An AI-Based Planning Framework for HAPS in a Time-Varying Environment

Jane Jean Kiam*, Enrico Scala**, Miquel Ramirez***, Axel Schulte*

* Bundeswehr University Munich
** University of Brescia
*** University of Melbourne

Email contact: jane.kiam@unibw.de

0

High Altitude Pseudo Satellite (HAPS)

Operating altitude: ~18 km

Long endurance: months

Light-weight: 120 kg

Completely solar-powered

Fig. 1 HAPS as an alternative to satellites for easy payload maintenance and flexible deployment

Low airspeed: ~30 m/s

Fixed-wing aircraft

Limited payload

Problem Statement

Repetitive monitoring of the points of interest (PoI) using multiple HAPS while considering:

- PoI ▲ and △ must be monitored using an infrared (IR) camera and an electro-optical (EO) camera respectively, and within a specific time window;
- A HAPS carries one camera, either IR of EO;
- Once onboard memory is full, the HAPS transfers the image data to the ground control station (GCS) at ●, where line-of-sight communication with the GCS is possible;
- The operation must take place within ▱; ⬣ are forbidden airspace, representing either an occupied airspace or a dangerous weather zone;
- Wind field affects the ground speed of the HAPS.

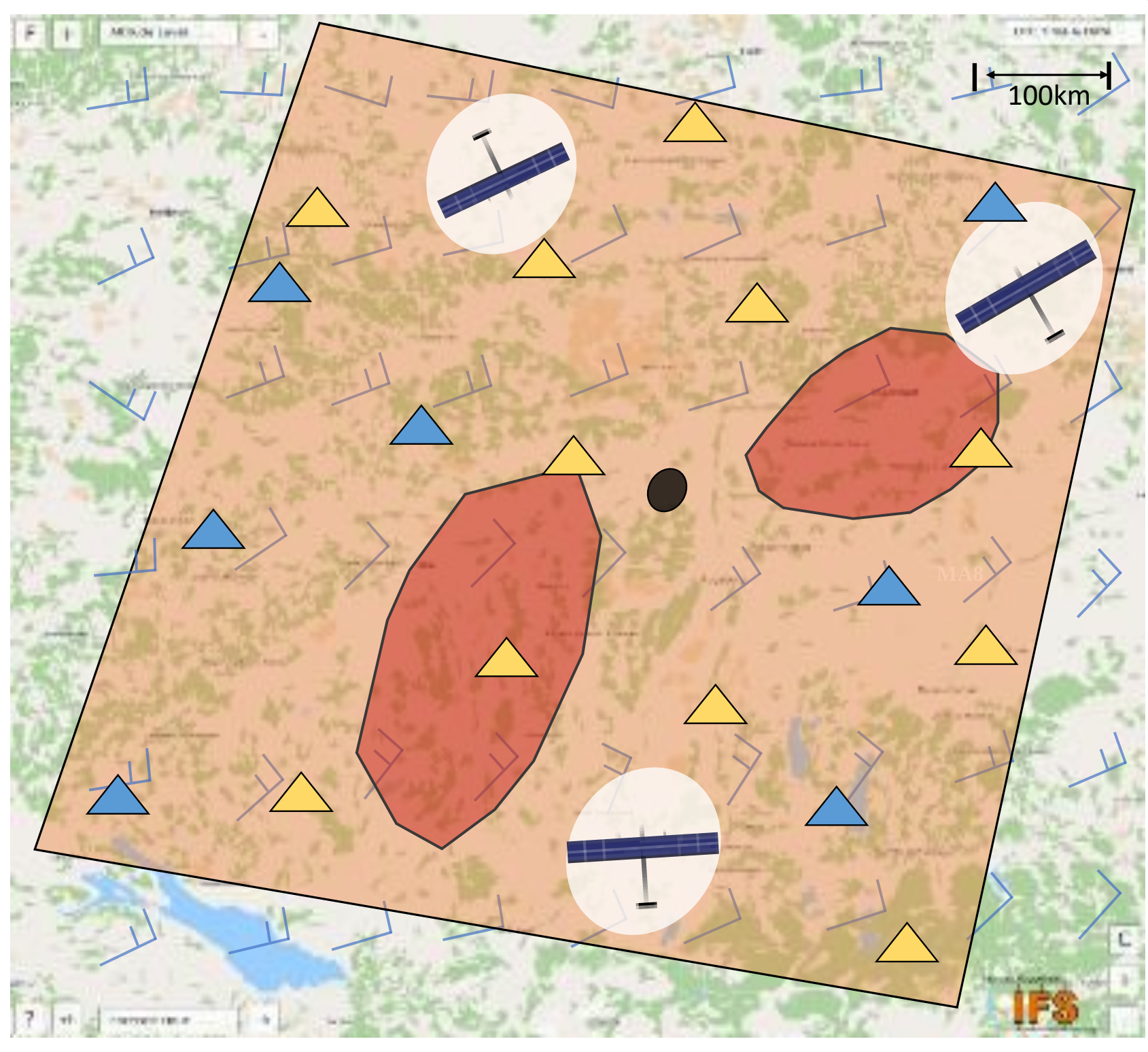


Fig. 2 Typical mission scenario for HAPS: monitoring of PoI

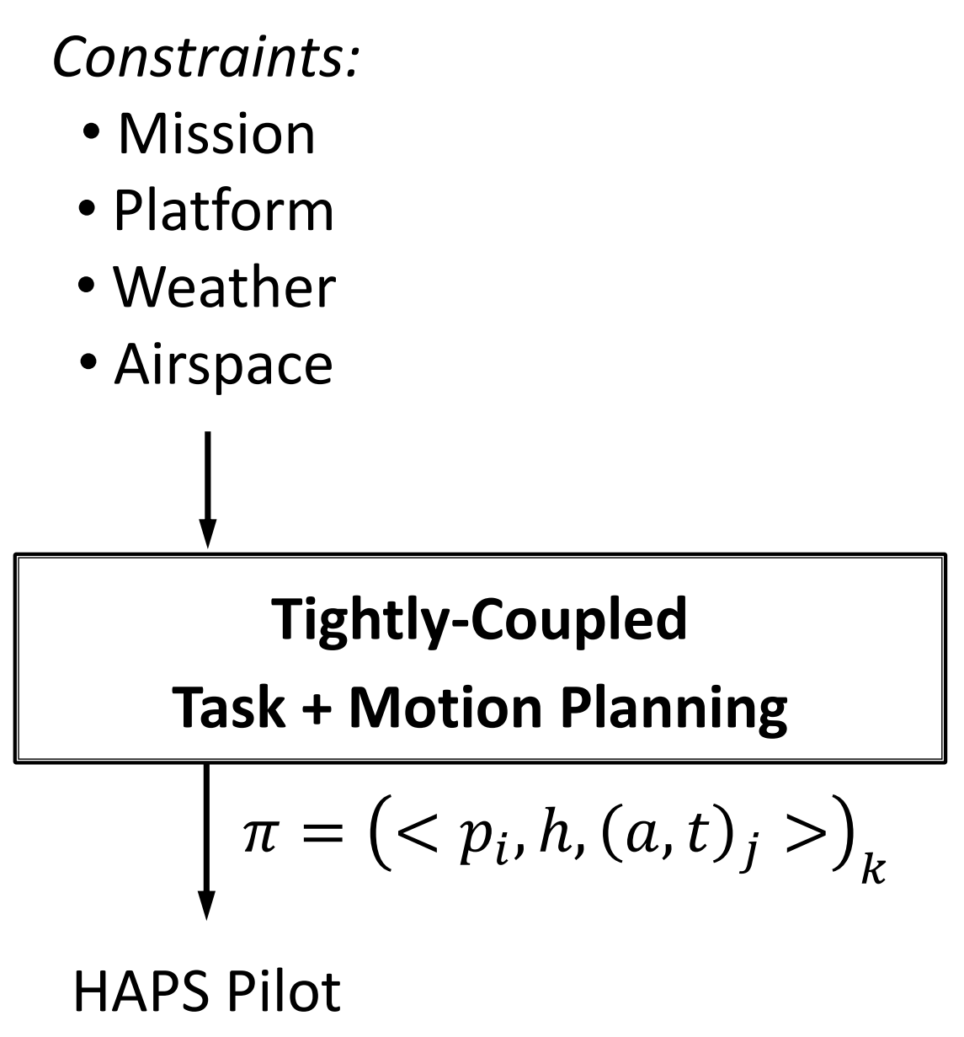
1

Tightly-coupled Task+Motion Planning with PDDL+?

