PIVOT: Predictive Incremental Variable Ordering Tactic for Efficient Belief Space Planning

Khen Elimelech and Vadim Indelman

Technion – Israel Institute of Technology

Introduction

Variable ordering has been widely examined for the state inference problem, but hardly so in the context of planning.

We present a novel tactic to improve the efficiency of Belief Space Planning (BSP), by minimizing the cost of belief updates, with no sacrifice in accuracy. The tactic also helps cutting down on the cost of loop-closing in inference.

The approach continues our previous work which examined efficient planning via belief sparsification [1,2].

Problem Definition

In a sequential Gaussian BSP problem, the belief at time k, given the controls and observations taken until that time, is:

$$b(\boldsymbol{X}_k) \doteq \mathbb{P}(\boldsymbol{X}_k | u_{1:k}, z_{1:k}) \approx \mathcal{N}(\boldsymbol{X}_k^*, \boldsymbol{\Lambda}_k^{-1})$$

The belief is represented using $\boldsymbol{R}$, the upper-triangular information root matrix, such that $\boldsymbol{R}^T\boldsymbol{R} = \boldsymbol{\Lambda}_k$.

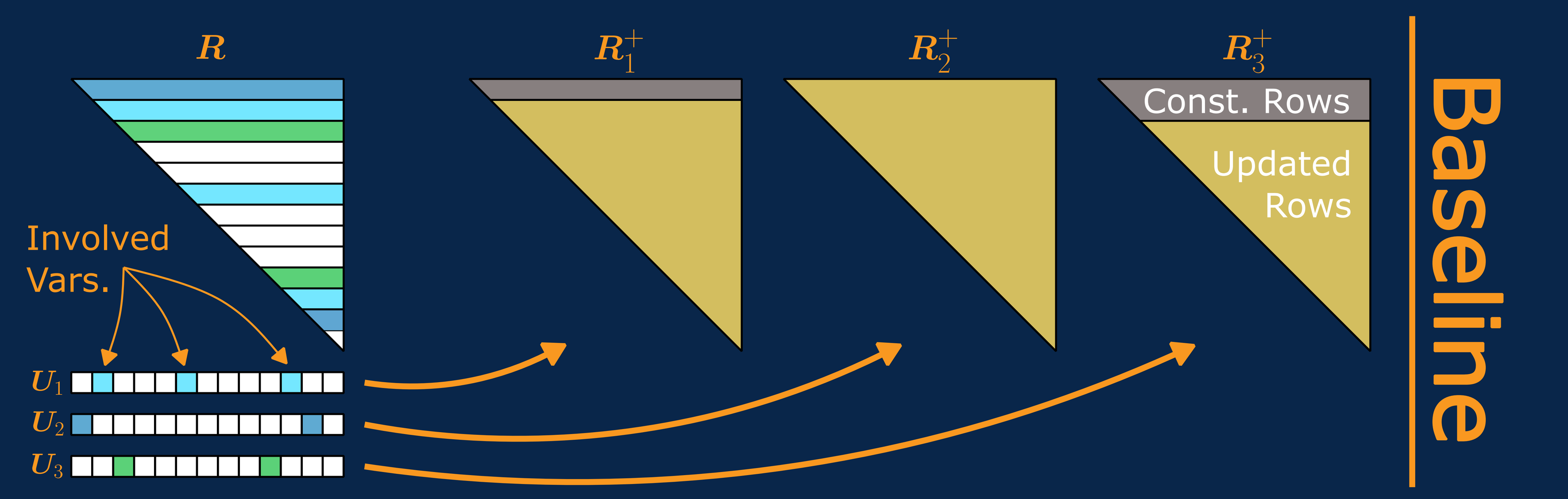
In information-theoretic BSP, we may measure the posterior uncertainty with $J(b,u) \doteq \ln|\boldsymbol{R}^+| + \eta$, where $\boldsymbol{R}^+$ is the posterior information root matrix considering control u.

