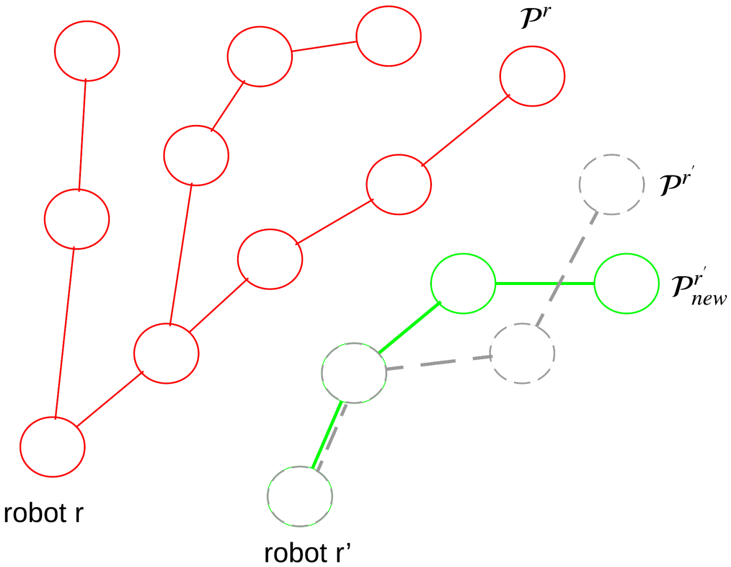
Algorithm Overview – Details

- Key Idea
  - Iterate vertices $v^{r'}$ that belong to $\mathcal{P}^{r'}$ or $\mathcal{P}^{r'}_{new}$
    - Find all nearby vertices $\{v^r\} \subseteq V^r$ to $v^{r'}$
    - Add v_i to V_{inv}
  - Identify multi-robot factors to be added or removed
  - Mark all candidate paths that go through vertex



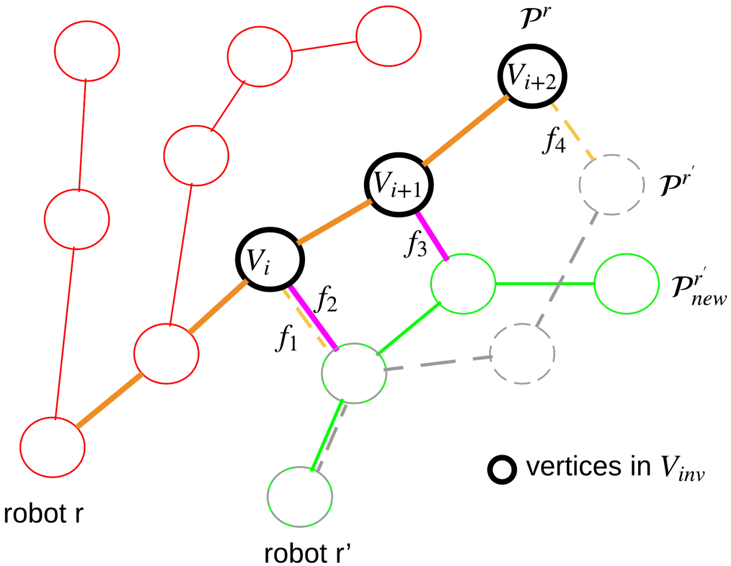
Algorithm Overview

- **Non-marked** paths
  - Calculate once the change $\Delta J^{r'}$ B $\sum_{l=1}^{L} \Delta c_l^{r'}$
  - Add $\Delta J^{r'}$ to old objective function



Algorithm Overview

- **Non-marked** paths
  - Calculate once the change $\Delta J^{r'}$ B $\sum_{l=1}^{L} \Delta c_l^{r'}$
  - Add $\Delta J^{r'}$ to old objective function



- **Marked** paths
  - Iterate over vertices in V_{inv}, add or remove multi-robot factors
  - Evaluate objective function from new Information matrix

Our method

$$\Lambda'_{k+l} = \Lambda_{k+l} - \sum_{\substack{f \in FG \\ f \notin FG' \\ f.t \le t_{k+l}}} \Lambda(f) + \sum_{\substack{f \in FG' \\ f \notin FG \\ f.t \le t_{k+l}}} \Lambda(f)$$

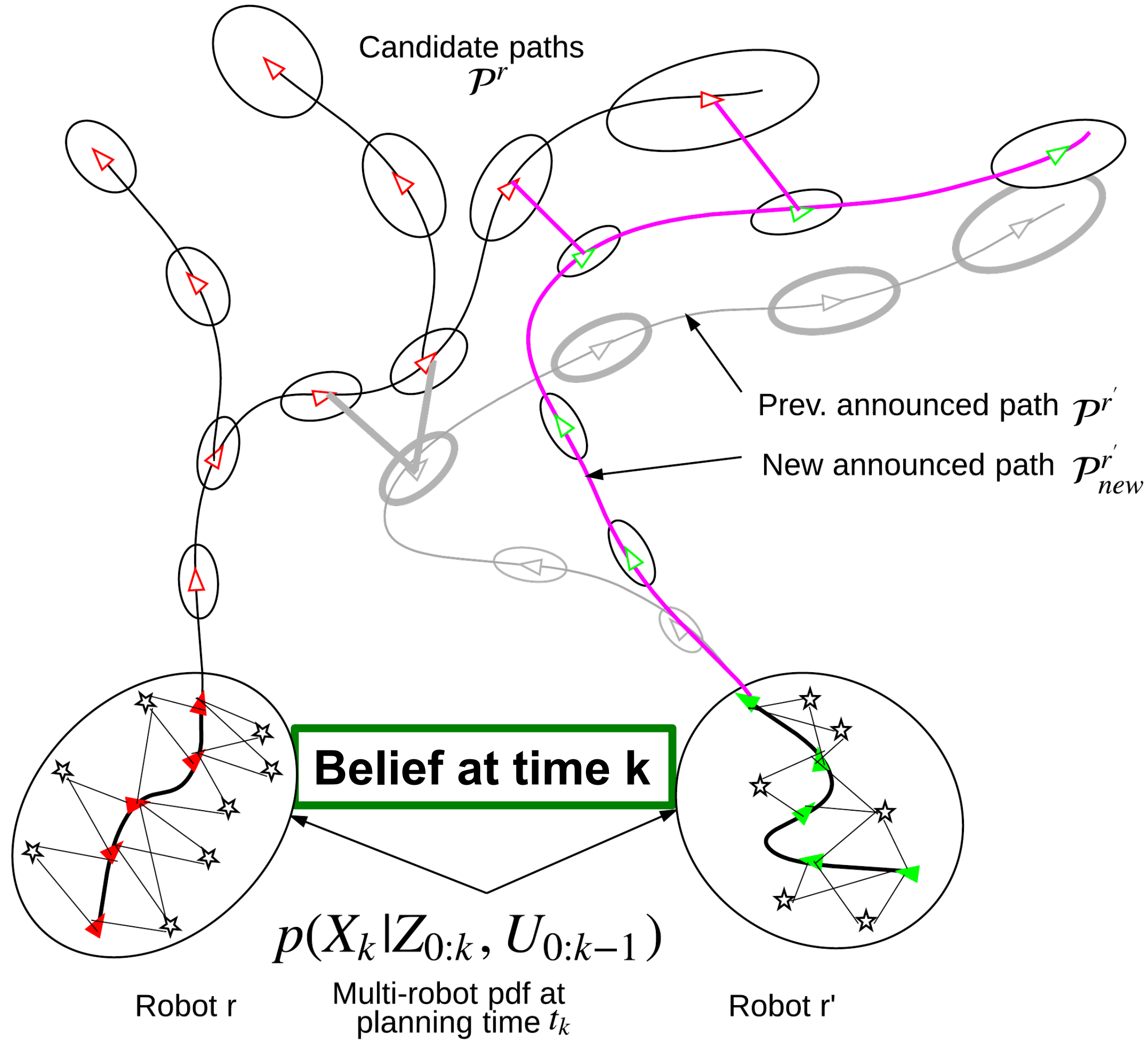
Standard method

$$\Lambda(\mathrm{P}) = \Lambda_k + \sum_{r=1}^{R} \left[\sum_{l=1}^{L(\mathrm{P}^r)} \Lambda_l^{r,local} + \sum_{\{i,j\}} \Lambda_{i,j}^{r,r'} \right]$$

From previous step!