HAPS planning problem is a combination of subproblems (1) a time-dependent multi-vehicle routing problem, (2) a classical (task) planning problem and a (3) kinodynamic motion planning problem.

PDDL+ (Problem Domain Definition Language) can formulate mixed discrete-continuous planning problems that can be solved with compatible planners.

While the formulation of subproblems (1) and (2) in PDDL+ are more commonly known, the formulation of subproblem (3) is not. Fig. 4 outlines how a kinodynamic motion planning problem can be mapped to PDDL+, while the figure on the right shows the one-step task and motion planning (TMP) architecture intended. *TMP is "tightly-coupled", as the search spaces are merged.*



Constraints:

- Mission
- Platform
- Weather
- Airspace

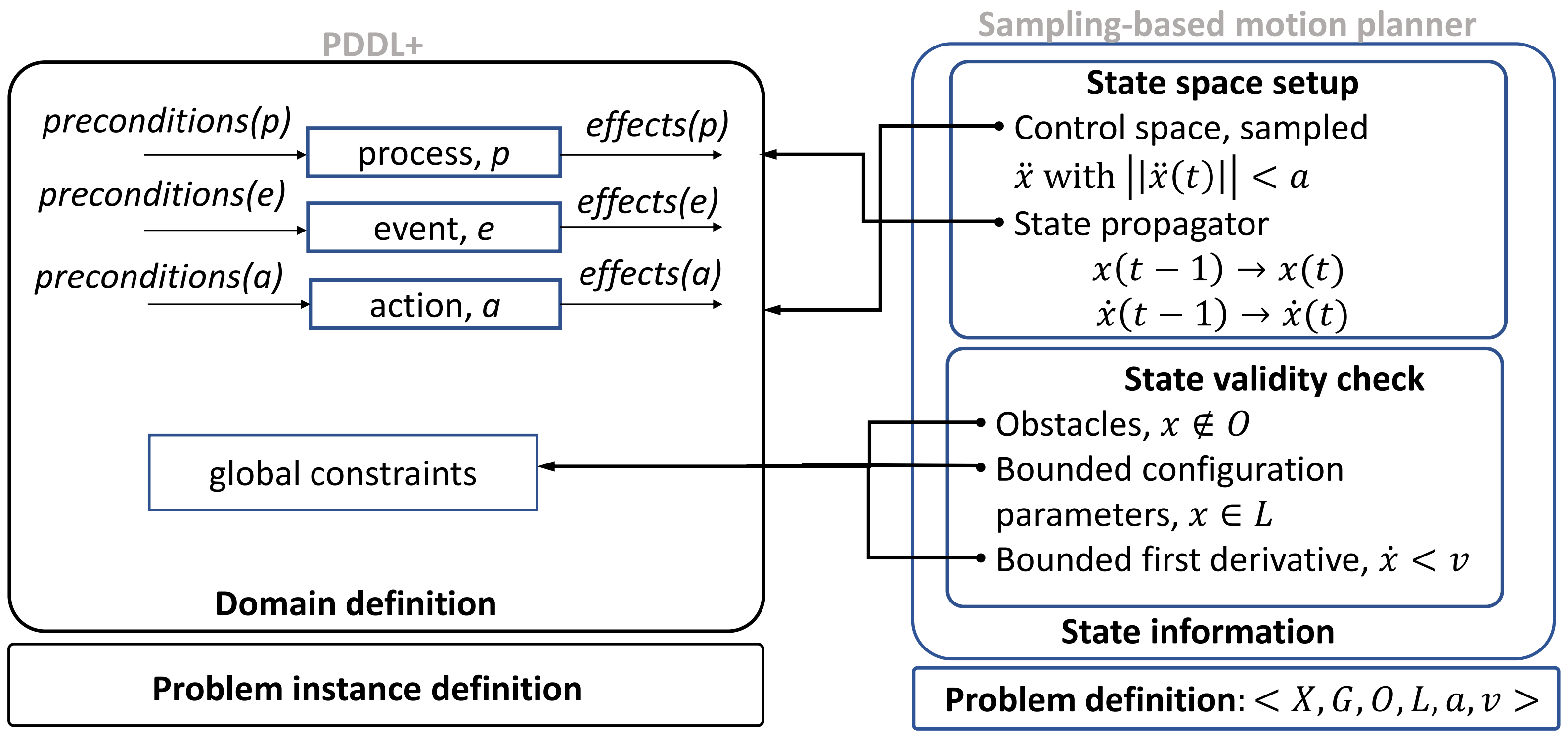
Tightly-Coupled Task + Motion Planning

$$\pi = \left(< p_i, h, (a,t)_j >\right)_k$$

HAPS Pilot

Fig. 3 Tightly-coupled TMP

where p_i is the PoI to monitor, h is the haps assigned, $(a,t)_j$ is the path (i.e. sequence of time-stamped actions) to follow to reach p_i



PDDL+

preconditions(p) → process, *p* → *effects(p)*

preconditions(e) → event, *e* → *effects(e)*

preconditions(a) → action, *a* → *effects(a)*

global constraints

Domain definition

Problem instance definition

Sampling-based motion planner

State space setup

- Control space, sampled $\ddot{x}$ with $||\ddot{x}(t)|| < a$
- State propagator
  $x(t-1) \rightarrow x(t)$
  $\dot{x}(t-1) \rightarrow \dot{x}(t)$

State validity check

- Obstacles, $x \notin O$
- Bounded configuration parameters, $x \in L$
- Bounded first derivative, $\dot{x} < v$

