PICKNIK

Jenseits von RRTConnect

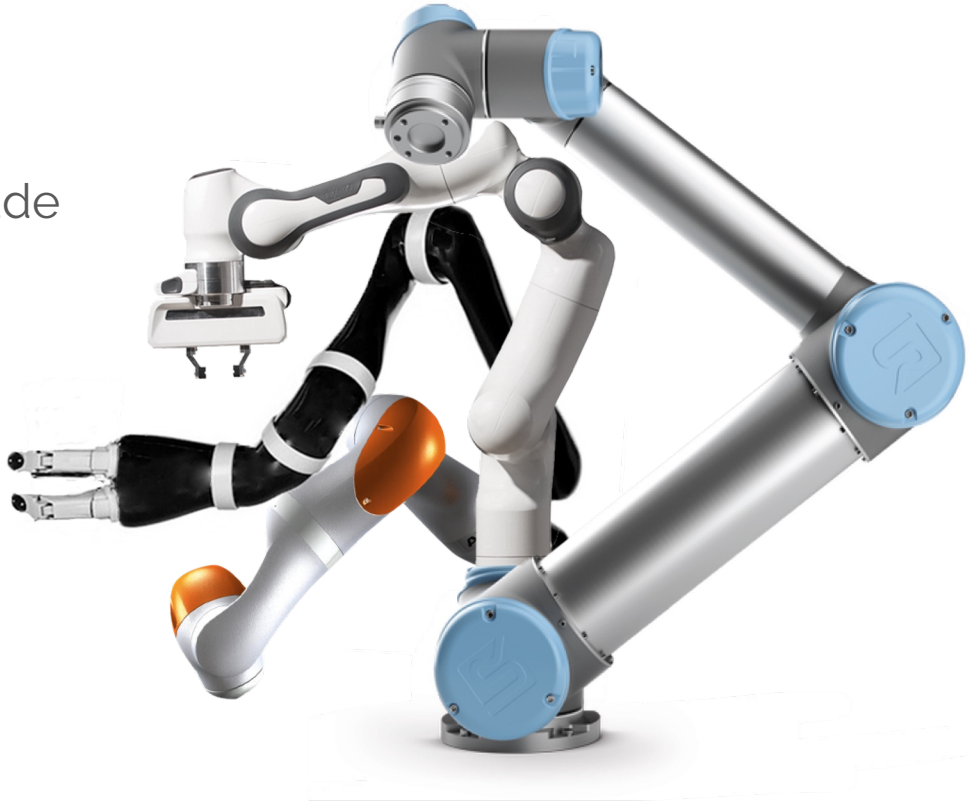
Plannerwahl in MoveIt 2

November, 2023

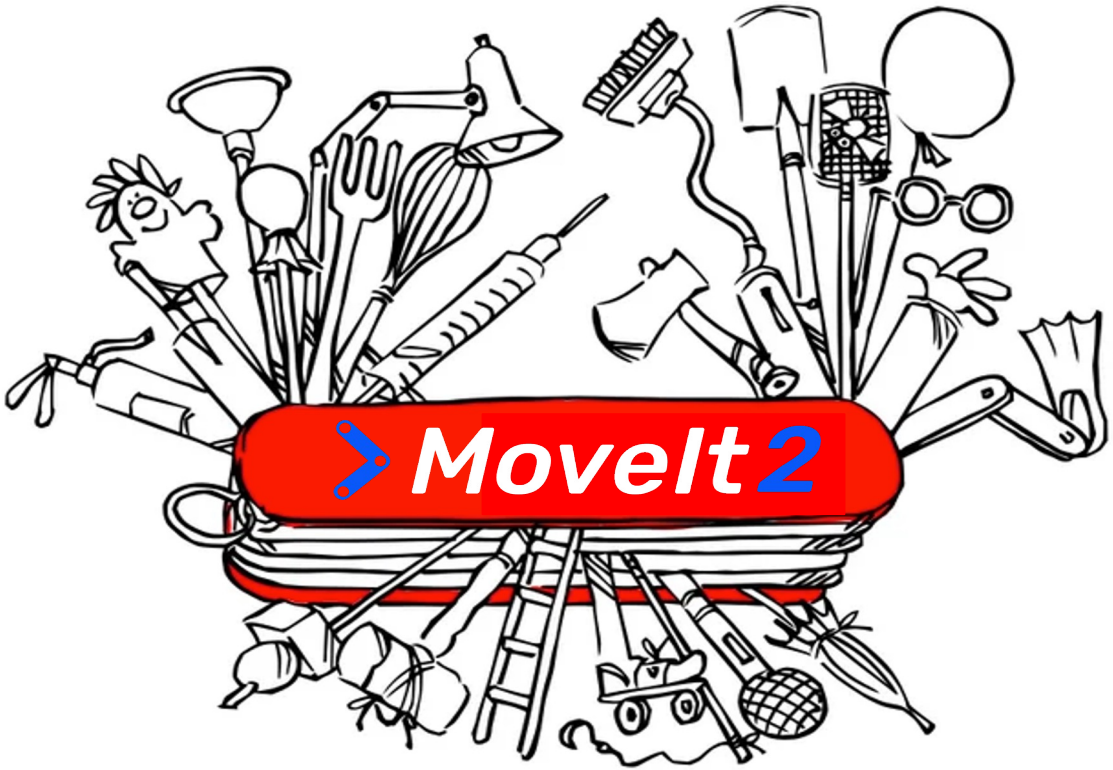
Sebastian Jahr
Robotics Engineer
sebastian.jahr@picknik.ai