Algorithm Overview – Prior Correlation

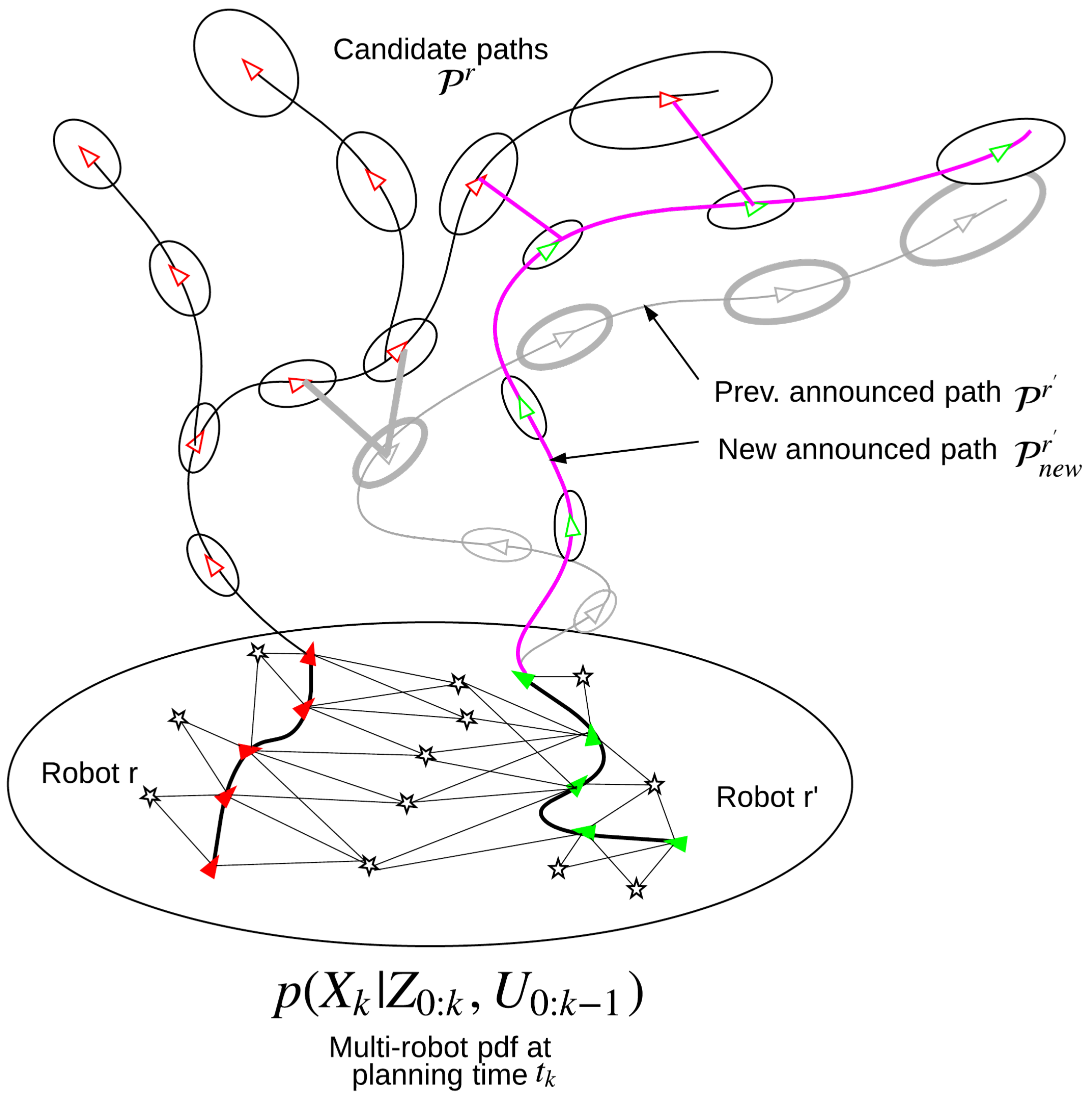
- No prior correlation at time k



Algorithm Overview – Prior Correlation

- With prior correlation at time k

Belief at time k

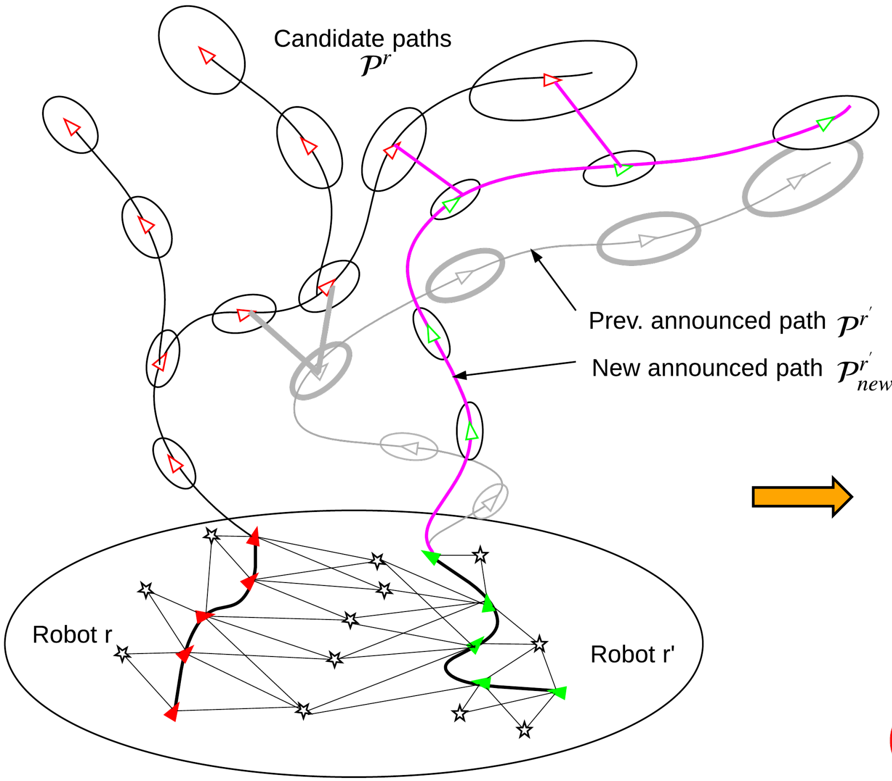


Algorithm Overview – Prior Correlation

- So far – robots' beliefs at time k were assumed to be not correlated
- In practice, correlation may exist, e.g.:
  - Robots have observed a mutual scene (or landmarks)
  - Robots made a direct observation of each other
- Our approach handles these cases as well (next slides)

