International Academy for Production Engineering
7th CIRP Conference on Assembly Technologies And Systems
Tianjin, China
May 10 - 12, 2018

Distributed Motion Planning for Industrial Random Bin Picking

by

Vojtěch Vonásek[a], Axel Vick[b] and Jörg Krüger[c]

[a]Czech Technical University in Prague, Faculty of Electrical Engineering, Prague, Czech Republic
[b]Fraunhofer Institute for Production Systems and Design Technology (IPK), Berlin, Germany
[c]Department of Industrial Automation Technology (IAT), Technische Universität Berlin, Germany

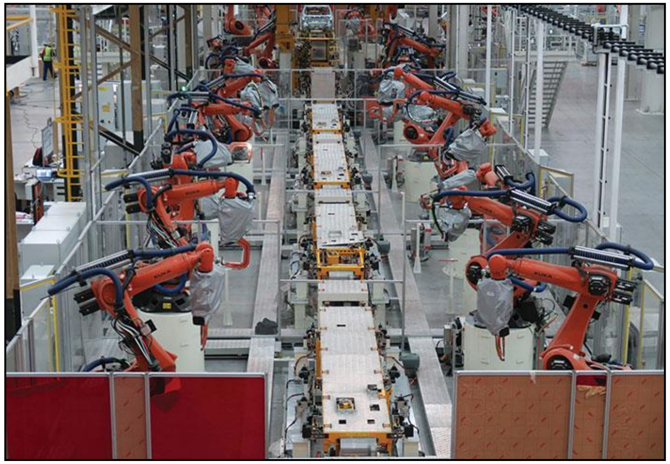
www.ipk.fraunhofer.de
www.iat.tu-berlin.de

Outline

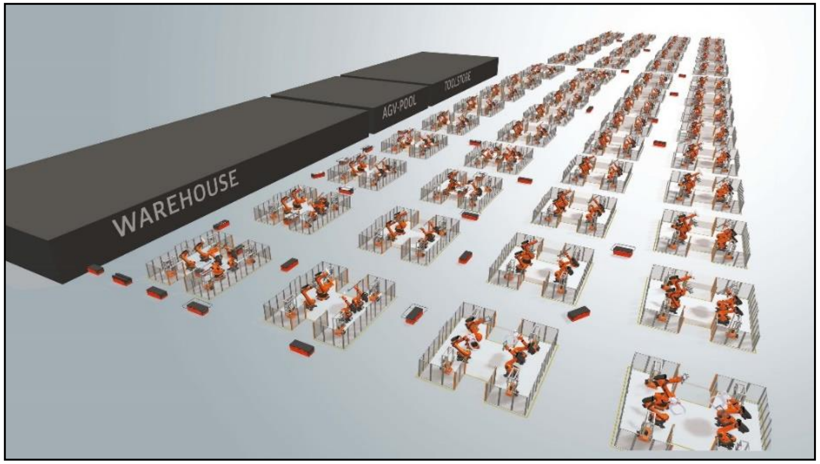
1. Introduction and Motivation
2. The Bin Picking Problem
3. Robot Path Planning Methods
4. Distributed System Architecture
5. Experimental Setup
6. Results and Discussion

Introduction and Motivation

- Manufacturing Evolution from Line to Matrix
  - Lot Size 1 Production in Robotic Cells
  - Loading of Machine Tools or Mobile Robots
  - Handling of Parts in changing Environment



© Volkswagen AG



© Kuka AG

Introduction and Motivation

- Human Robot Collaboration Trend
  - Manipulating material and products either by humans or by robots results in unknown position, orientation, configuration and other properties of the parts, even if put to a predefined zone, like swap areas or bins.
  - Placing and picking parts to and from bins while considering changing environment will be a regular task for industrial robots in the near future.
- Industrial Bin Picking demands for fast online Motion Planning

The Bin Picking Problem

- A bin contains known objects, that need to be removed from that bin.
- Tasks and Methods
  - Find object (Computer Vision) [1,2]
  - Move to pick point (Motion Planning) [4]
  - Grasp object (Gripper Design) [3]
  - Move object to place point (Motion Planning) [4]
- Motion Planning Task needs to generate a collision-free trajectory for robot and object in a minimum of time.

