A distributed approach to holonomic path planning

Andreagiovanni Reina[+], Gianni A. Di Caro[*], Frederick Ducatelle[*] and Luca M. Gambardella[*]

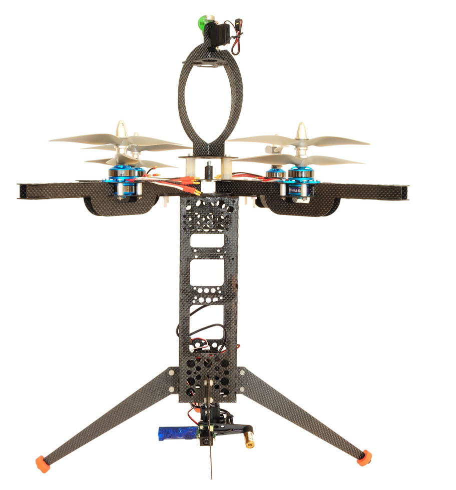
[+]Department of Electronics and Information (DEI), Politecnico di Milano
Piazza L. da Vinci, 32 - 20133 Milano, Italy
andreagiovanni.reina@mail.polimi.it

[*]Dalle Molle Institute for Artificial Intelligence Studies (IDSIA)
Galleria 2, 6928 Manno - Lugano, Switzerland
{gianni, frederick, luca}@idsia.ch

Abstract

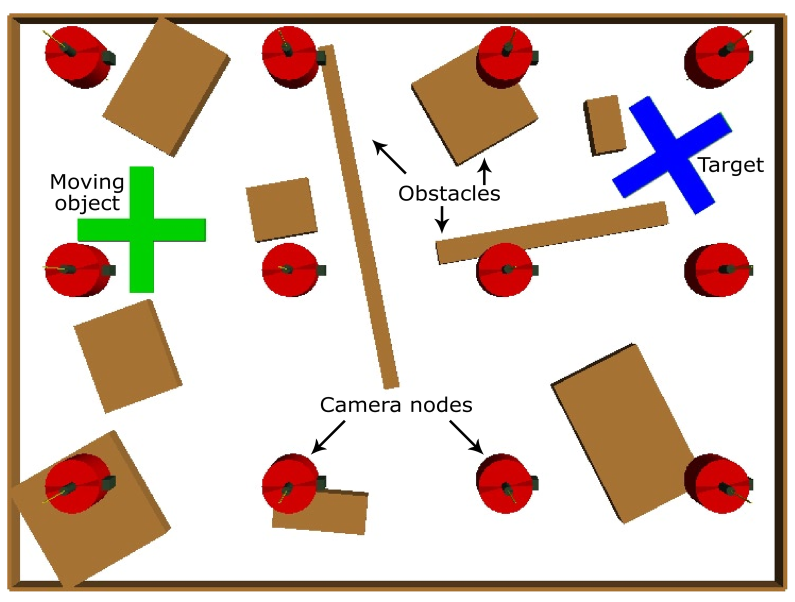
We study a distributed approach to the path planning problem. We focus on holonomic kinematic motion in a plane with static obstacles. The problem consists in planning the path of a rigid object of arbitrary shape that has to be transported from an initial to a final location through a constrained path. The planner observes the environment from above through a visual system. We consider the case in which the path covers a large area, such that the planner architecture consists of a wireless network of observer nodes which each can see a portion of the area. A centralized solution is neither robust nor scalable. To overcome these difficulties we propose a fully distributed approach: each observer node locally calculates the part of the path relative to the area that it sees, and communicates to its neighbors the information which permits the cooperative execution of the planning. Our goal is to calculate effective paths in a way that is scalable, resource efficient, and robust to calibration and alignment errors.