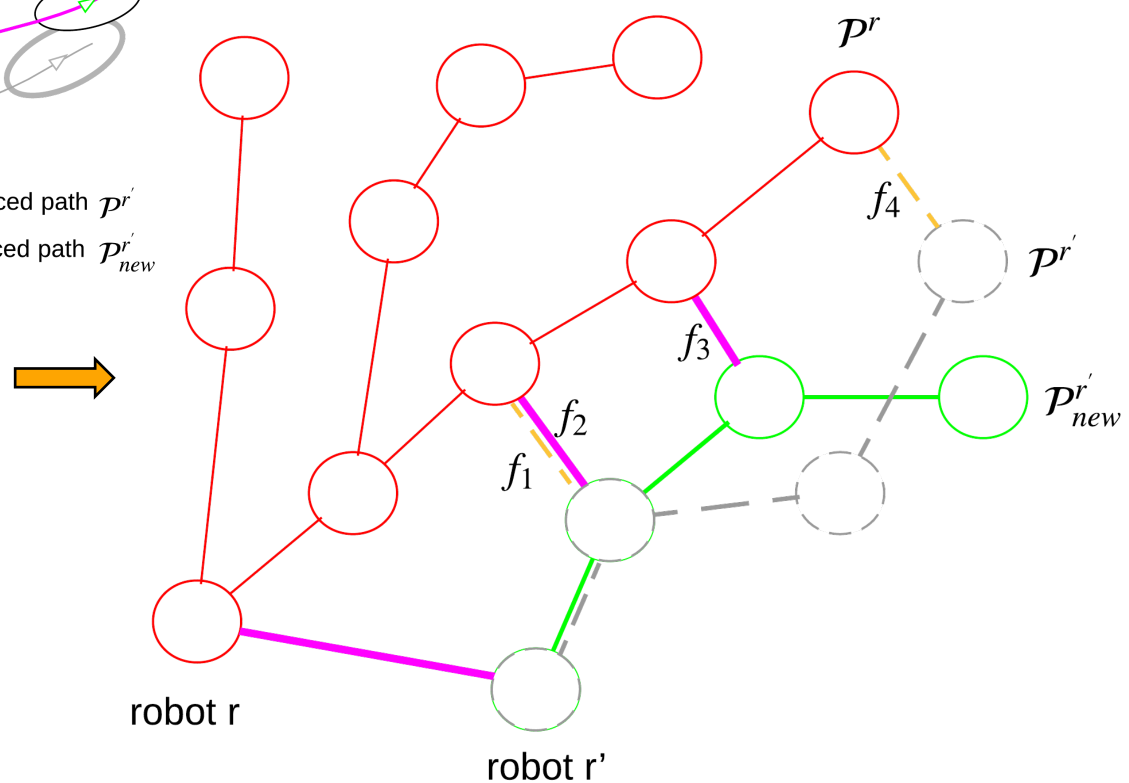
Algorithm Overview – Prior Correlation

- Two possible cases
  - **With multi-robot factors**
  - Without multi-robot factors



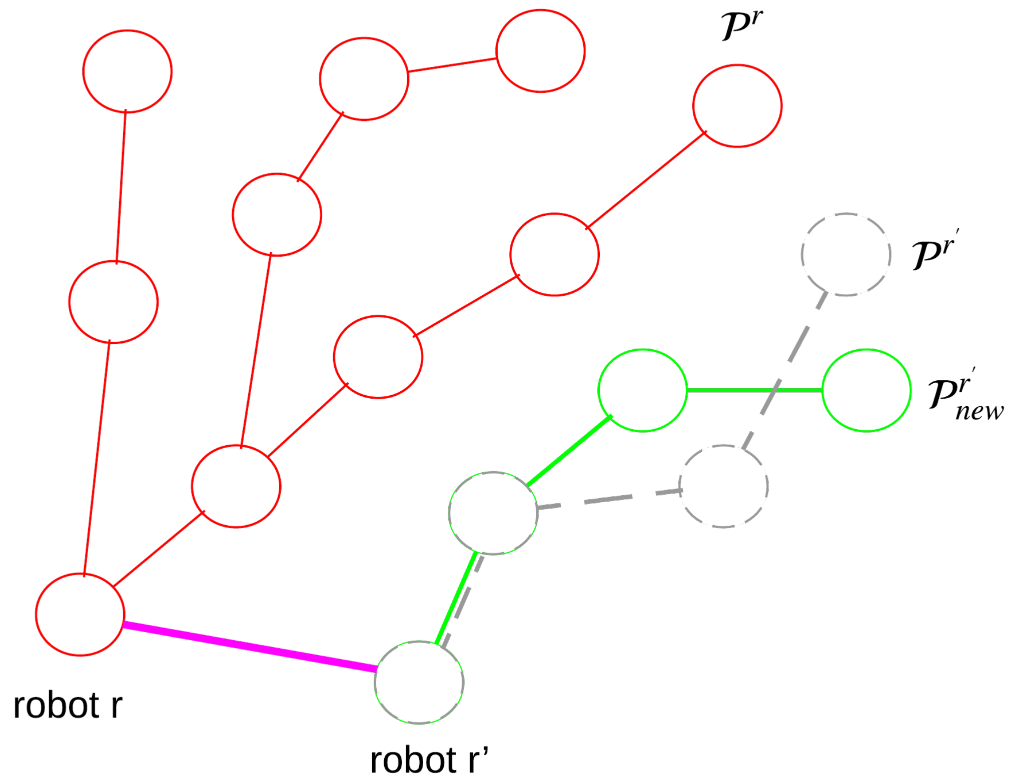
$p(X_k | Z_{0:k}, U_{0:k-1})$

Multi-robot pdf at planning time t_k



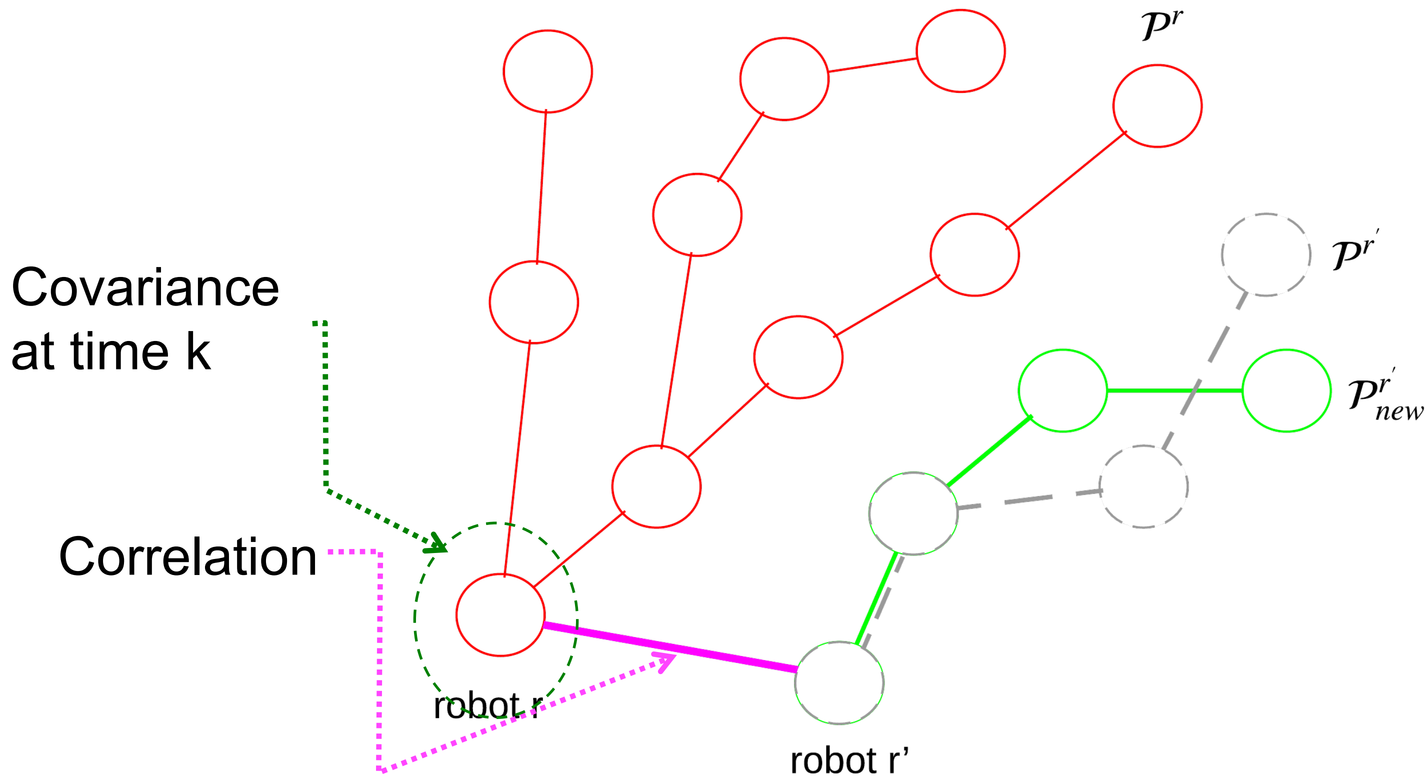
Algorithm Overview – Prior Correlation

- Two possible cases
  - With multi-robot factors
  - **Without multi-robot factors**