Robot Path Planning Methods

- Probabilistic Roadmaps (PRM) [5]
  - Sampling of 6-DOF configuration space
  - Store samples that are collision-free in workspace
  - Connect samples with k-nearest neighbor to roadmap
  - Find shortest path in roadmap to generate trajectory

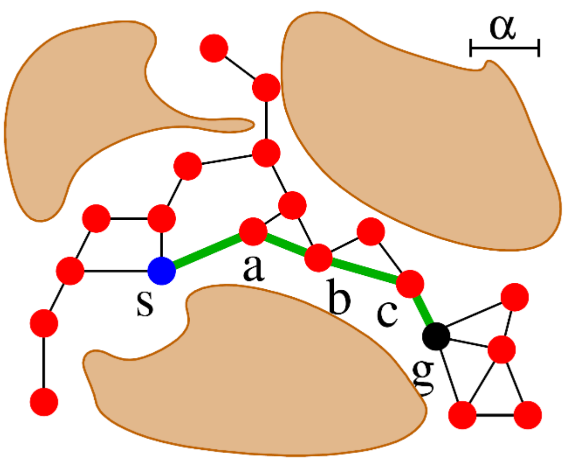


Fig 3: Schematic Robot Configuration Space with Obstacles and PRM

Robot Path Planning Methods

- Rapidly exploring Random Trees (RRT) [6]
  - Generate random sample in configuration space
  - Find nearest node in (initial) tree and expand it
    - Generate n samples around node configuration
    - Keep collision-free samples
    - Add nearest one to random sample to tree
  - Terminate on approaching goal configuration

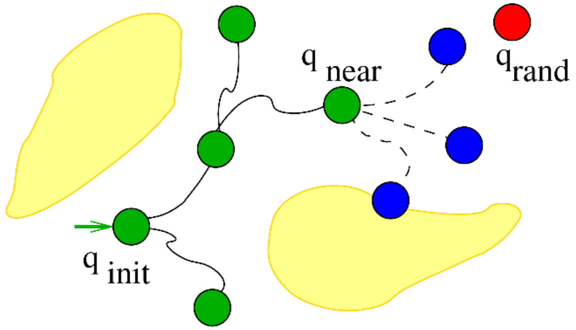
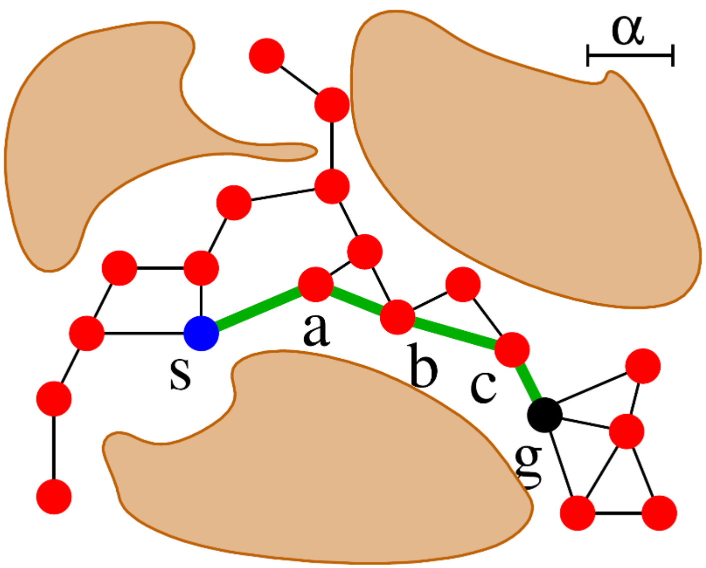


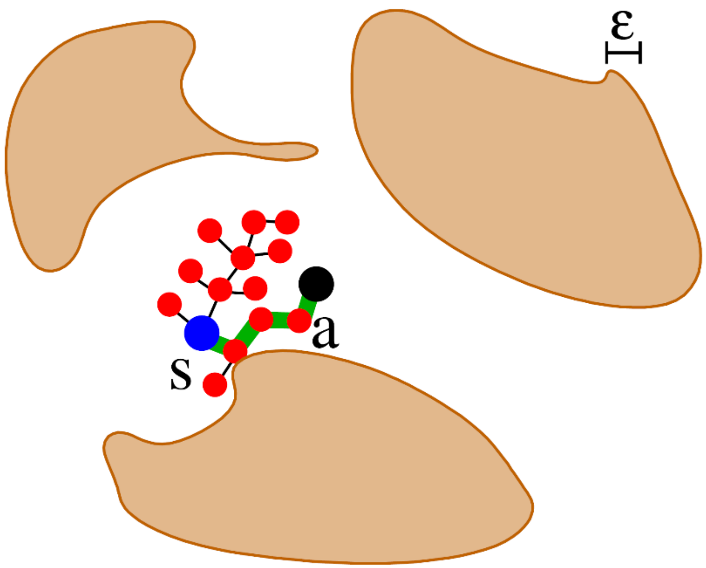
Fig 4: Schematic Robot Configuration Space with Obstacles and RRT