1 The robot model

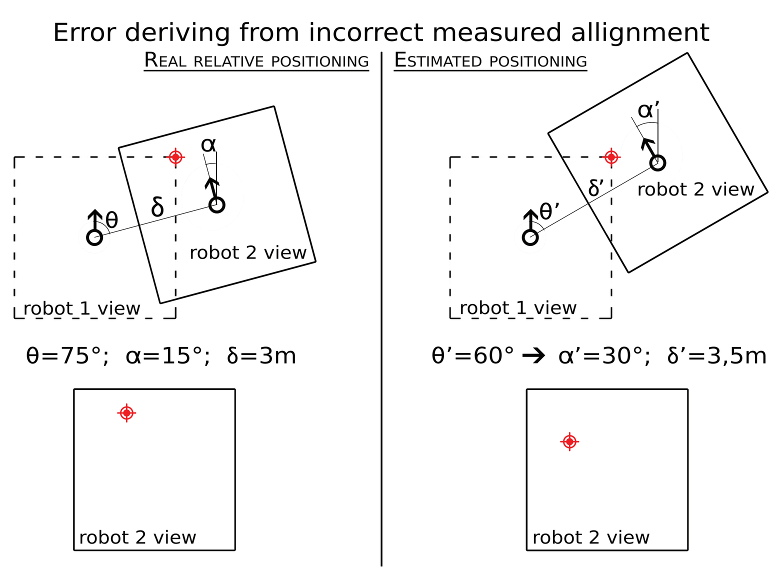


As reference models for the planner nodes, we consider the *eye-bot* robots, that are being developed in the Swarmanoid project (*http://www.swarmanoid.org*). These are small flying robots that can attach to the ceiling, are equipped with a video camera pointing to the floor, and have an infrared system for measuring the relative bearing and distance between two robots and for wireless communications.

2 Problem description

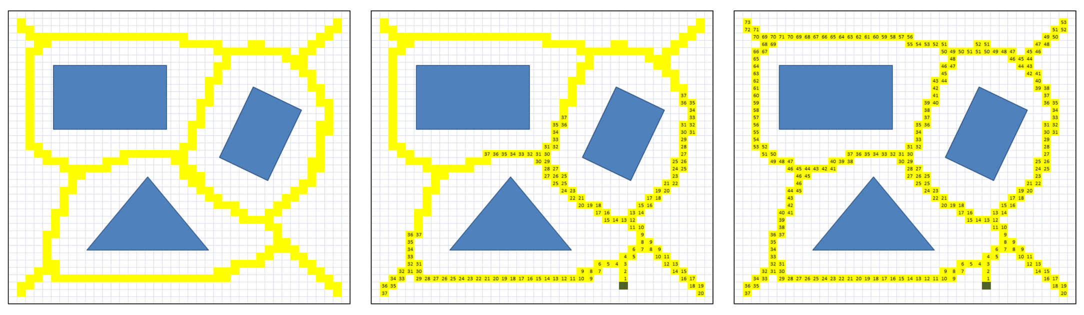


Eye-bots attach to the ceiling and cover the entire area between the start and the goal. Each robot has a limited vision of the environment and we assume that the partial view of neighbor robots overlaps, in order to permit the sharing of the rigid object position. However, the overlapping is subject to errors deriving from errors in camera calibrations and in the measure of robots' relative positioning.

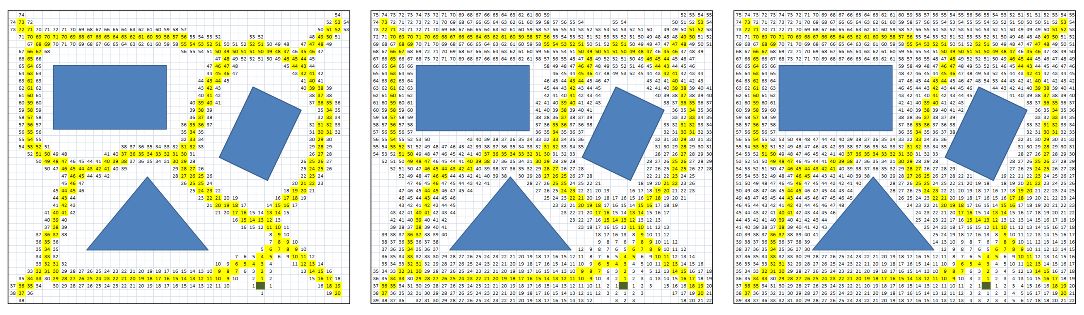


3 Path planning

Our distributed planner is derived from the numerical potential field technique for a single camera planner (Latombe et al., 1991; Choset et al., 2005) which is computed using the *wave front expansion with skeleton* on a bidimensional uniform cell partitioning. This solution first spreads the potential over a subset of the free space, called *skeleton*, which corresponds to the *Voronoi diagram*; then the potential is computed in the rest of the map. The potential descent is performed using A*.



Potential field expansion over the skeleton.



Diffusion of the potential field over the remaining free space.