State information

Problem definition: $< X, G, O, L, a, v >$

Fig. 4 Mapping kinodynamic motion planning paradigm into PDDL paradigm

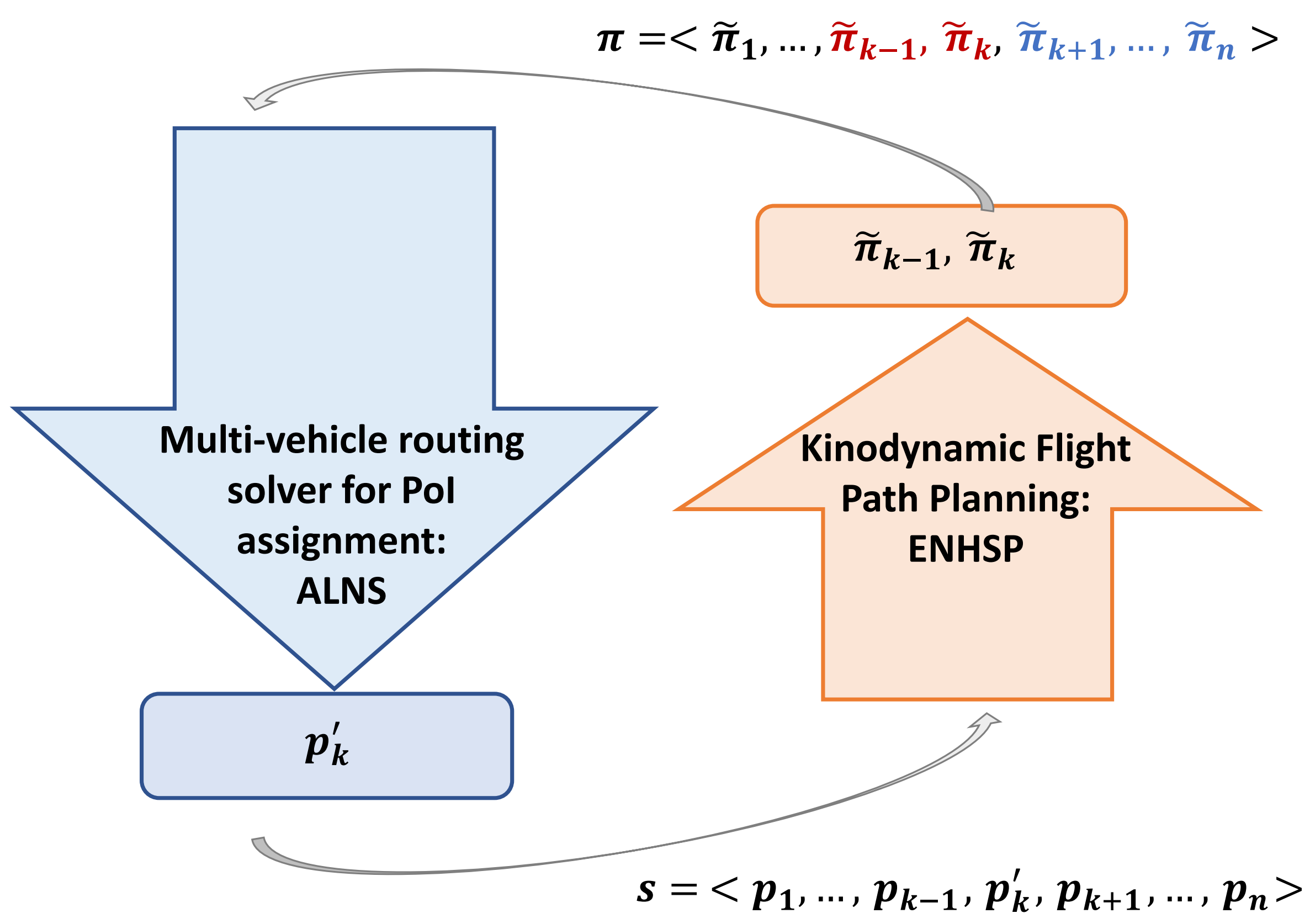
2

*ENHSP (Expressive Numeric Heuristic Search Planner)** is to date the only PDDL+ planner that supports the complex problem, especially thanks to its ability to cope with non-linear numeric operations, e.g. trigonometric operations, needed in the modelling of the platform dynamics, i.e. the update of longitude and latitude of the HAPS respectively:

$$\dot{\lambda} = \frac{v_{\mathrm{wind,E}} + v^*_{\mathrm{TAS}} \cos\gamma \sin\chi}{(R + alt)\cos\phi}, \quad \text{and} \quad \dot{\phi} = \frac{(v_{\mathrm{wind,N}} + v^*_{\mathrm{TAS}} \cos\gamma \cos\chi)}{R + alt}.$$

However, scaling up ENHSP for solving the complex task+motion planning problem with multiple HAPS and multiple PoI in a tightly-coupled manner as in Fig. 3 was not possible. The domain-independent heuristic cannot cope. Therefore, we resort to:

Integrated Task and Motion Planning



$$\pi = < \tilde{\pi}_1, \dots, \tilde{\pi}_{k-1}, \tilde{\pi}_k, \tilde{\pi}_{k+1}, \dots, \tilde{\pi}_n >$$