Robot Path Planning Methods

- Two Stage Planning Approach



1. PRM



2. RRT

Distributed System Architecture

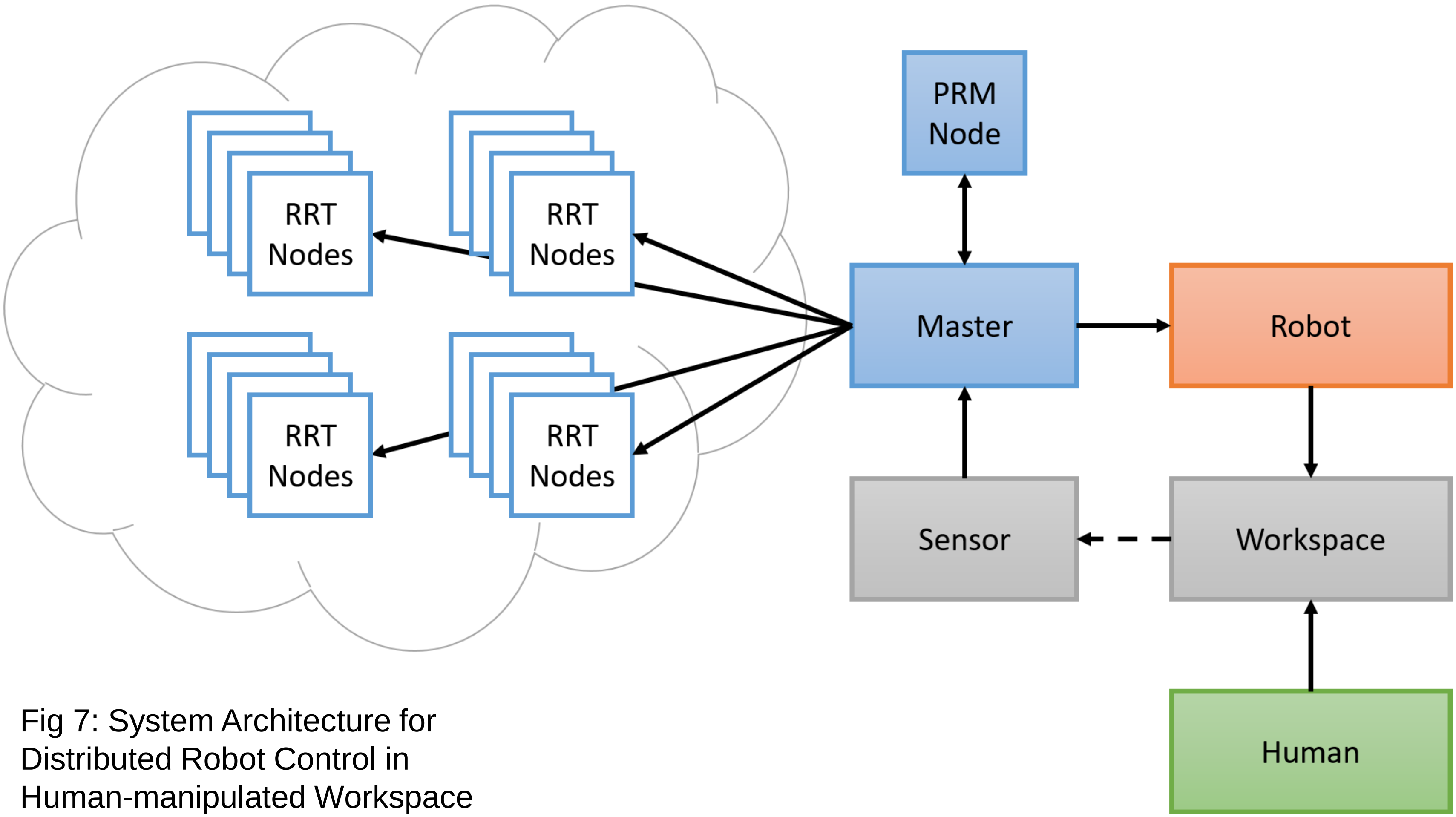


Fig 7: System Architecture for Distributed Robot Control in Human-manipulated Workspace

Distributed System Architecture

- Split path to coarse waypoints with PRM.
- Find collision-free fine paths between waypoints with RRT in parallel.
- Master node distributes tasks to PRM and RRT services and joins the final fine trajectories.
- If RRT services are more than waypoints then
  - Fine RRT paths are requested multiple times and the first planner terminating wins. (OR strategy)
  - New PRM waypoints are inserted in the longest edge.