Inhalt

- MoveIt!?
- Anforderungen an Planer und Pfade
- Planungsalgorithmen in MoveIt 2
- Parallel Planning
- MoveIt 2 Update

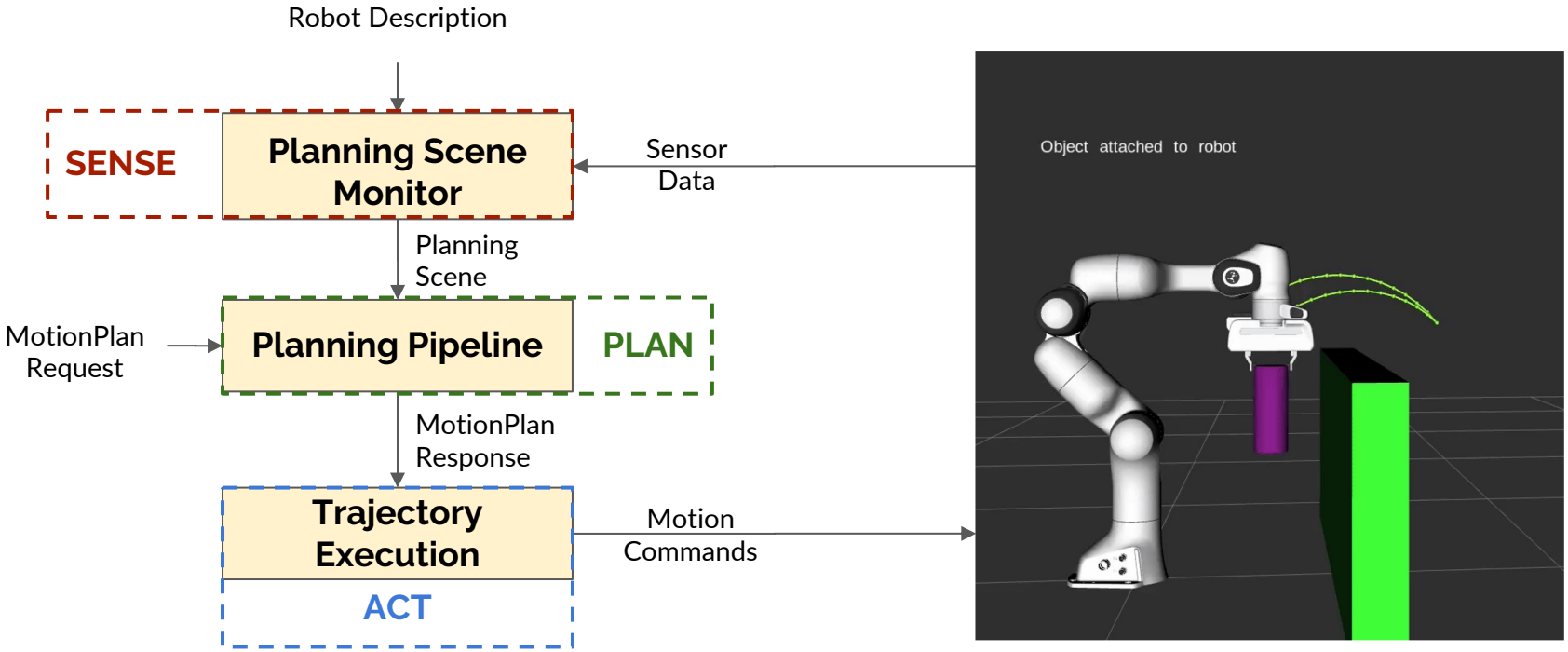


PICKNIK



PICKNIK

Konzept



Mehr Informationen: https://moveit.picknik.ai/main/index.html

PICKNIK

MoveIt 2 ist ein Open-Source Bewegungsplanung Framework!

Planning Algorithms
OMPL (Open Motion Planning Library), Pilz Industrial Motion Planner, CHOMP, STOMP

Inverse Kinematics
KDL, IKFast, pick_ik, BioIK, TRAC-IK

Perception
Point Cloud, Depth Image

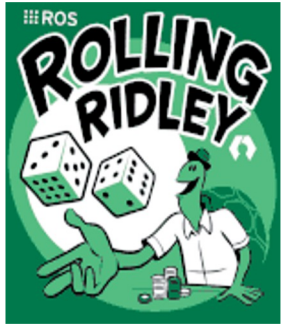
Benchmarking
Database support, Planner, IK

Collision Checking
FCL, Bullet

Trajectory Generation