Algorithm Overview – Prior Correlation

- Two possible cases
  - With multi-robot factors
  - **Without multi-robot factors**
- If covariance changes significantly, mark all paths
- Otherwise, do not mark

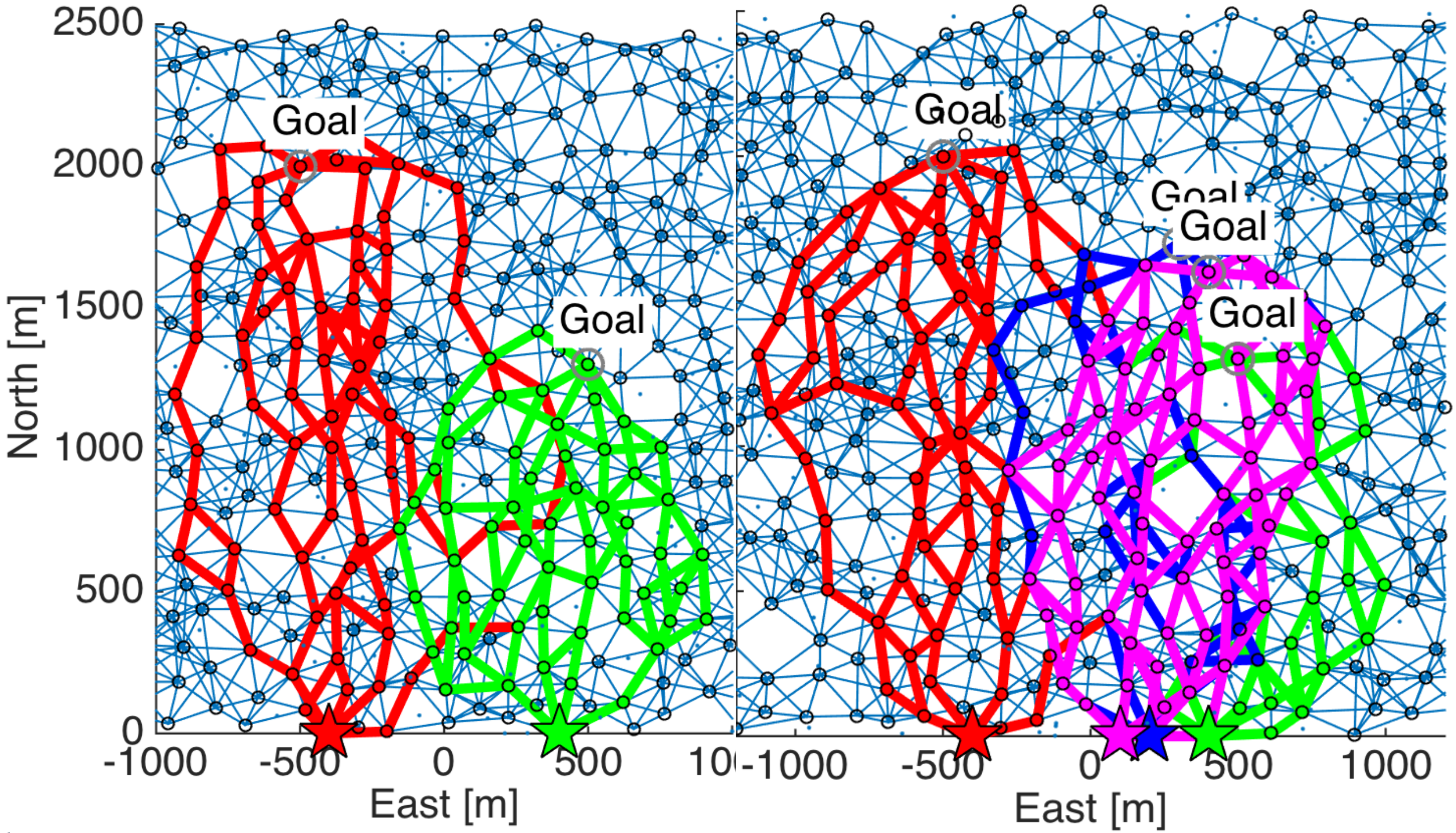


Results

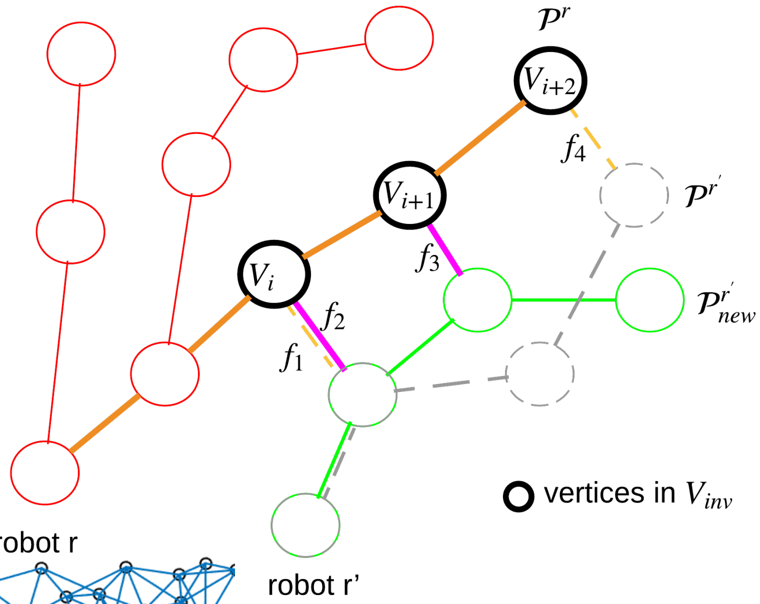
- Scenario: multiple robots autonomously navigating to goals in unknown environment
- Simulation results
  - 25 candidate paths per robot
  - PRM
  - No GPS
- Next slides:
  - Basic scenario: In-depth study for single goal
  - Larger scale scenario: multiple goals and planning sessions, SLAM in between