Computation and Communication Infrastructure

Laptop
(4 threads)

Server / Private Cloud
(40 threads)

Public Cloud
(> 1000 threads)

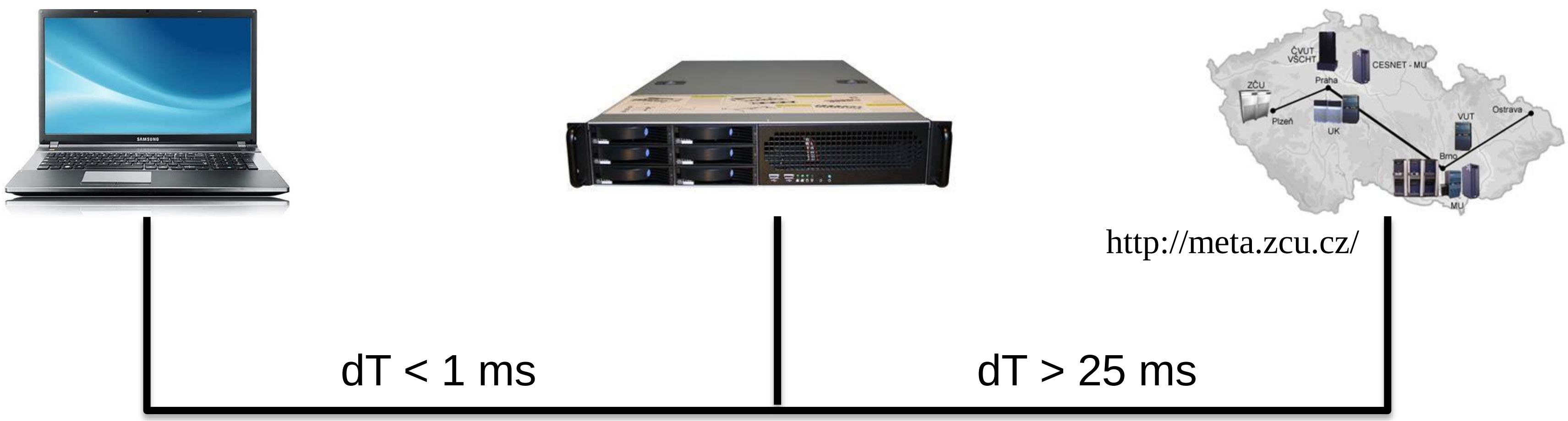


Fig 8: Computing Power vs. Latency

Computation and Communication Infrastructure

- Hypothesis
  - Computation of all services on one device (Laptop) has lowest Latency but lowest Performance
  - Distributing RRT services to Private Cloud has low Latency and high Performance
  - Distributing RRT services to Public Cloud has high Latency and medium Performance
- Infrastructure Choice depends on Availability and Scalability