4 Distributed path planning

The distributed algorithm has three phases:

1. ***Neighbor detection***. Each robot builds a neighbor table, in which the relative positions of nearby nodes are stored with some estimation error.
2. ***Potential field diffusion***. Each robot expands the potential field on its part of the map, and sends to its neighbors the frontier values. The system minimizes the communication overhead, transmitting only the information of the skeleton cells near the frontier.
3. ***Path calculation***. The robot above the start position begins path calculation. When the trajectory exits from its area of view, the robot sends the object coordinates to a selected neighbor. Then the process iterates from robot to robot until the target position is reached.

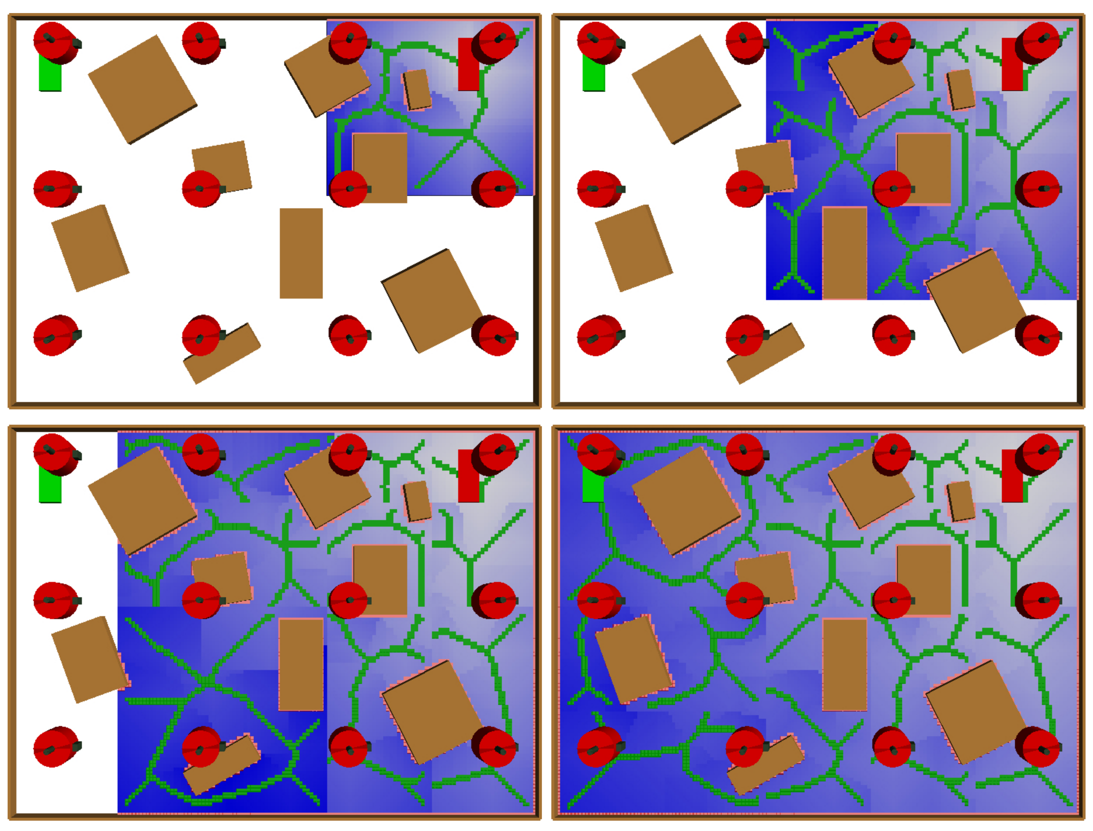
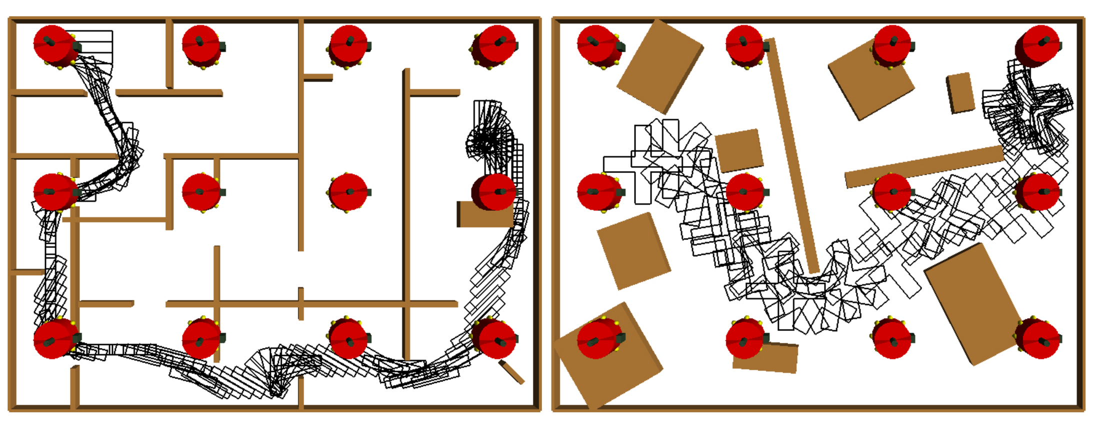