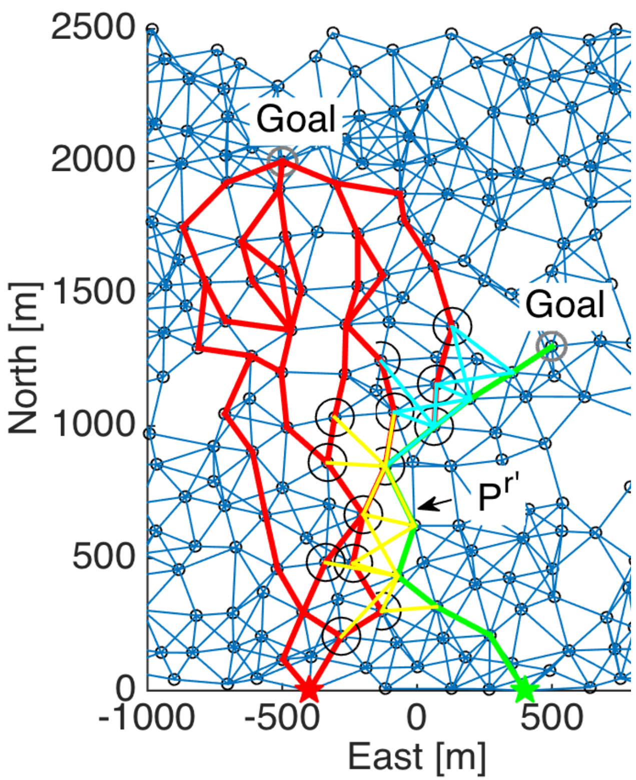
Results - Basic scenario

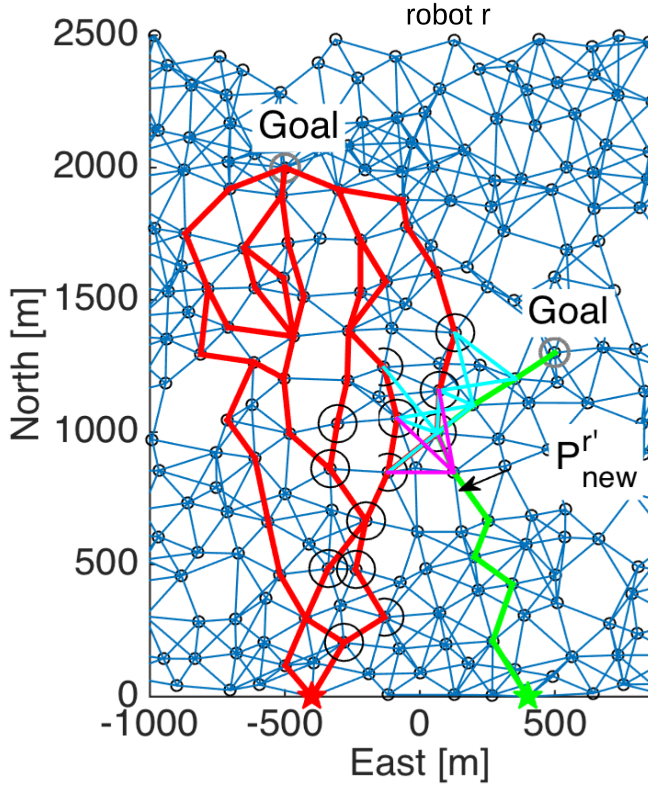
2 robots: 25 candidate paths **4 robots: 25 candidate paths**



Results - Basic scenario



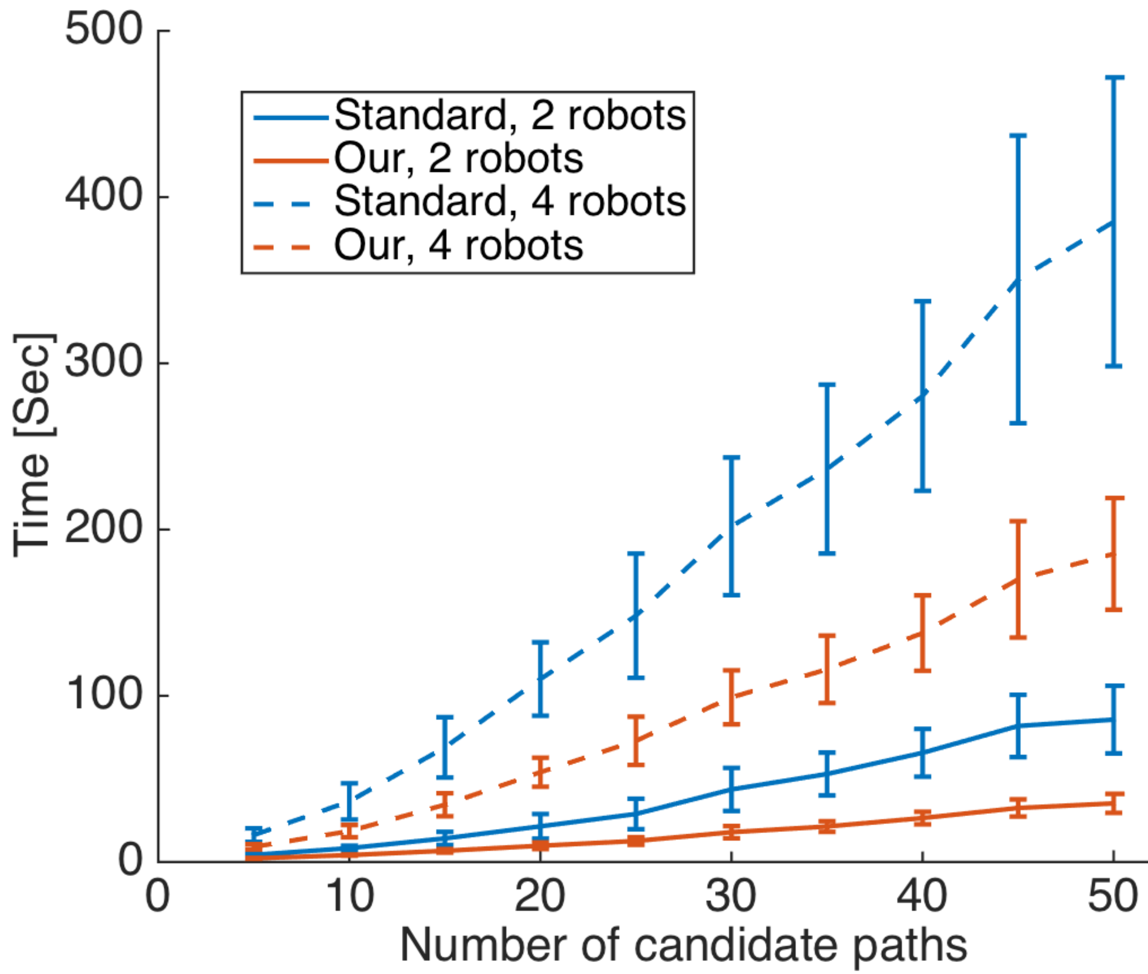




Results - Basic scenario

- **Statistical study of 50 runs (2 and 4 robots):**

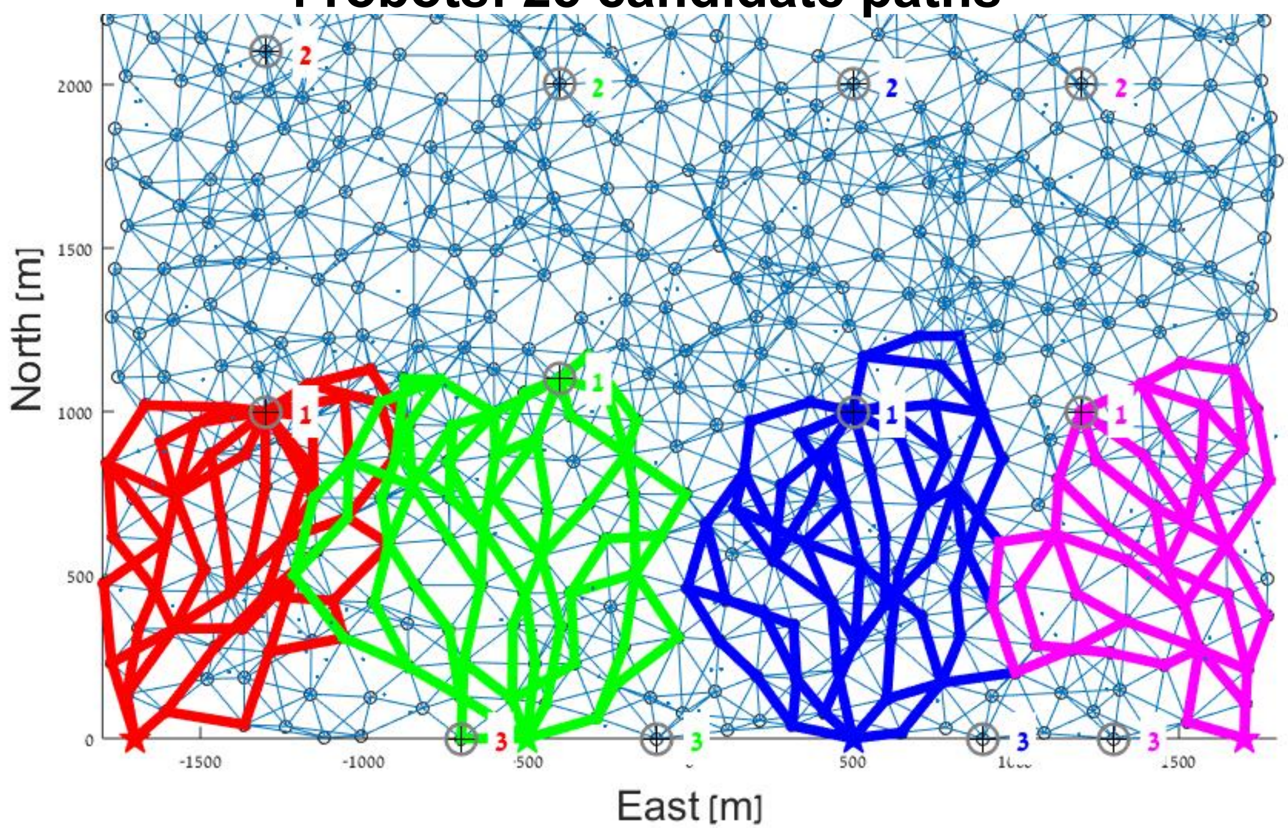
- **More than twice efficient**
- **Same results as Standard approach**