Experimental Setup

- Motion Planning between Start/Goal Positions

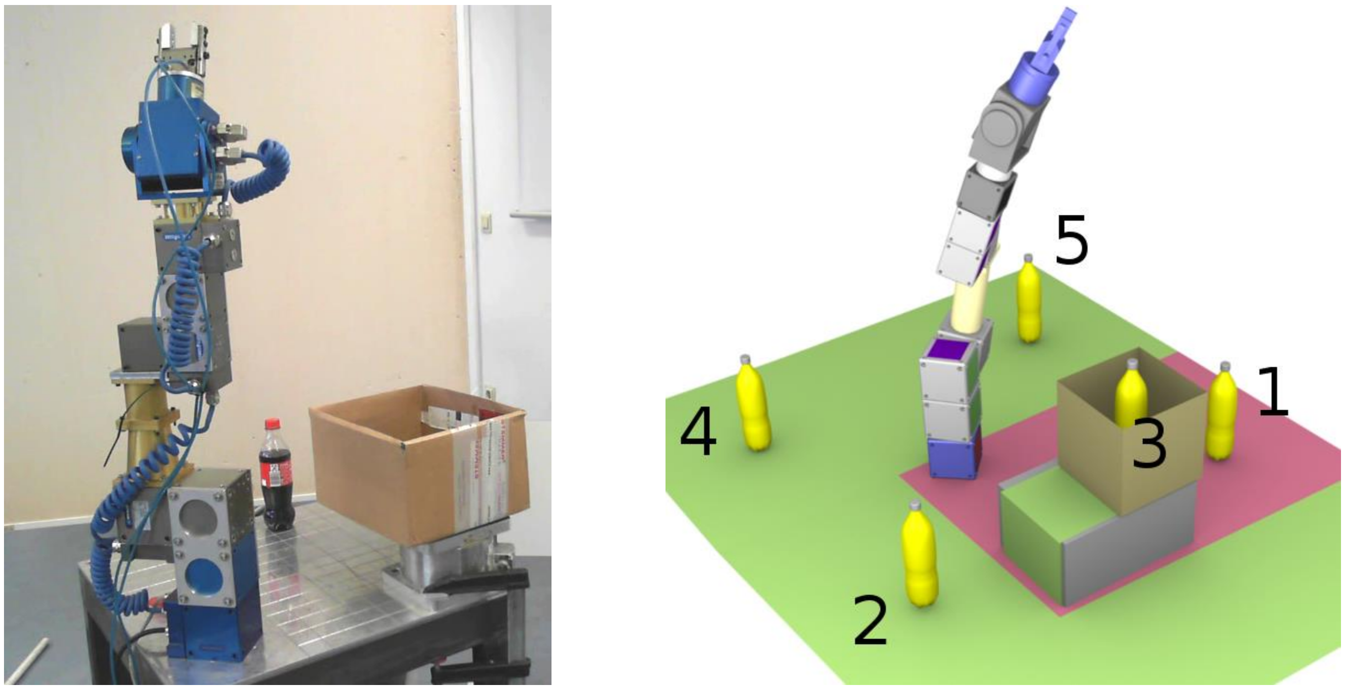


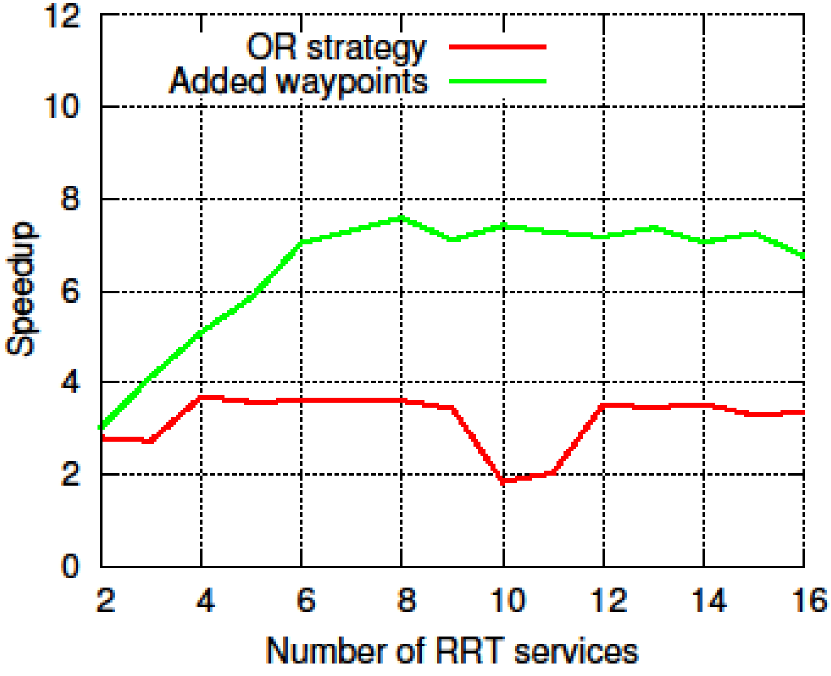
Fig 9: Modular Robot AMTEC in Laboratory and Blender Model

Experimental Setup

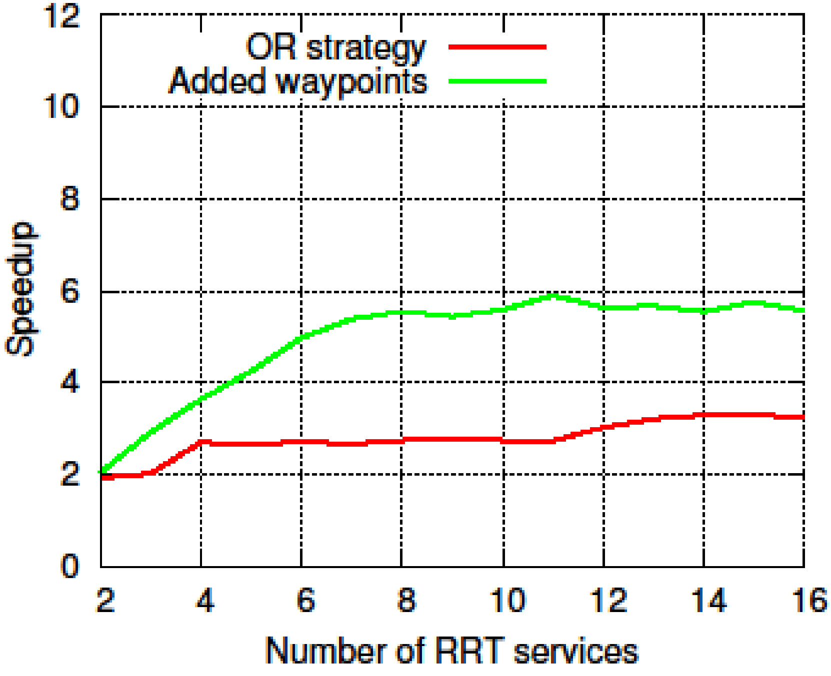
- Verification with 6-DOF Amtec robot
- 5 start/goal configurations
- Lazy-PRM [22] with resolution α = 1cm
- Task-space RRT [24] with resolution ε = 0.1cm
- Computation on virtual machines (VM)
  - 1 VM with 16 cores (centralized)
  - 16 VMs with 1 core (distributed)
- Latency between Master and RRT nodes
  - 28 ms (std 0.12ms)