Given a set $\mathcal{U}$ of candidate control actions, we wish to find the optimal one $u^* = \underset{u \in \mathcal{U}}{\operatorname{argmax}} J(b,u)$.

Approach

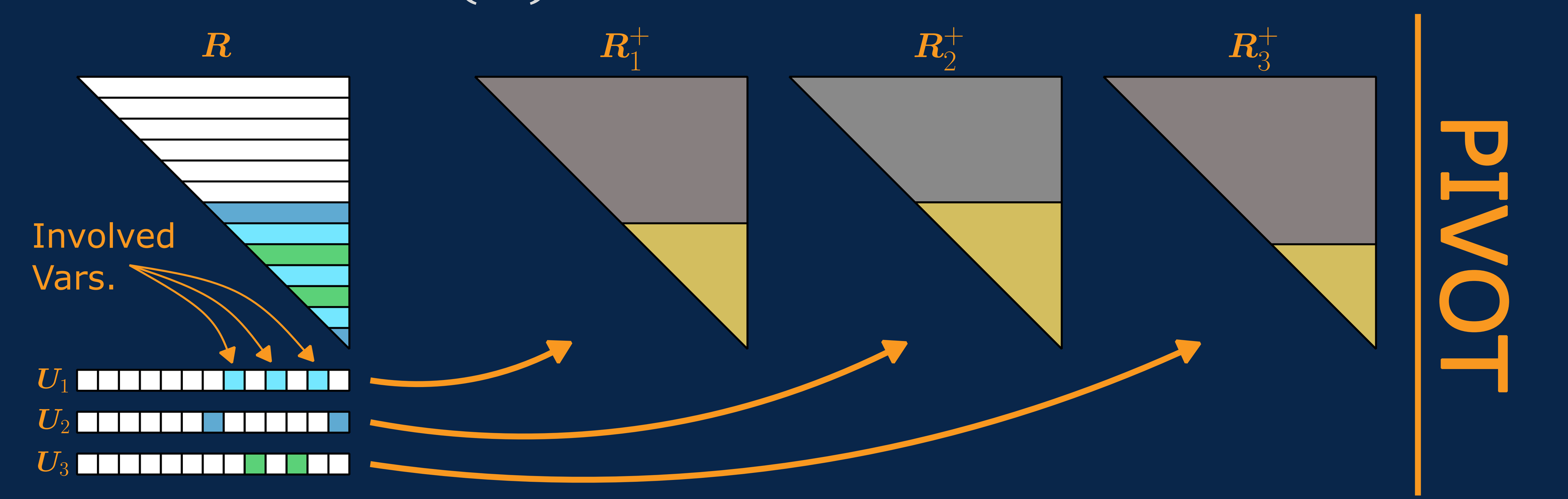
In planning, update the posterior information root matrix for each candidate action u using its (whitened) Jacobian $\boldsymbol{U}$.

For each Jacobian, identify the involved variables (non-zero columns), and update only from the first involved variable.



Avoid updating unnecessary variables while planning, by applying the predictive variable order:

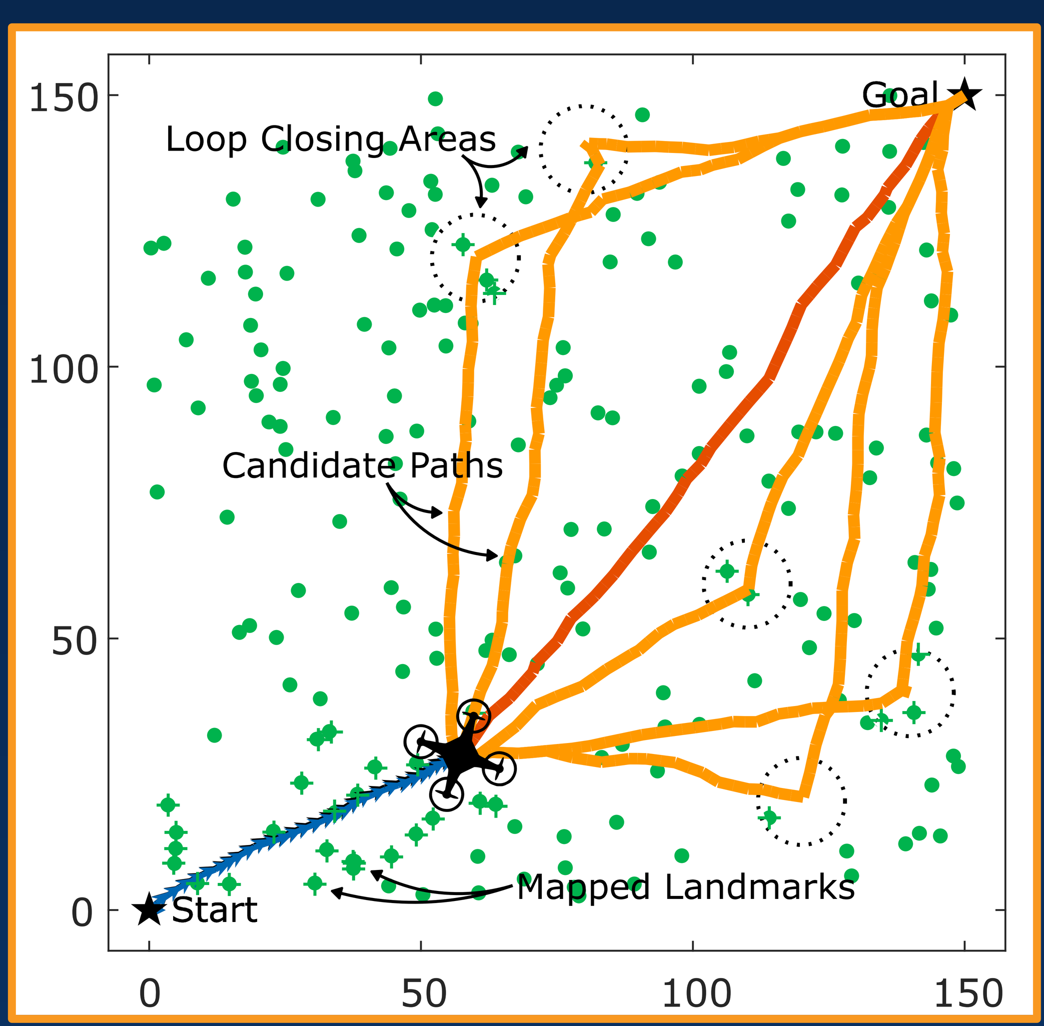
$$P = \begin{bmatrix} \neg\mathcal{J}nvolved(\mathcal{U}) \\ \mathcal{J}nvolved(\mathcal{U}) \end{bmatrix}$$ pushing involved variables forwards.



