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

Khen Elimelech and Vadim Indelman

ANPL Autonomous Navigation and Perception Lab

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.

Problem Definition

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

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

Given a set $\mathcal{U}$ of candidate control actions, we wish to find the optimal one

$$u^* = \underset{u \in \mathcal{U}}{\operatorname{argmax}} J(b, u).$$

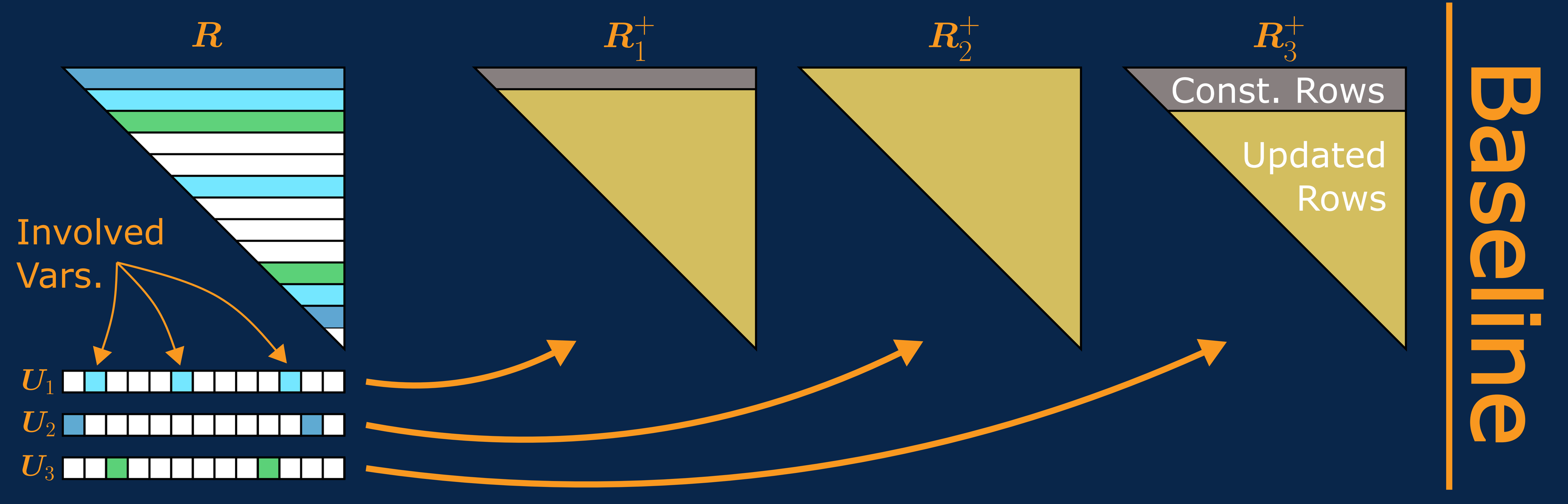
For example, in information-theoretic BSP, we may measure the posterior uncertainty (entropy). To do so, we should propagate the belief according to the candidate actions.

In state-of-the-art smoothing and mapping approaches, the belief is represented using $\boldsymbol{R}$, the upper-triangular information root matrix, such that $\boldsymbol{R}^T\boldsymbol{R} = \boldsymbol{\Lambda}_k$. Each row of the matrix corresponds to a state variable (in order).

Then, when planning, for each candidate action u, we should use its (whitened) Jacobian $\boldsymbol{U}$ in order to update the prior information root matrix $\boldsymbol{R}$. Each column of the Jacobian matrix corresponds to a state variable (in order), and each row to a new constraint/factor.

Approach

For each Jacobian, we can identify the involved variables (non-zero columns). According to the information root formulation, the belief should be incrementally updated only starting from the first involved variable.