Results and Discussion

- G1 → G2, Speedup gain to local computation

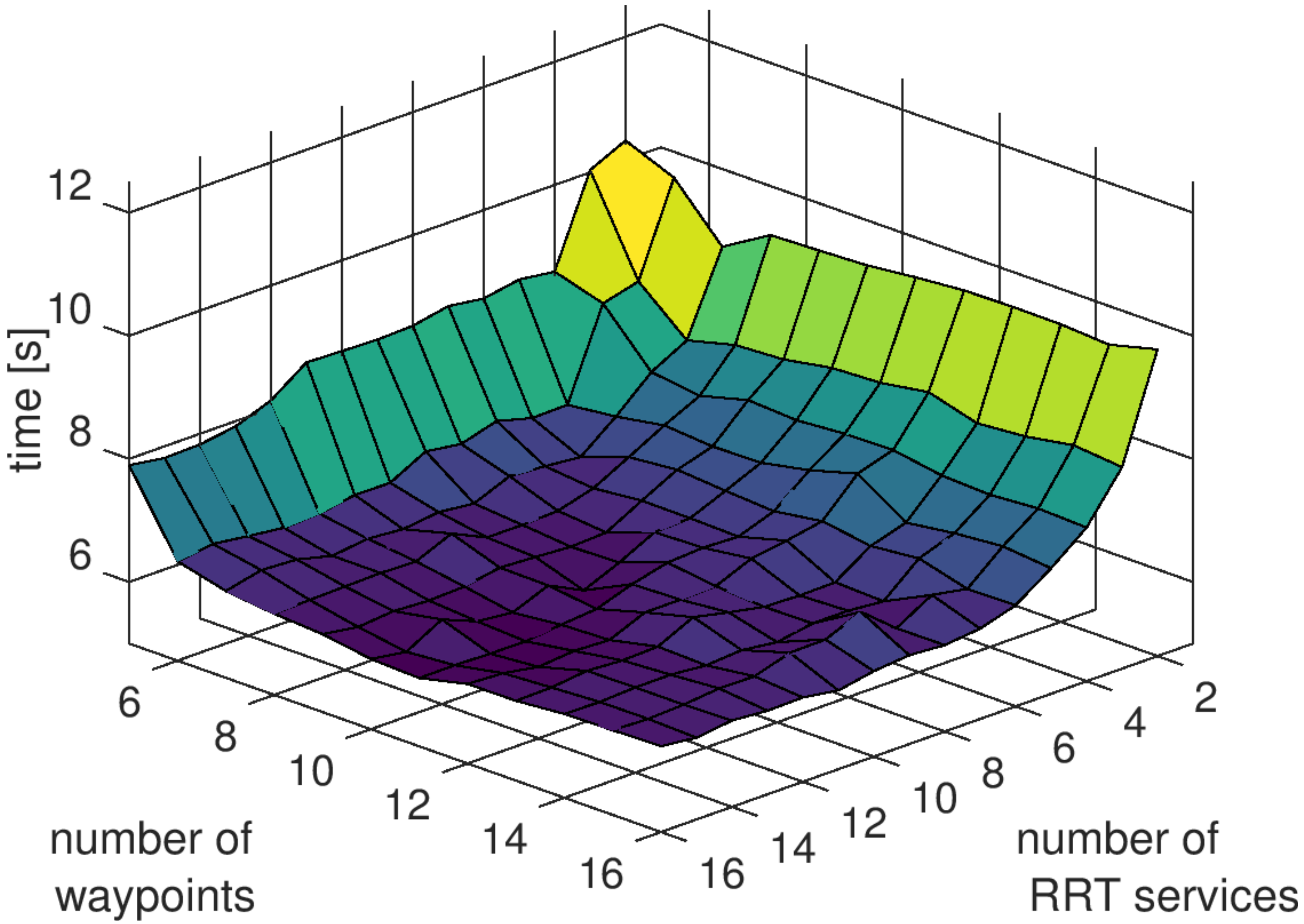


1 VM, 16 cores



16 VMs, 1 core

Results and Discussion



Results and Discussion

- High complexity tasks benefit from parallelization
  - The configuration space of the robot, that is handling a non-trivial object through a variable environment, is highly restricted when navigating in the workspace.
  - The presented approach is able to react to the challenging and additionally changing conditions in a 2-6 times faster way than classical methods.
  - The use of a distributed computing infrastructure, like virtual machines in the cloud, is feasible even with moderate latency connections.

Future Work

- Edge Cloud Control with Low Latency Networks
  - Factory Logistics (AGV) Use Case
  - Private/Edge Cloud Infrastructure
  - 4G/5G Mobile Communication
  - Centralized Path Planning Services
- Real-time Closed Loop Feedback Control in Distributed Control Systems
  - Using Model-predictive Control to handle variable Latency for Linear Systems