$\tilde{\pi}_{k-1}, \tilde{\pi}_k$

Multi-vehicle routing solver for PoI assignment: ALNS

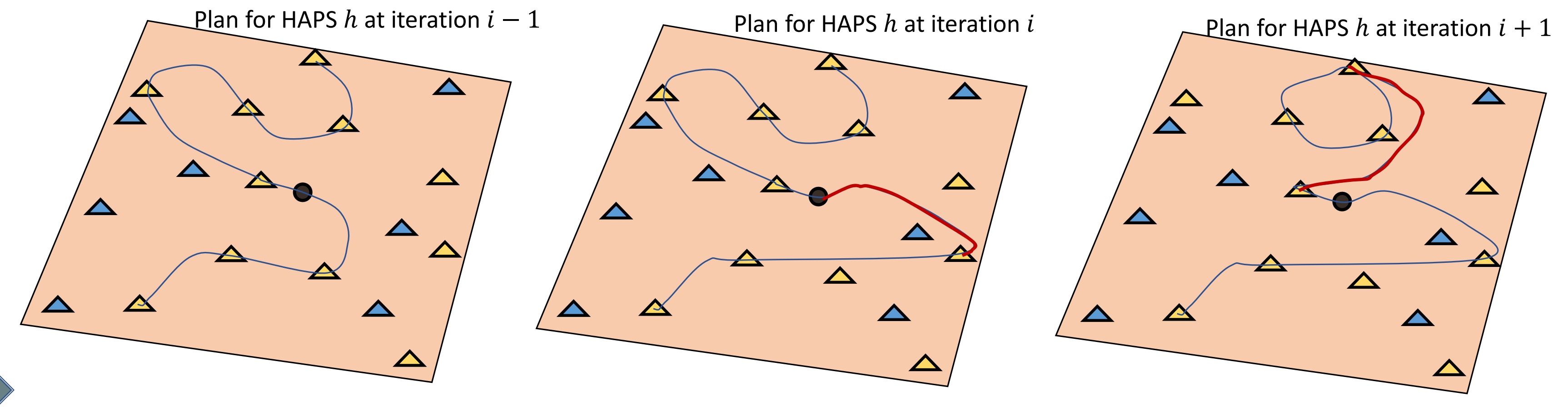
Kinodynamic Flight Path Planning: ENHSP

p'_k

$$s = < p_1, \dots, p_{k-1}, p'_k, p_{k+1}, \dots, p_n >$$

Fig. 5 Integrated task and motion planning: exploring search spaces in an interleaving manner

ALNS (Adaptive Large Neighbourhood Search), a powerful time-dependent multi-vehicle routing problem solver, is exploited for assigning PoI to the HAPS. The search space for motion and simple task planning are embedded in the search space of the multi-vehicle routing planning.



Plan for HAPS h at iteration $i-1$

Plan for HAPS h at iteration i

Plan for HAPS h at iteration $i+1$

Fig. 6 Iterations of ALNS to find a better assignment of PoI resulting in "piecewise" planning problem (simple task +motion planning on the section in red) solvable using ENHSP

In other words, ALNS acts as a "higher-level" wrapper for ENHSP. The advantages of doing so are:

- The search in the "high-level" and "low-level" spaces is *interleaving* (see Fig. 5), instead of *sequential*, as done in previous work, where planning is performed in two steps: strategic and tactical, avoiding therefore an iterative rejection of non-feasible high-level strategic plan by the low-level tactical planner.
- We break down the complex problem into smaller ones that are solvable for ENHSP, and stitch the new "pieces" of plan (or rather partial plan) computed by ENHSP, i.e. $\tilde{\pi}_{k-1}, \tilde{\pi}_k$ with the remaining partial plan, i.e. $\tilde{\pi}_{k+1}, \dots, \tilde{\pi}_n$.
- The implementation can be "anytime", if we initialize ALNS with a feasible plan, since the plan will be improved partially by searching in the neighborhood (see Fig. 6).
- Problem-dependent meta-heuristics can be exploited easily within ALNS.

* Scala E., Haslum P., Thiebaux S., Ramirez M.: (2016) ***Interval-Based Relaxation for General Numeric Planning*** in Proc. of the European Conference on Artificial Intelligence (ECAI-2016)

3

Validation and Performance Benchmarking

Performance tests on the integrated task and motion planning approach against the previous sequential planning framework, in which a complete task plan is generated to be "refined" by the numeric motion planner. The integrated approach has a higher success rate for generating feasible plans within an imposed planning time, with comparable plan quality (i.e. comparable number of PoI monitored) within a given plan horizon.