TOTG, Ruckig

Simulation
Gazebo, Isaac Sim

Online Planning
MoveIt Servo, Hybrid Planning



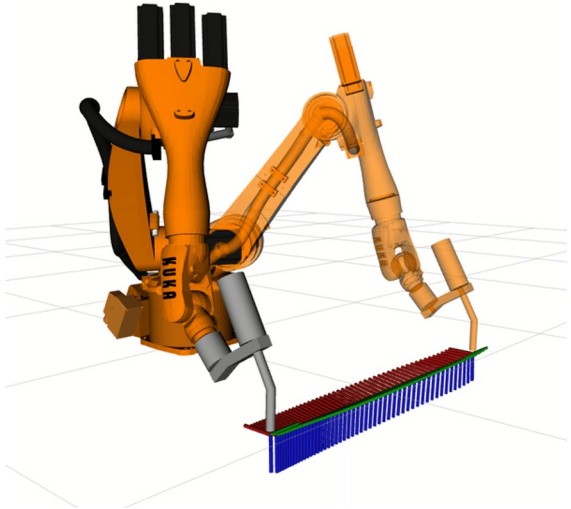
PICKNIK

Anforderungen an Planungsalgorithmen

PICKNIK

Anforderungen an Planungsalgorithmen

- Algorithmus
  - Schnell
  - Vollständig (Complete)
  - Deterministisch
- Lösung
  - Zwangsbedingungen (Constraints)
    - Frei von Kollisionen
    - Ausführbar vom Roboter
  - Optimal
    - kurz, glatt, Abstand von Objekten, minimale Bewegung
      ...



→ Es gibt bisher keinen universellen Planungsalgorithmus

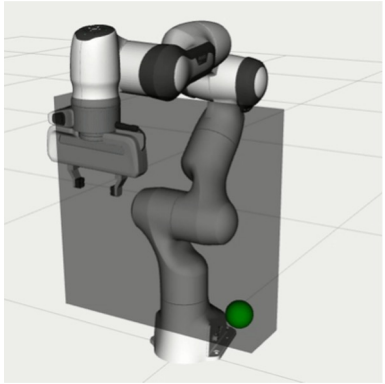
PICKNIK

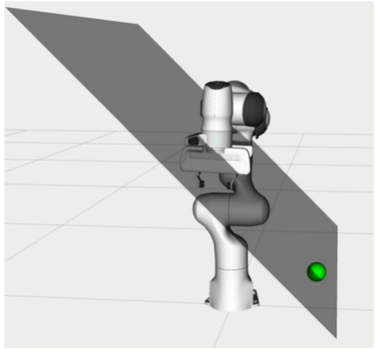
Welche Planungsalgorithmen gibt es?