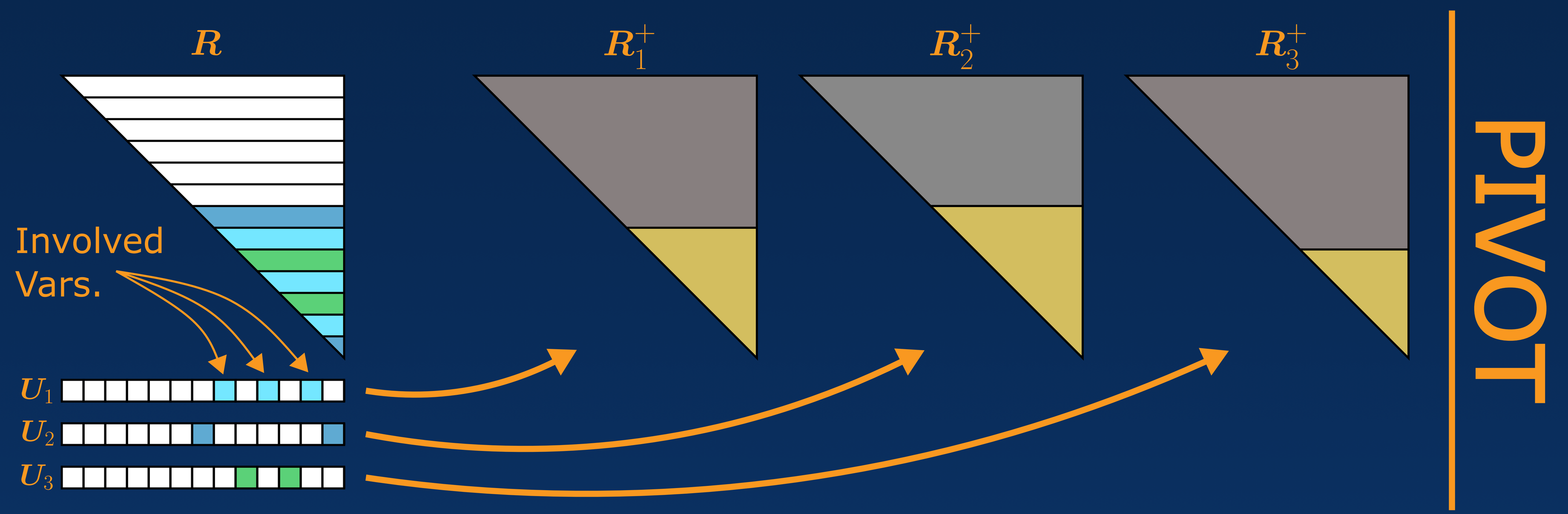


Even rows of uninvolved variables may be affected by the updates, due to poor variable order.

The concept:

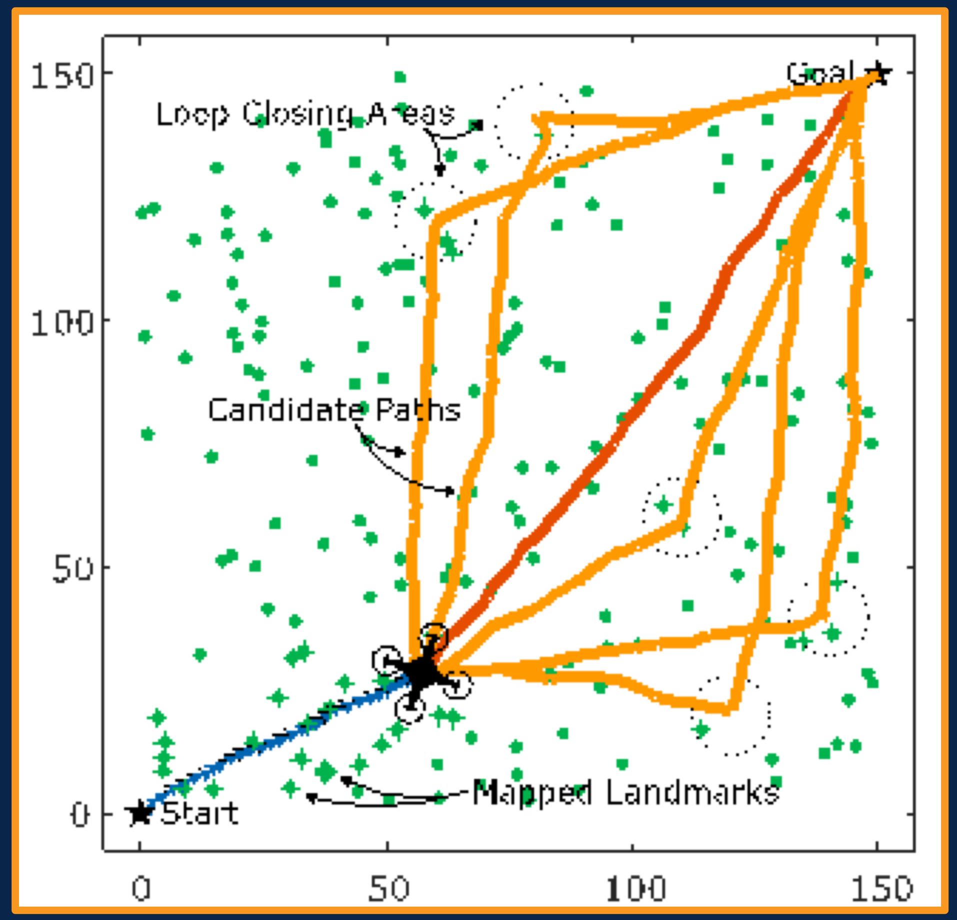
Avoid updating unnecessary variables while planning, by applying the predictive variable order before each planning session:

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



- ✓ **Conceptually Novel!**
  Precursory re-ordering based on (predicted) future knowledge
- ✓ **No sacrifice in accuracy**
- ✓ **Order can be incrementally updated when re-planning!**

Results



Active-SLAM problem: Robot navigating to a goal, while mapping the unknown environment.