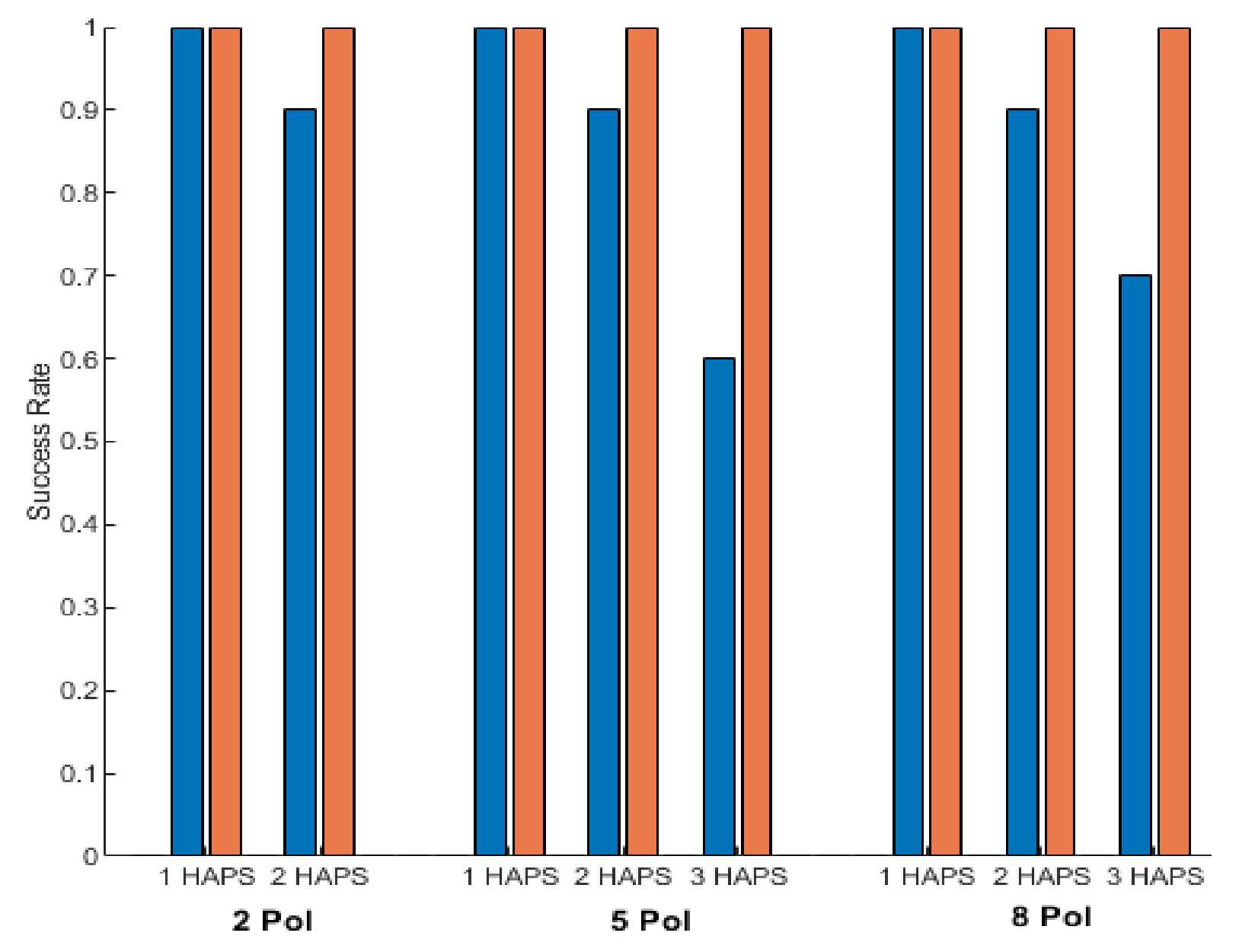


Fig. 6 Success rate: generation of feasible plans within the limited planning time