PICKNIK

OMPL

- Software Bibliothek mit Vielzahl von sampling-based Planungsalgorithmus (*RRTConnect*, *RRT**, *PRM*, ...)
- Findet i.d.R. einen Pfad falls möglich
  - “Rückgrat” von MoveIt
- Unterstützt Constrained Planning unabhängig vom konkreten Planungsalgorithmus zB für
  - TCP Zwangsbedingungen zB Schweißen, Polieren etc.
  - Bewegungen in definierten Volumen aus Sicherheitsgründen



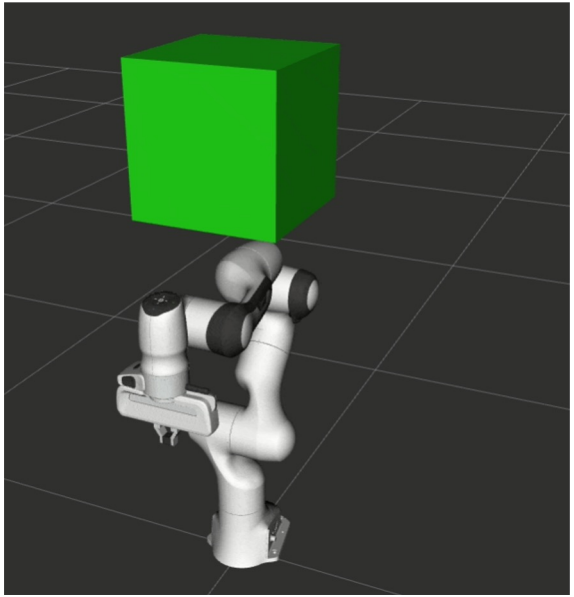


Quelle: https://ompl.kavrakilab.org/

PICKNIK

STOMP und CHOMP

- CHOMP
  - **C**ovariant **H**amiltonian **O**ptimization for **M**otion **P**lanning
  - Plant glatte kollisionsfreie Pfade mit Covariant Gradient Descent Ansatz
  - Minimiert Kombi aus Glattheit (Smoothness) und Constraint (z.B. Kollisions) Kostenfunktionen
- STOMP
  - **S**tochastic **T**rajectory **O**ptimization for **M**otion **P**lanning
  - Plant glatte kollisionsfreie Pfade mit probabilistischem Optimierungsalgorithmus → Gradient wird stochastisch approximiert → Kostenfunktion muss nicht differenzierbar sein

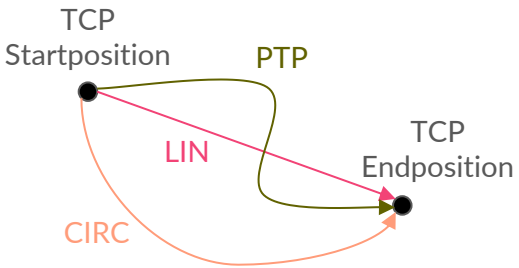


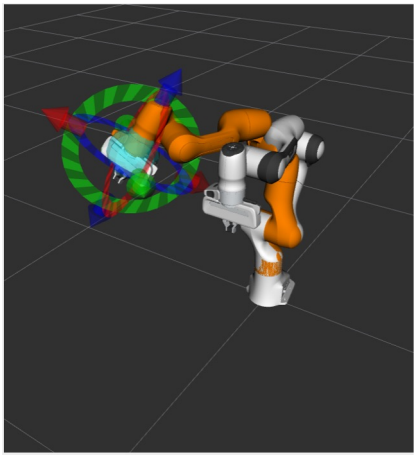
Quellen: Kalakrishnan et al. 2011 (STOMP), Ratliff et al., 2009c (CHOMP)

PICKNIK

PILZ Industrial Motion Planner

- Generiert Trajektorien für den Endeffektor
  - LIN → Lineare Interpolation im Kartesischen Raum
  - PTP → Lineare Interpolation im Konfigurations Raum (Joint Space)
  - CIRC (auch ARC genannt) → Kreisförmige Interpolation im Kartesischen Raum





Mehr Infos:
https://ros-planning.github.io/moveit_tutorials/doc/pilz_industrial_motion_planner/pilz_industrial_motion_planner.html

PICKNIK

Einen Planer auszuwählen ist hart!

