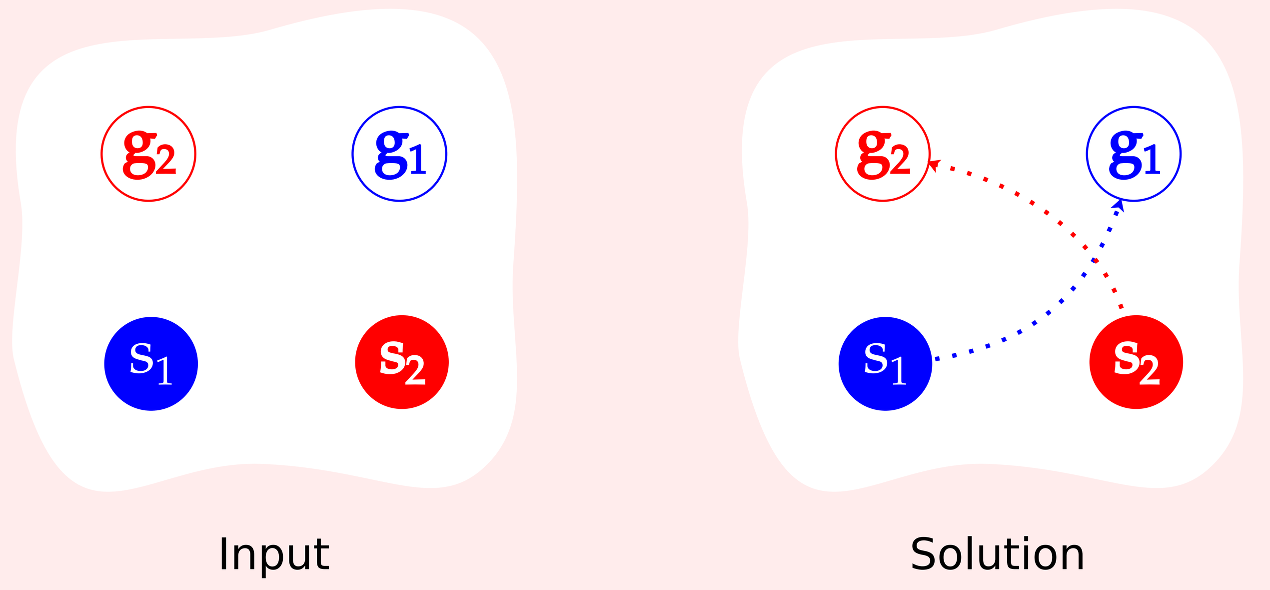
Finding Near-optimal Solutions in Multi-robot Trajectory Planning

Michal Čáp (CTU in Prague), **Peter Novák** (TU Delft), **Alexander Kleiner** (iRobot Inc.)

Problem

Let us have an environment with obstacles occupied by *n* robots. Given a start position, destination position and a cost function for each robot, find the trajectories for the robots that are mutually collision-free and the sum of their cost is minimal.



Even the non-optimal formulation is NP-hard [1].

k-step Penalty Method

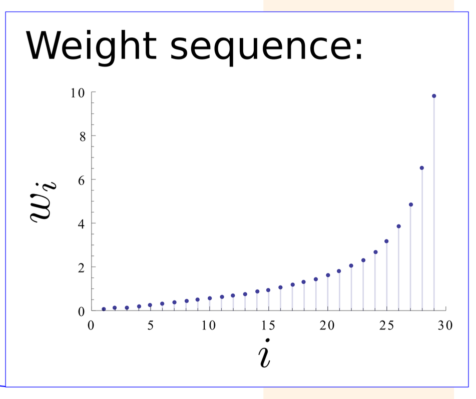
Inspired by a distributed optimization approach [2] previously applied to the multi-robot Rendezvous problem.

Starts from individually optimal trajectory for each robot, which are then penalized for being in collision with other robots. The penalty is gradually increased and the individual trajectories are iteratively replanned to account for the increased penalty until a collision-free solution is found.