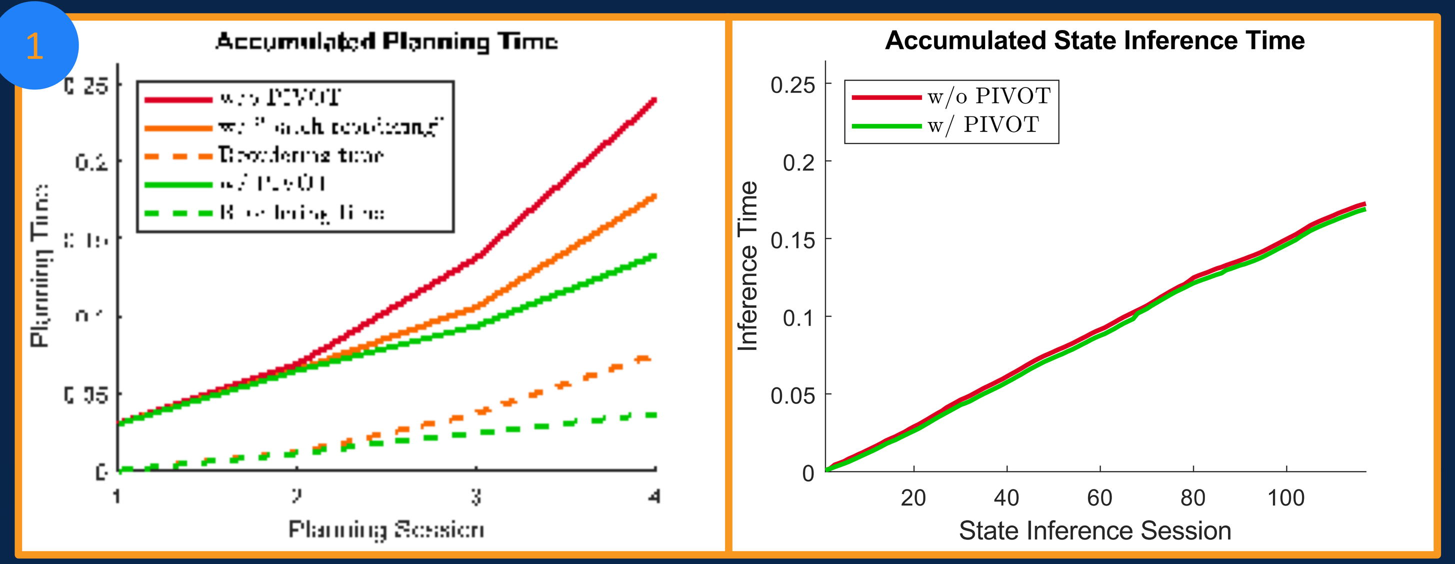
(Noisy) prior knowledge in small areas

Objective function balances trajectory length vs. uncertainty reduction, via loop closing.

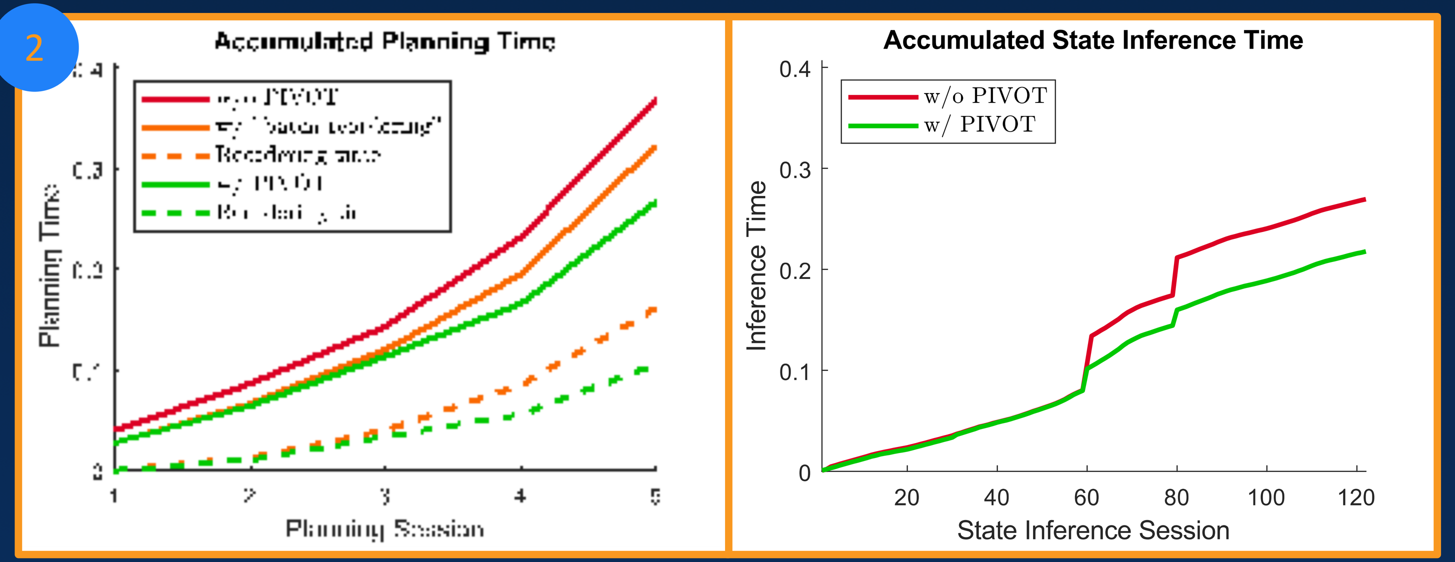
Re-planning is performed every 30 inference steps.

Results summary:
Accumulated planning and inference times.

(1) the agent headed directly to the goal;



(2) the path included loop-closing.



- ✓ **We see a contribution to loop-closing in later state inference sessions, by pushing forward relevant variables ahead of time.**
- ✓ In both cases we demonstrate a reduction of 25% of the total run-time (inference + planning), with no sacrifice in accuracy.

A new preprint "Efficient Decision Making and Belief Space Planning using Sparse Approximations" is available now. Check out www.khen.io