Illustration of the potential field diffusion phase.



Examples of path trajectories.

4.1 Heuristics to escape local minima and loops

Given the distributed nature of the approach, the same local minimum in the potential field can negatively affect the path calculation phase of multiple robots. Moreover, the calculated path can present loops due to distributed sub-path composition. To overcome these issues, which can cause an overhead in computational and communication resource usage, we propose three heuristics:

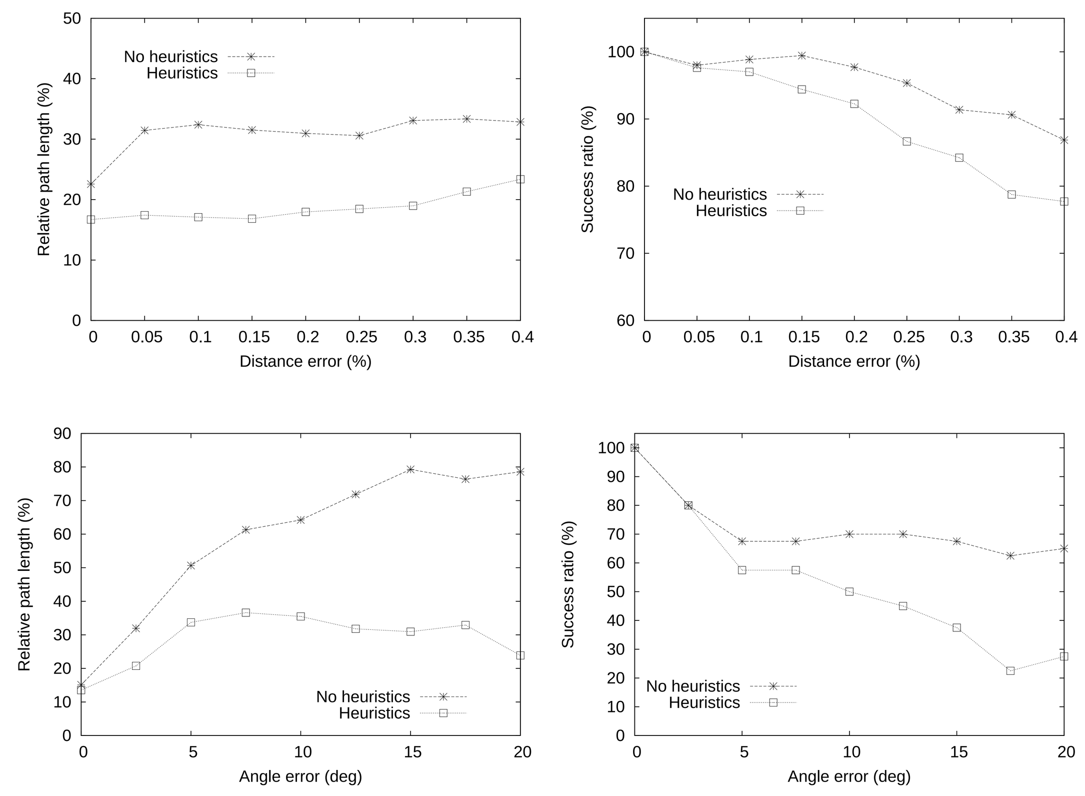
- *Smart Loop Avoidance*: during path calculation a node can detect loops. In this case, it coordinates with all involved nodes to retrace the search back to the beginning of the loop.
- *Skeleton Pruning*: pre-pruning of the skeleton during the potential field diffusion, to block passages that would not let the object passing through.
- *Narrow Passage Detection*: during path calculation a node can detect a narrow passage, block it, and then repeat the potential field diffusion step.