Future Work

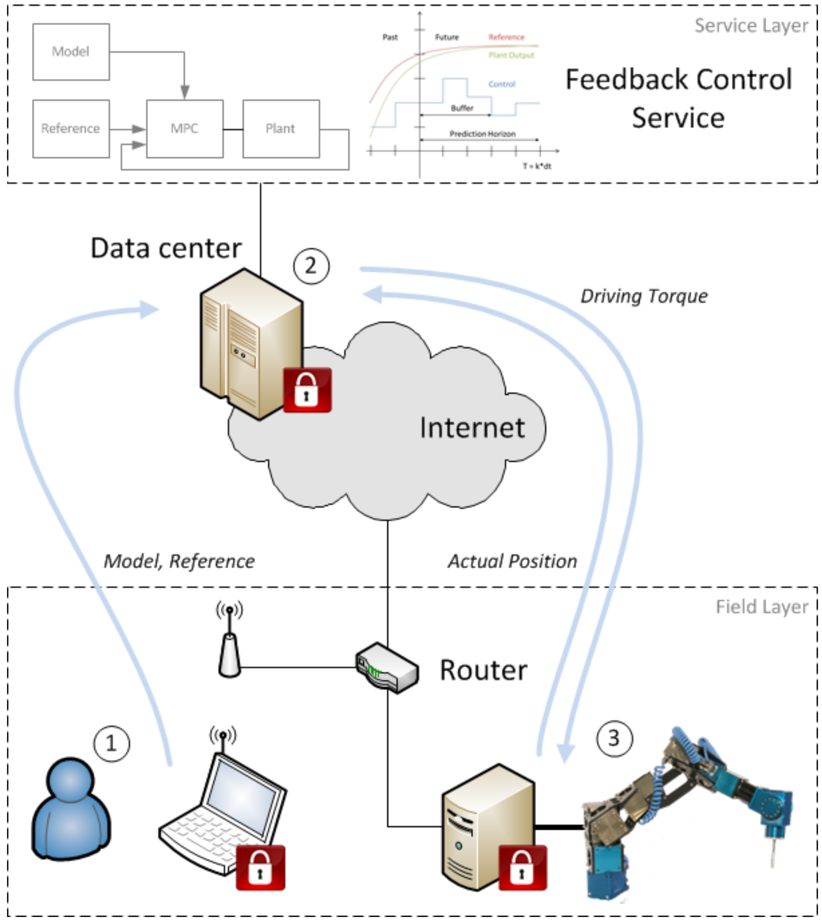


Fig 12: Scheme of cloud-based feedback control for industrial robots

References

[1] C. Papazov, S. Haddadin, S. Parusel, K. Krieger, and D. Burschka. Rigid 3D geometry matching for grasping of known objects in cluttered scenes. The International Journal of Robotics Research, 31(4):538–553, 2012.
[2] H.-Y. Kuo, H.-R. Su, S.-H. Lai, and C.-C.Wu. 3D object detection and pose estimation from depth image for robotic bin picking. In IEEE International Conference on Automation science and engineering (CASE), pages 1264–1269, 2014.
[3] K. Yamazaki, M. Tomono, T. Tsubouchi, and S. i. Yuta. A grasp planning for picking up an unknown object for a mobile manipulator. In IEEE International Conference on Robotics and Automation (ICRA), 2006.
[4] D. Buchholz, D. Kubus, I. Weidauer, A. Scholz, and F. M. Wahl. Combining visual and inertial features for e cient grasping and bin-picking. In IEEE international conference on robotics and automation (ICRA), 2014.
[5] S. M. LaValle. Rapidly-exploring random trees: A new tool for path planning, 1998. Technical report 98-11.
[6] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. IEEE Transactions on Robotics and Automation, 12:566–580, 1996.
[7] Z. Sang and X. Xu. The framework of a cloud-based CNC system. In Manufacturing Systems 4.0 - Proceedings of the 50th CIRP Conference on Manufacturing Systems, volume 63, pages 82 – 88, 2017.
[8] A. Vick, V. Vonasek, R. Penicka, and J. Krüger. Robot control as a service - towards cloud-based motion planning and control for industrial robots. In International Workshop on Robot Motion and Control (RoMoCo), pages 33–39, 2015.
[9] A. Vick, J. Guhl, and J. Kr¨uger. Model predictive control as a service – concept and architecture for use in cloud-based robot control. In 2016 21st International Conference on Methods and Models in Automation and Robotics (MMAR), pages 607–612, Aug 2016.
[10] C. A. Niebuhr and A. Vick. Distributed real-time control service framework for human-robot interaction applications. In 2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA), pages 1–8, Sept 2016.
[11] L. Thames and D. Schaefer. Software-defined cloud manufacturing for industry 4.0. In The Sixth International Conference on Changeable, Agile, Reconfigurable and Virtual Production (CARV2016), volume 52, 2016.
[12] X. V. Wang, M. Givehchi, and L. Wang. Manufacturing system on the cloud: A case study on cloud-based process planning. In Manufacturing Systems 4.0 - Proceedings of the 50th CIRP Conference on Manufacturing Systems, volume 63, pages 39 – 45, 2017.