OMPL	+ Große Auswahl an Samling-based Planern + Probabilistisch vollständig + Findet relativ schnell und sehr zuverlässig eine Lösung	- Pfade sind oft nicht optimal und intuitiv - Parametrisierung erfordert Expertenwissen	➔ Geeignet für komplexe Probleme (Viele Kollisionsobjekte + Zwangsbedingungen) zB Pick & Place in unstrukturierter Umgebung
PILZ	+ Generiert Kartesische Trajektorien + Sehr schnell + Bewegungen typisch für Industrielle Arme (LIN, PTP & CIRC)	- Beachtet keine Zwangsbedingungen - Keine kollisionsfreie Bewegung	➔ Typische Industrieanwendungen in strukturierten Umgebungen zB Schweißen, Bewegung Freiräumen
CHOMP/ STOMP	+ Generiert glatte und reproduzierbare Pfade + Kann anwendungsspezifische Zwangsbedingungen und Kostenfunktionen einbeziehen	- Relativ langsam - Nicht vollständig - Sehr hart zu parametrisieren	➔ Nützlich für optimale Bewegungen zB Optimiere Taktzeit

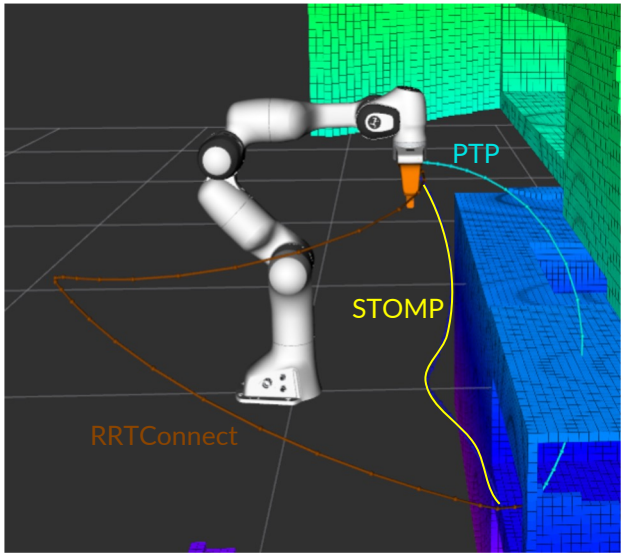
PICKNIK

Parallel Planning

PICKNIK

Motivation

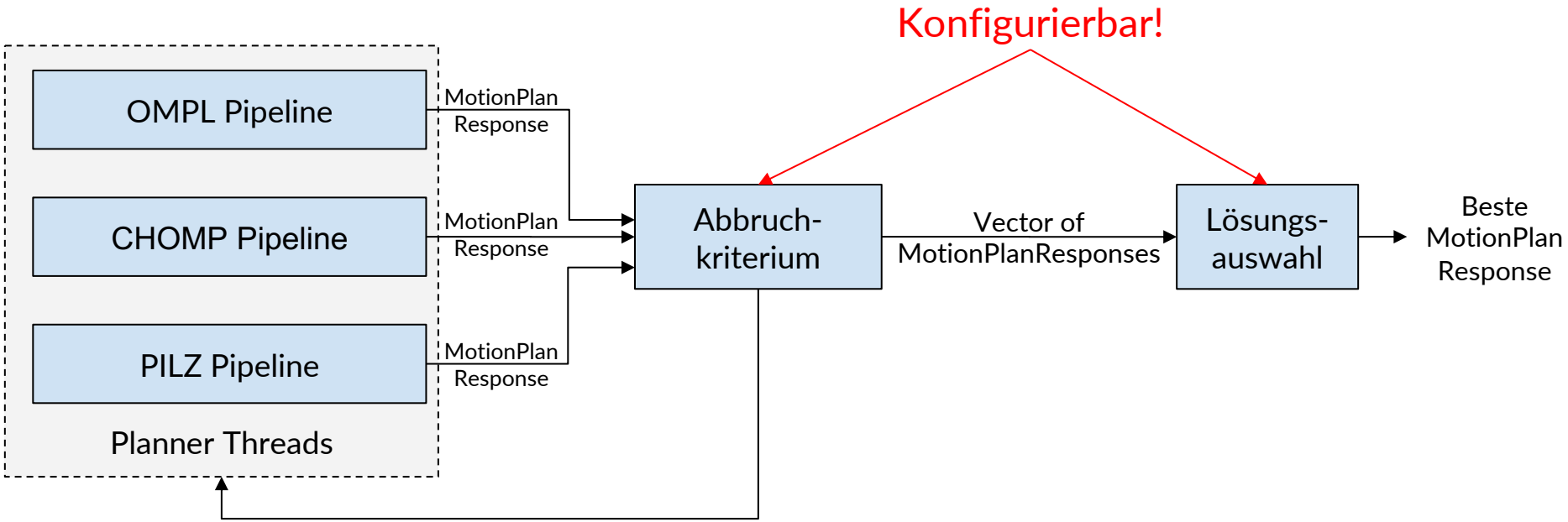
- Oft ist es unklar welcher Planer die (beste) Lösung berechnet
- Manchmal braucht man einen Backup Lösung
- Robuster Planer für unterschiedliche Probleme
  - Für jedes Problem individuell Algorithmus wählen ist mühsam