Algorithm 1: k-step Penalty Method

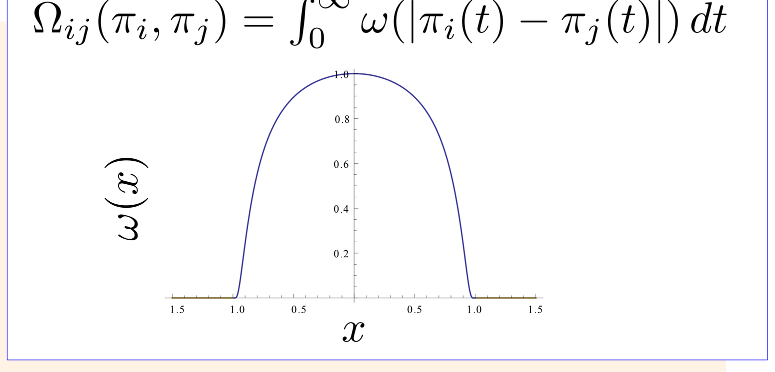
Algorithm PM (k)
  for $i \leftarrow 1 \ldots n$ **do** $\pi_i \leftarrow$ Replan $(i, 0)$
  for $i \leftarrow 1 \ldots n(k-2)$ **do**
    $r \leftarrow i \bmod n(k-2)$
    $w_i \leftarrow \tan(\frac{i}{n(k-2)+1} \cdot \frac{\pi}{2})$
    $\pi_i \leftarrow$ Replan (i, w_i)
  for $i \leftarrow 1 \ldots n$ **do** $\pi_i \leftarrow$ Replan (i, ∞)
  if $\forall_{ij}\ \Omega_{ij}(\pi_i, \pi_j) = 0$ **then return** $\langle \pi_1, \ldots, \pi_n \rangle$
  else report failure
Function Replan (r, w)
  return trajectory π for robot r that minimizes $c(\pi) + w \sum_{j \neq r} \Omega_{rj}(\pi, \pi_j)$

Weight sequence:



Penalty function:

$$\Omega_{ij}(\pi_i, \pi_j) = \int_0^\infty \omega(|\pi_i(t) - \pi_j(t)|)\, dt$$



Acknowledgements

This work was supported by the Grant Agency of the Czech Technical University in Prague grant SGS15/160/OHK3/2T/13.

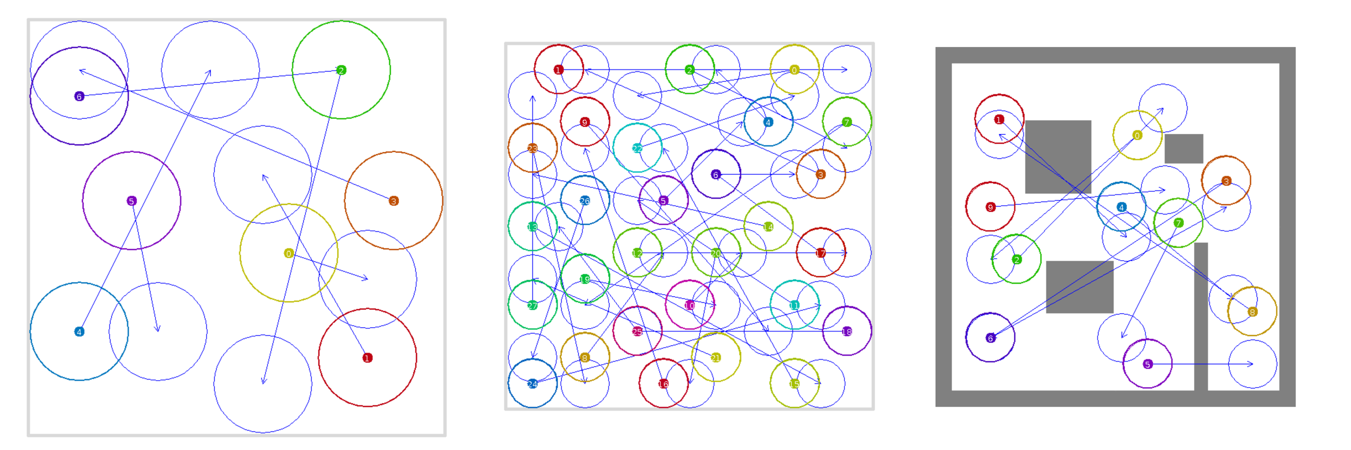
References

[1] Paul G. Spirakis and Chee-Keng Yap. Strong np-hardness of moving many discs. Inf. Process. Lett., 19(1):55–59, 1984.
[2] Subhrajit Bhattacharya, Vijay Kumar, and Maxim Likhachev. Distributed optimization with pairwise constraints and its application to multi-robot path planning. RSS 2010
[3] Standley, Trevor Scott, "Finding Optimal Solutions to Cooperative Pathfinding Problems.", AAAI Press (2010).
[4] Van Den Berg, Jur and Guy, Stephen and Lin, Ming and Manocha, Dinesh, "Reciprocal n-body collision avoidance", Robotics Research (2011).