[13] M. Elbanhawi and M. Simic. Sampling-based robot motion planning: A review. IEEE Access, 2:56–77, 2014.
[14] N. M. Amato and L. K. Dale. Probabilistic roadmap methods are embarrassingly parallel. In IEEE International Conference on Robotics and Automation (ICRA), volume 1, pages 688–694. IEEE, 1999.
[15] D. Devaurs, T. Sim´eon, and J. Cort´es. Parallelizing RRT on distributed memory architectures. IEEE International Conference on Robotics and Automation (ICRA), pages 2261–2266, 2011.
[16] I. A. Sucan and L. E. Kavraki. A sampling-based tree planner for systems with complex dynamics. IEEE Transactions on Robotics, 28(1):116–131, Feb 2012.
[17] C. Park, J. Pan, and D. Manocha. Parallel motion planning using poissondisk sampling. IEEE Transactions on Robotics, 33(2):359–371, 2017.
[18] E. Plaku and L. E. Kavraki. Distributed sampling-based roadmap of trees for large-scale motion planning. In IEEE International Conference on Robotics and Automation (ICRA), pages 3868–3873. IEEE, 2005.
[19] S. A. Jacobs, K. Manavi, J. Burgos, J. Denny, S. Thomas, and N. M. Amato. A scalable method for parallelizing sampling-based motion planning algorithms. In IEEE International Conference on Robotics and Automation (ICRA), pages 2529–2536. IEEE, 2012.
[20] S. A. Jacobs, N. Stradford, C. Rodriguez, S. L. Thomas, and N. M. Amato. A scalable distributed RRT for motion planning. In IEEE International Conference on Robotics and Automation (ICRA), pages 5088–5095, 2013.
[21] C. Rodriguez, J. Denny, S. A. Jacobs, S. Thomas, and N. M. Amato. Blind RRT: A probabilistically complete distributed RRT. In IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2013.
[22] R. Bohlin and L. E. Kavraki. Path planning using Lazy PRM. In IEEE International Conference on Robotics and Automation (ICRA), volume 1, pages 521–528, 2000.
[23] S. M. LaValle and J. J. Kuner. Rapidly-Exploring Random Trees: Progress and Prospects. In Algorithmic and Computational Robotics: New Directions, pages 293–308, 2000.
[24] A. Shkolnik and R. Tedrake. Path planning in 1000+ dimensions using a task-space Voronoi bias. In IEEE international conference on Robotics and Automation (ICRA), pages 2892–2898, 2009.
[25] S. Gottschalk, M. C. Lin, and D. Manocha. OBBTree: a hierarchical structure for rapid interference detection. In Proceedings of the 23rd annual conference on Computer graphics and interactive techniques, 1996.
[26] A. Yershova and S. M. LaValle. Improving motion-planning algorithms by e cient nearest-neighbor searching. IEEE Transactions on Robotics, 23(1):151–157, 2007.

Thank You.