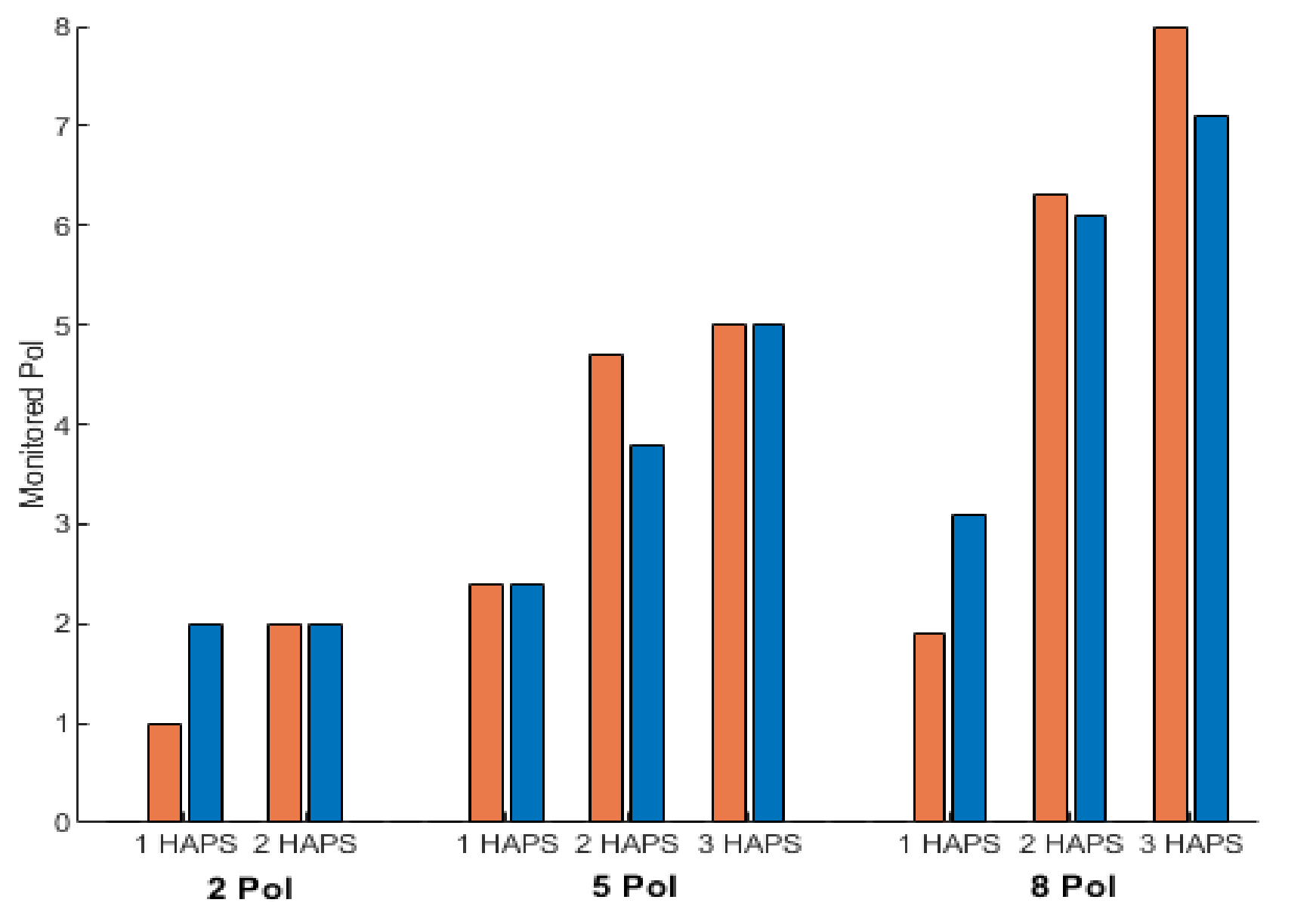


Fig. 7 Plan Quality: number of PoI monitored within a fixed plan horizon

This is because the sequential planning approach tends to, in tougher environments, suffer from an iterative "rejection" of task plans, i.e. the planning fails to generate feasible plans at the higher task planning level, which will be noticed later and rejected by the lower level numeric motion planner.
The integrated framework, on the other hand, benefits from the "anytime" implementation, and is therefore more robust in tougher environments.

4

Conclusion and Future Work

This work has exhibited the advantages of combining search spaces of different granularity levels to solve a complex real-world planning problem. As future work, the framework can be made more generic, in order to be aligned with the "domain-independent" nature of the PDDL+ planner (i.e. ENHSP). This can be done by

- Including an interface to the framework for defining domain-dependent meta-heuristics;
- Using a domain-independent "stitching" method while improving the plan locally by "removing" and "inserting" a new task assignment.