Results – Larger Scale Scenario

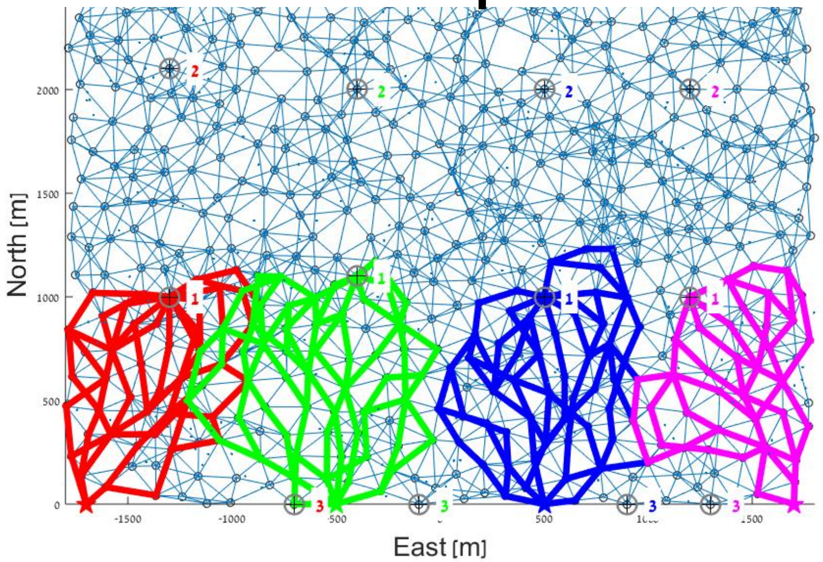
- Each robot has multiple goals
- Multiple planning sessions
- SLAM session given calculated robot paths (actions)
- Two first robots start with high position uncertainty