✓ **Order can be incrementally updated when re-planning!**

✓ **Contributes to loop-closing in later inference sessions!**

Results

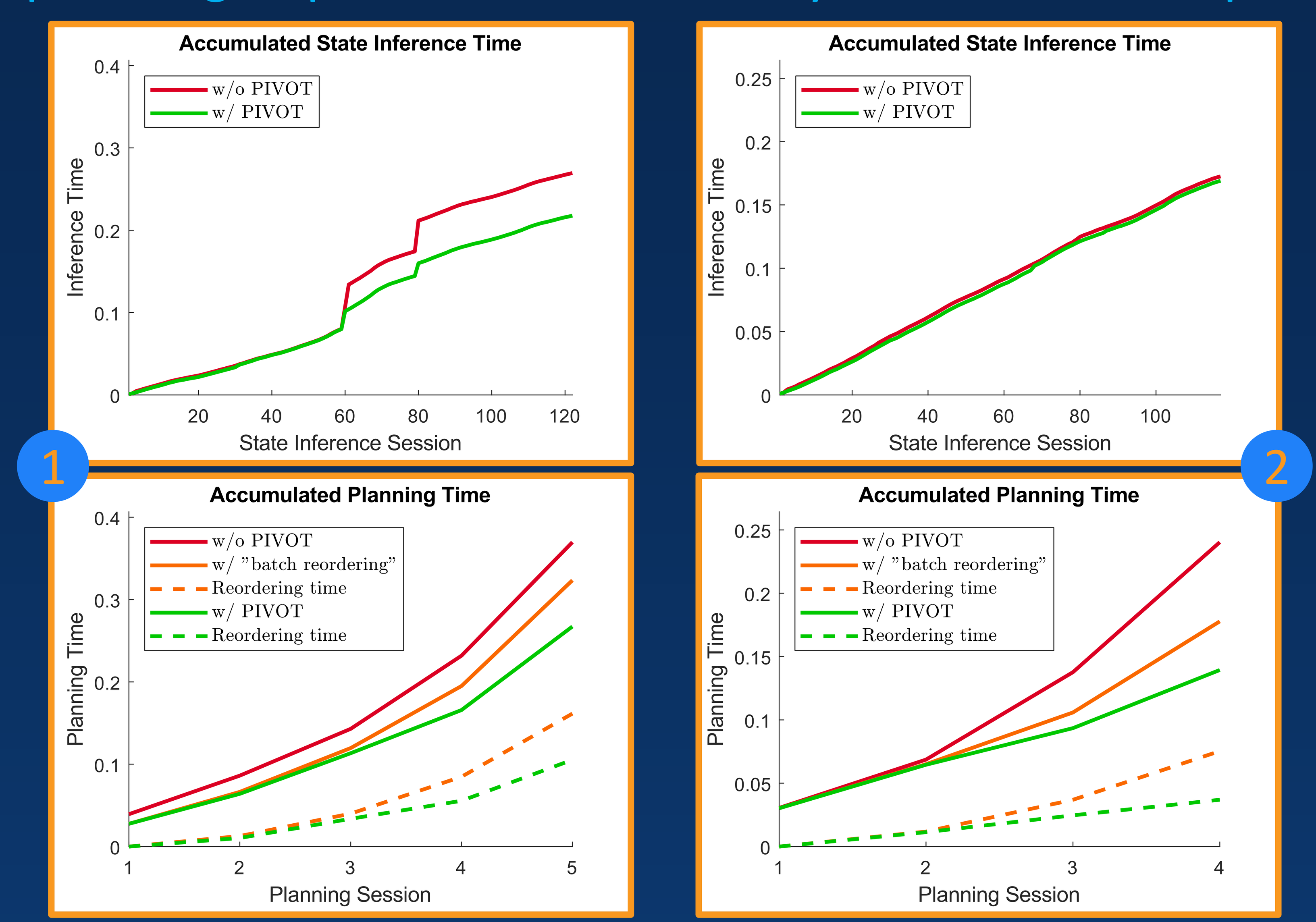


Active-SLAM problem: navigating to a set goal.

Unknown map, prior knowledge only over small areas.

Deciding between a short direct trajectory, and longer loop-closing trajectories, in order to reduce uncertainty.

Accumulated planning and inference times. Two scenarios:
(1) the path included loop-closing (left);
(2) the agent headed directly to the goal (right).
Re-planning is performed after every 30 inference steps.



References

[1] K. Elimelech and V. Indelman. Consistent sparsification for efficient decision making under uncertainty in high dimensional state spaces. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2017.

[2] K. Elimelech and V. Indelman. Scalable sparsification for efficient decision making under uncertainty in high dimensional state spaces. In *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2017.

This work was supported by the Israel Science Foundation.
The authors also thank Intel AI for supporting this work.
Experiments used Intel Core i9-7920X CPU.

www.khen.io

ANPL Autonomous Navigation and Perception Lab

TECHNION Israel Institute of Technology