5 Experimental results

Through simulation, we tested our approach in a set of 20 sample scenarios differing in obstacle structure and object shape.

5.1 Effect of heuristics vs. position errors

We studied the effect of the relative positioning error between nodes in terms of angle and distance error for the algorithm with and without the heuristics. The results for different error values are shown in the plots below. In all the experiments the camera nodes maintain a 4x3 grid formation.



Performance vs. errors. The error values on the x-axis indicate the standard deviation of a zero mean Gaussian distribution used to sample the distance/angle error between the robot pairs. For each scenario we ran 40 trials. Each data point represents the average of 20x40 experiments.

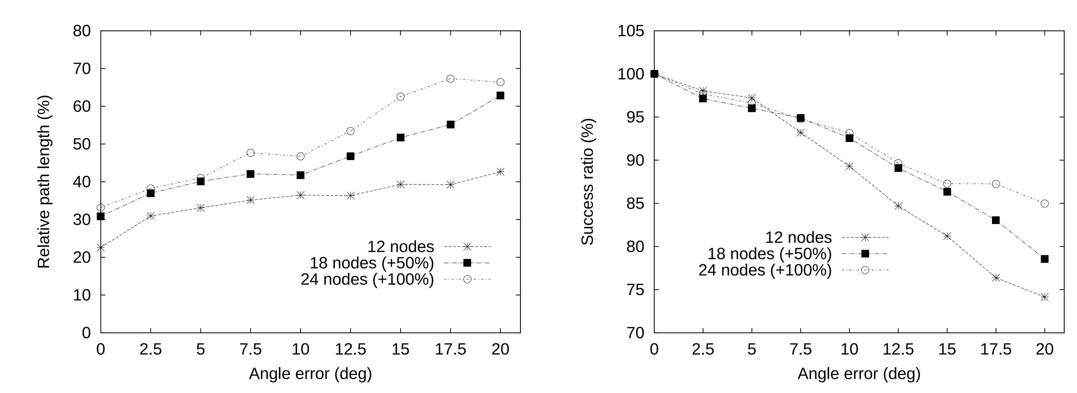
For relatively low errors the performance is always very close to the reference. While for increasing errors:

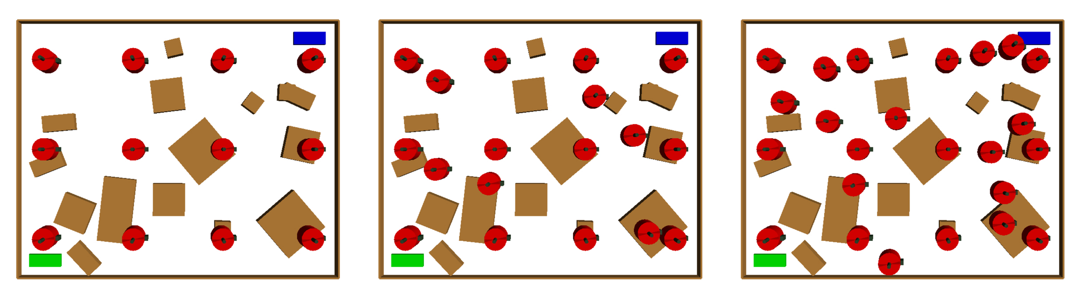
- The algorithm with heuristics degrades rapidly in terms of success rate but slowly for path quality.
- The algorithm without heuristics behaves in opposite way.

In both cases, the system is relatively sensitive to errors on the angle, while it is quite robust to distance errors.

5.2 Scalability performances

In a distributed approach the imprecision in potential diffusion necessarily increases with the number of nodes. We study the effect on performance of increasing the density of the nodes over a fixed area varying angle errors.





12 robots | 18 robots [+50%] | 24 robots [+100%]

The increase in node number negatively affects performance. However, node redundancy can at the same time counterbalance the increase in the relative positioning error, as is well evidenced by the success rate plot.

References

Latombe, Jean-Claude. *Robot Motion Planning*. Kluwer Academic Publishers. Norwell, MA, USA, 1991.

H. Choset, K. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. Kavraki and S. Thrun. *Principles of Robot Motion: Theory, Algorithms, and Implementations*. MIT Press, Cambridge, 2005.

Acknowledgments This work has been partially supported by the *Zeno Karl Schindler Foundation* (Genève, Switzerland), and by the *SWARMANOID* FET project, funded by the European Commission.