4 robots: 25 candidate paths

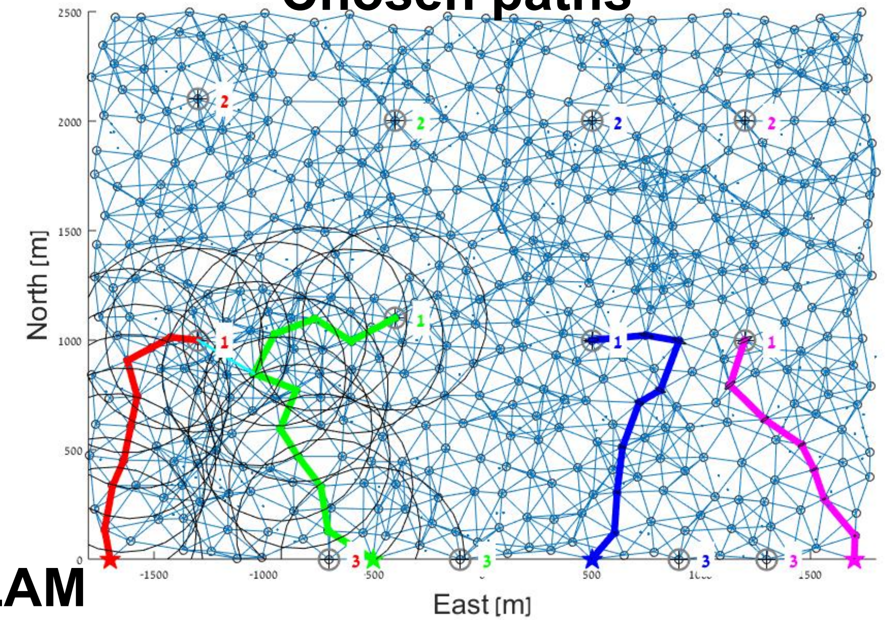


Results – Larger Scale Scenario – Planning 1

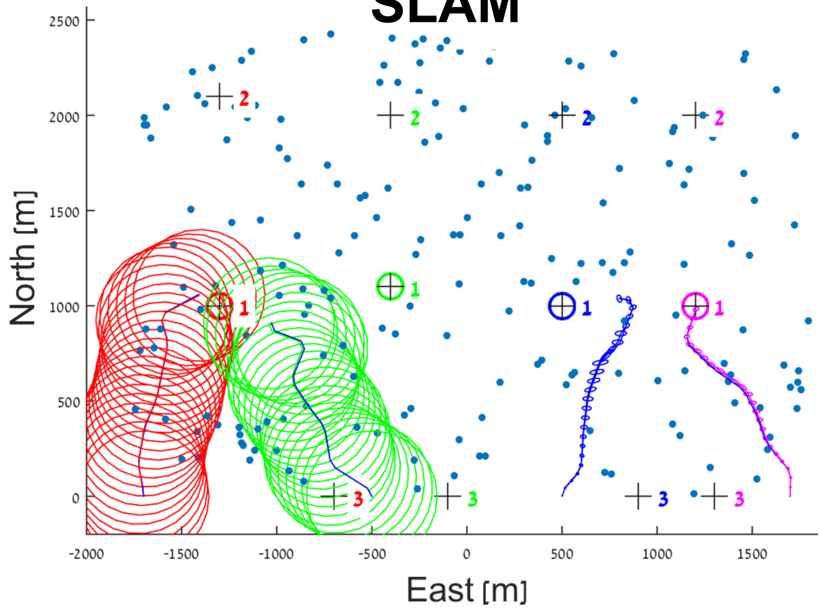
Candidate paths



Chosen paths

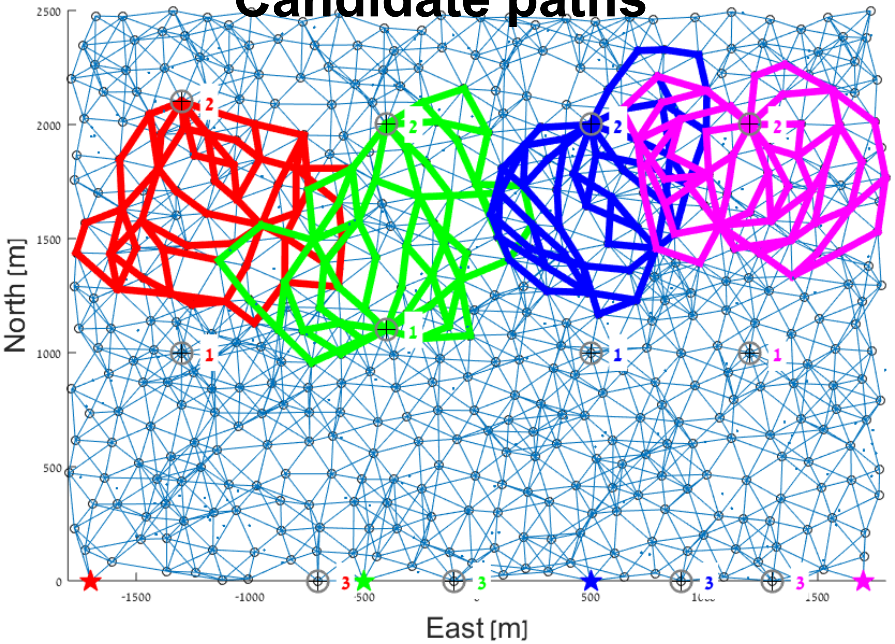


SLAM

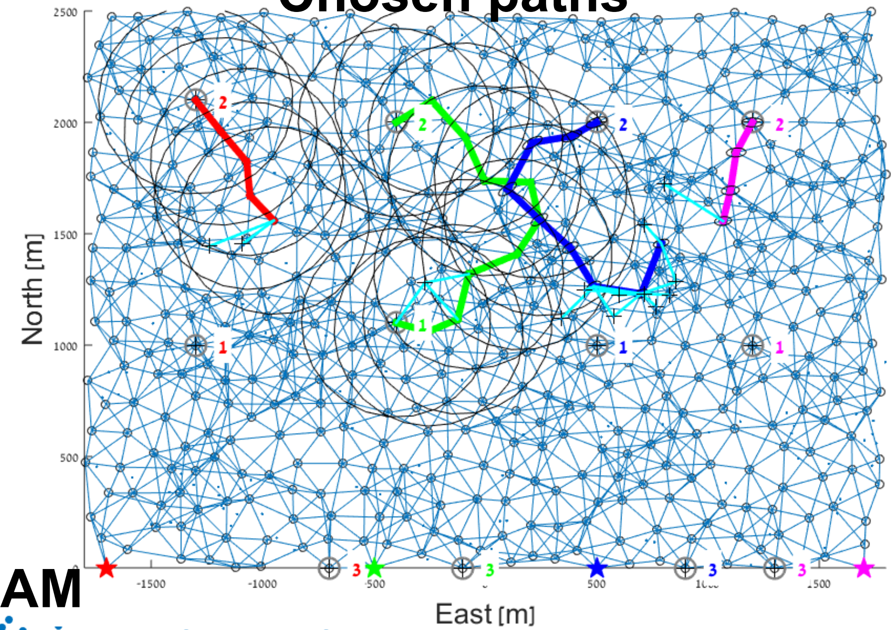


Results – Larger Scale Scenario – Planning 2

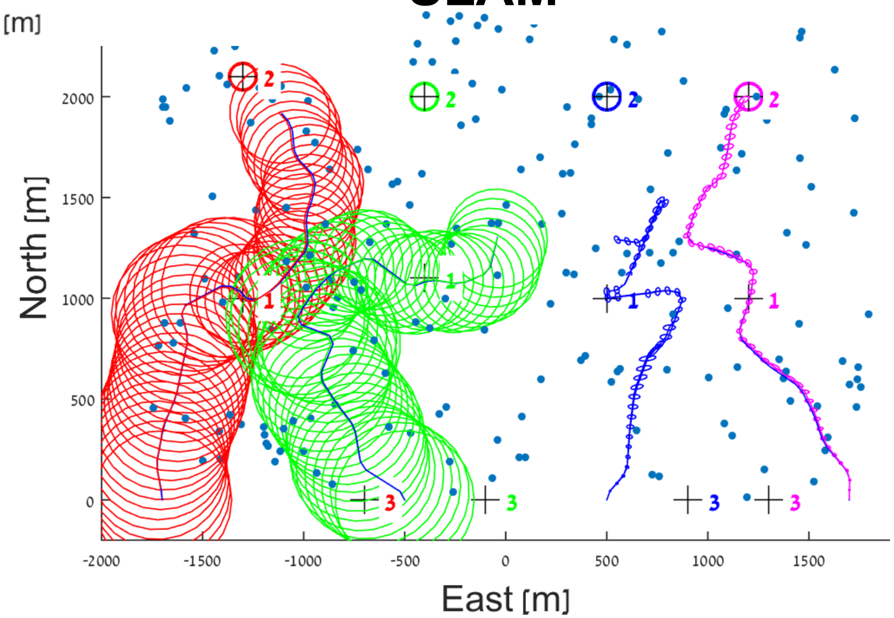
Candidate paths



Chosen paths

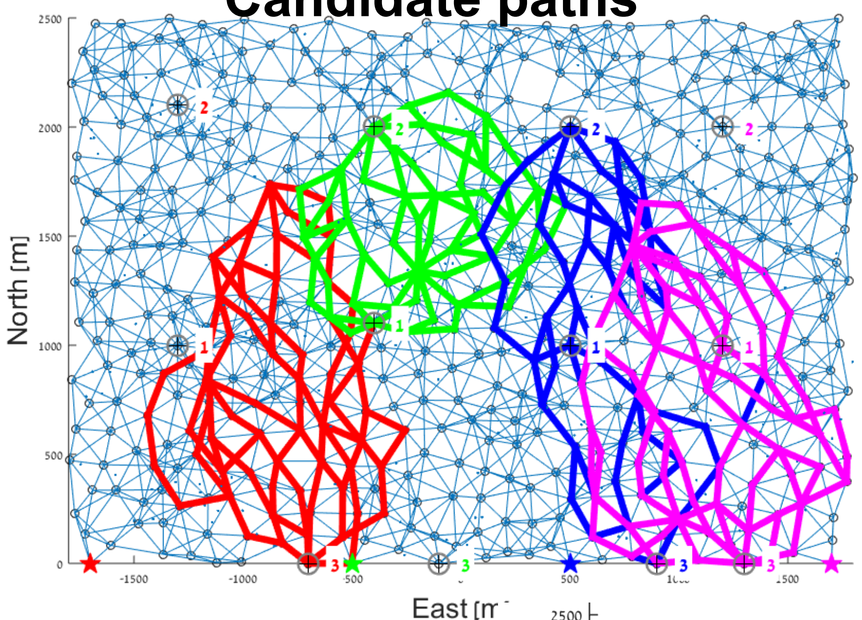


SLAM

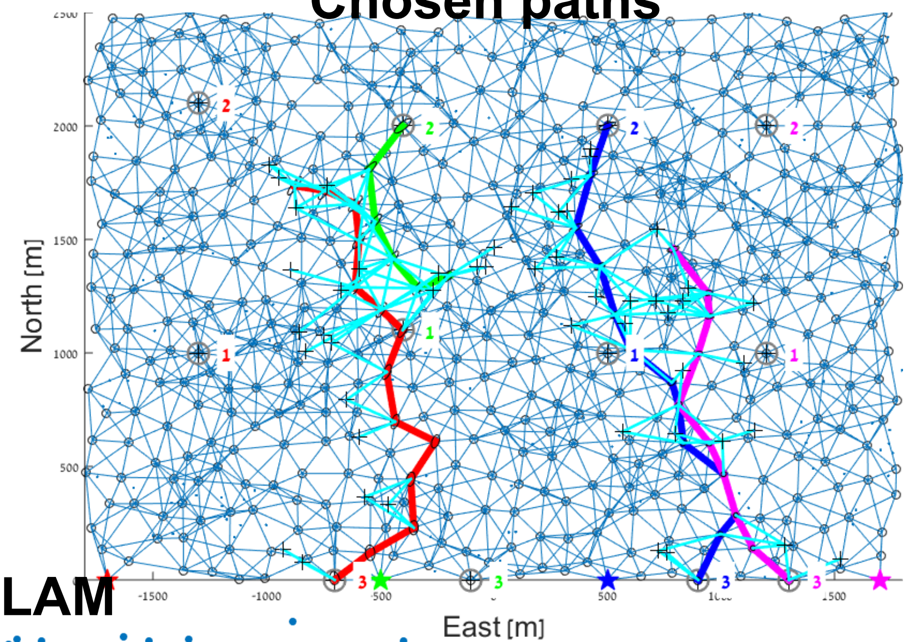


Results – Larger Scale Scenario – Planning 3

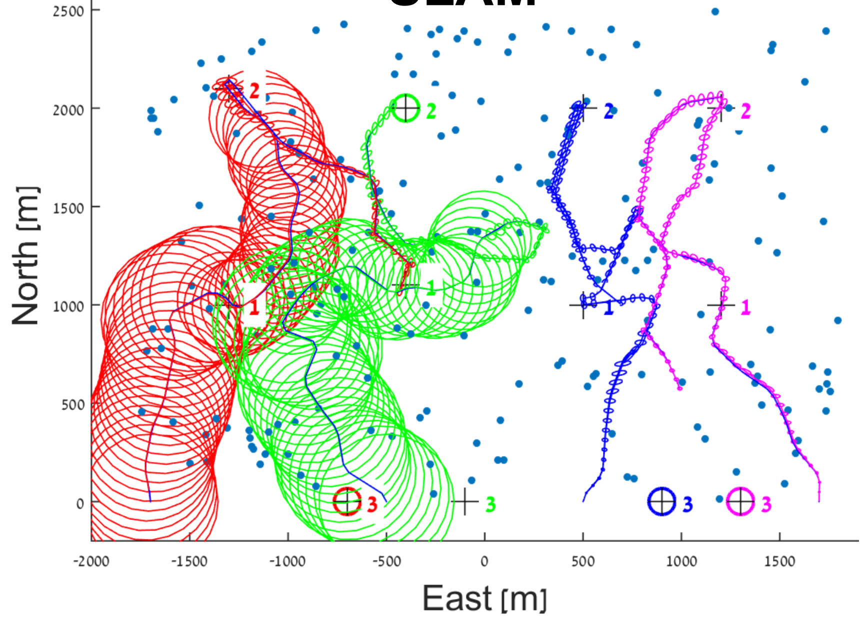
Candidate paths



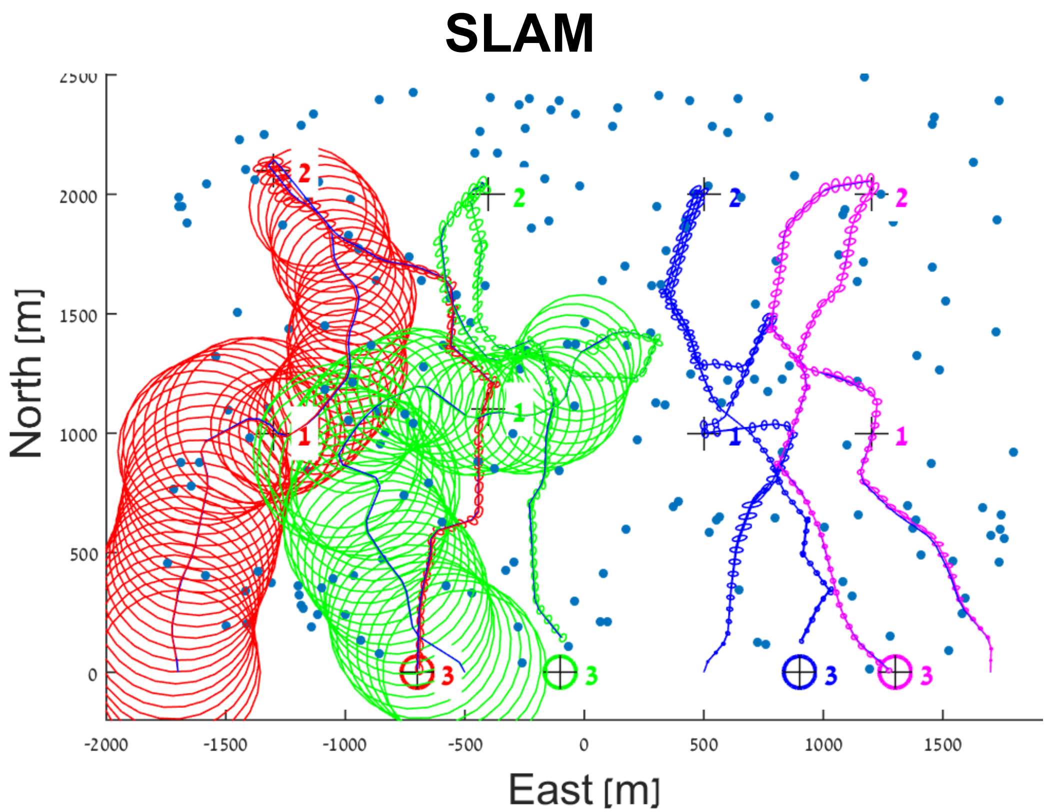
Chosen paths



SLAM



Results – Larger Scale Scenario – Final Result

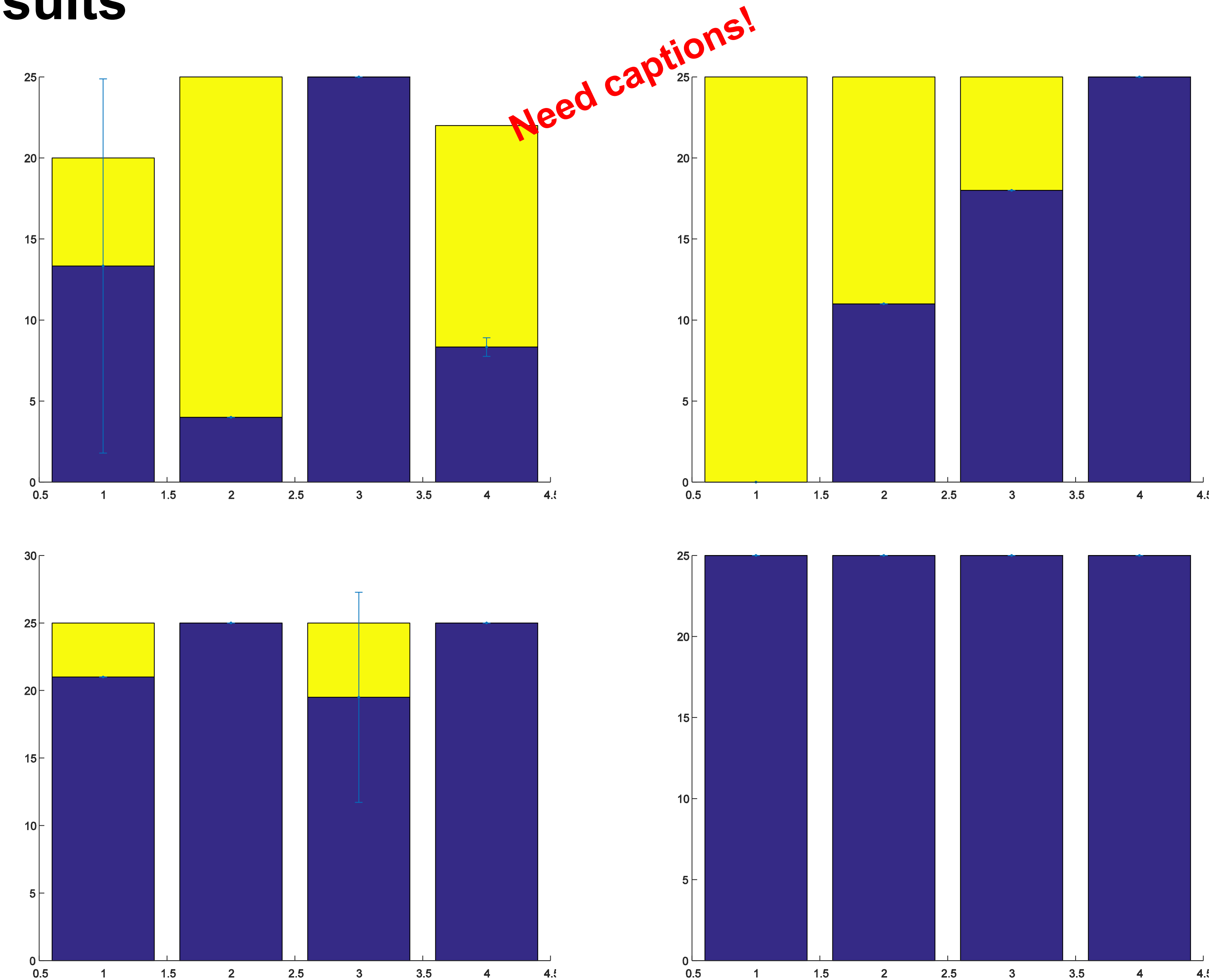


Conclusions and Future Work

- Collaborative multi-robot belief space planning in unknown environments
- **Contribution**:
  - Identify impacted paths due to change in announced paths
  - Efficiently re-evaluate belief only for impacted paths
  - One-time re-calculation for all non-impacted paths
  - Performance study in simulation

- Future work includes:
  - Concept may be generalized to other BSP approaches
  - Implement method in an incremental setting (e.g. RRG, RRT)
  - Extend approach to active cooperative localization and target tracking

Results



Results

Timing results

Introduction – Localization

- Navigation in **known environment** or with GPS.

  Localization: Where am I?

What happens when map is unknown and without GPS?