Experimental Evaluation

Scenarios:

We focus on dense instances because they require search in a high-dimensional joint state space and are therefore hard for the existing optimal methods. Sparse instances, on the other hand, can be usually split into independent low-dimensional subconflicts and efficiently solved separately.



Scenario A   Scenario B   Scenario C

Compared algorithms:

k=3 k=20 k=100 **PM** **k-step Penalty Method** (proposed approach)

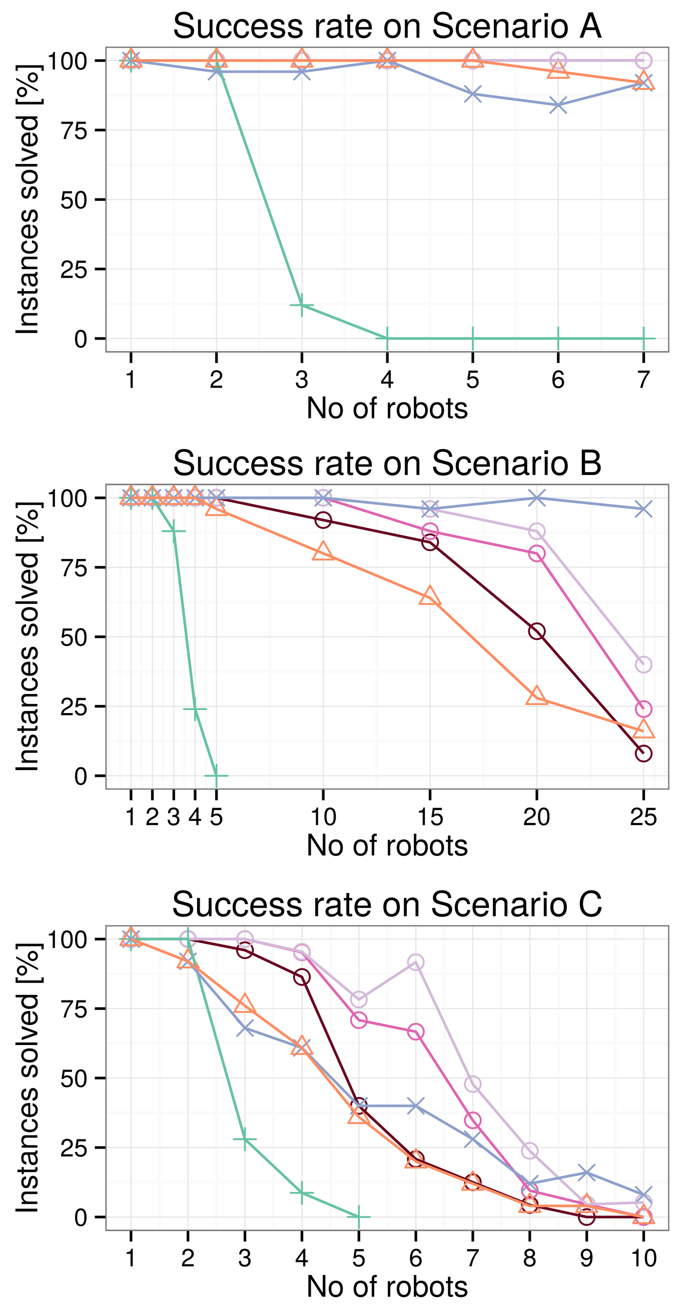
PP **Prioritized Planning** (popular heuristic approach)

OD **Operator Decomposition** [3] (state-of-the-art optimal algorithm)

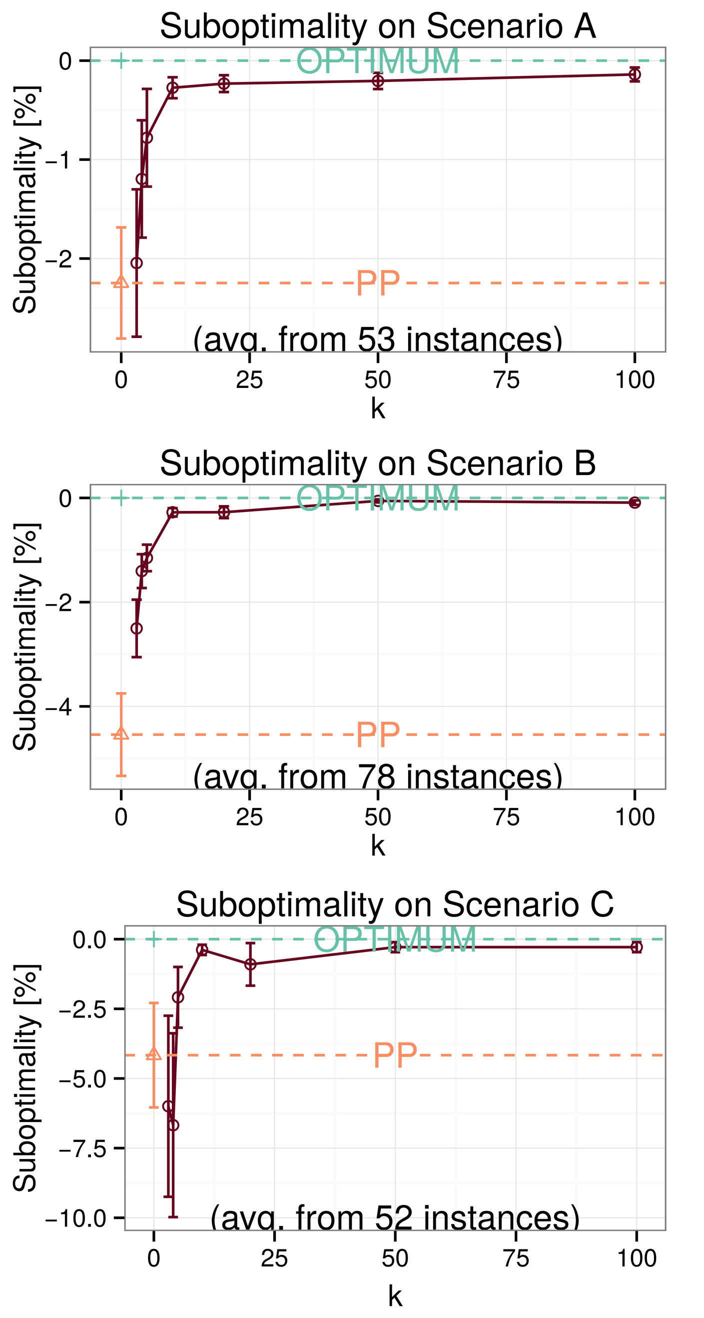
ORCA **Optimal Reciprocal Collision Avoidance** [4] (state-of-the-art reactive algorithm)

Results:

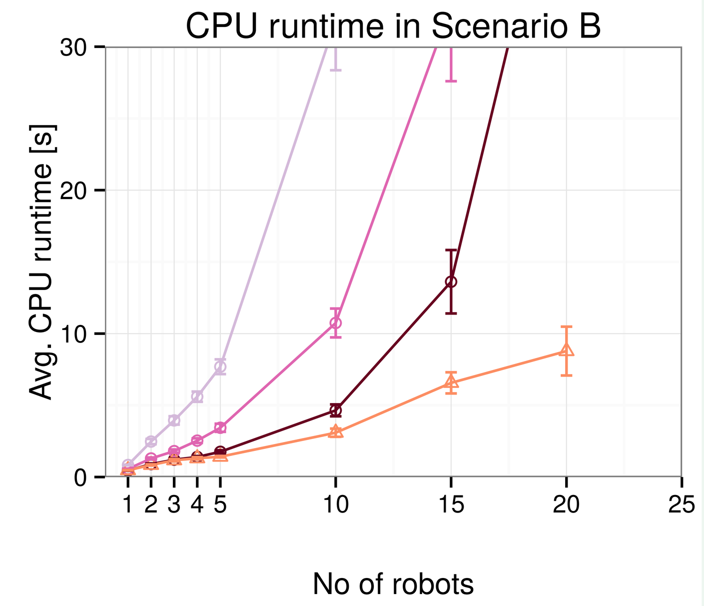
Success rate: the percentage of instances successfully solved by PM(k=3, 20, 10), PP, OD, and ORCA in each scenario in max 1 hour.



Suboptimality: average suboptimality of solution generated by PM(k=1, ..., 100) and PP on instances where optimum was known.



CPU runtime: average CPU runtime to find a solution by PM(k=3, 20, 10) and PP.



Time out of goal: average difference in solution quality generated by PM(k=1, ..., 100), PP, and ORCA on instances with 10 robots in Scenario B.