→ Nutze Portfolio von Planungsalgorithmen

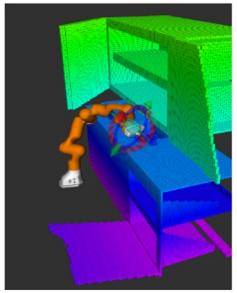
PICKNIK

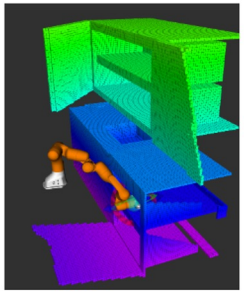
Implementierung

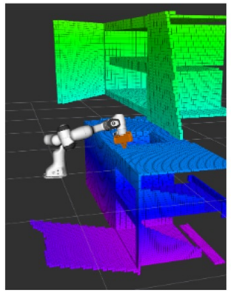


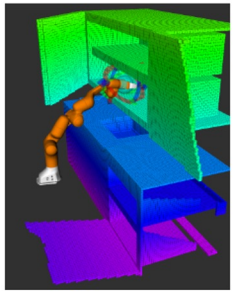
PICKNIK

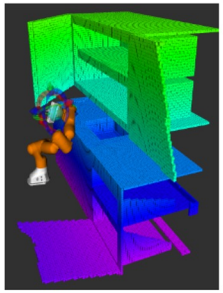
Evaluation

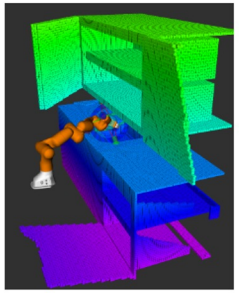

Query 0

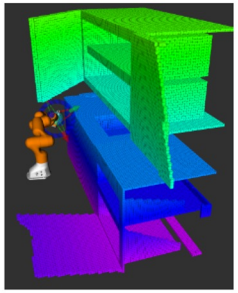

Query 1

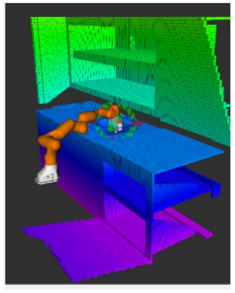

Query 2

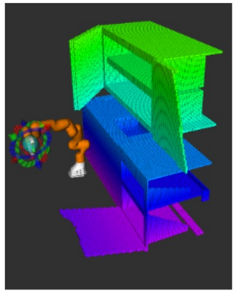

Query 3

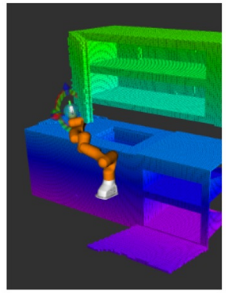

Query 4


Query 5

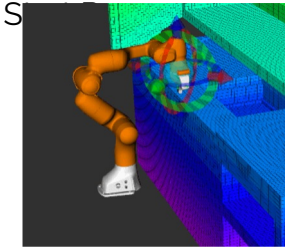

Query 6


Query 7


Query 8

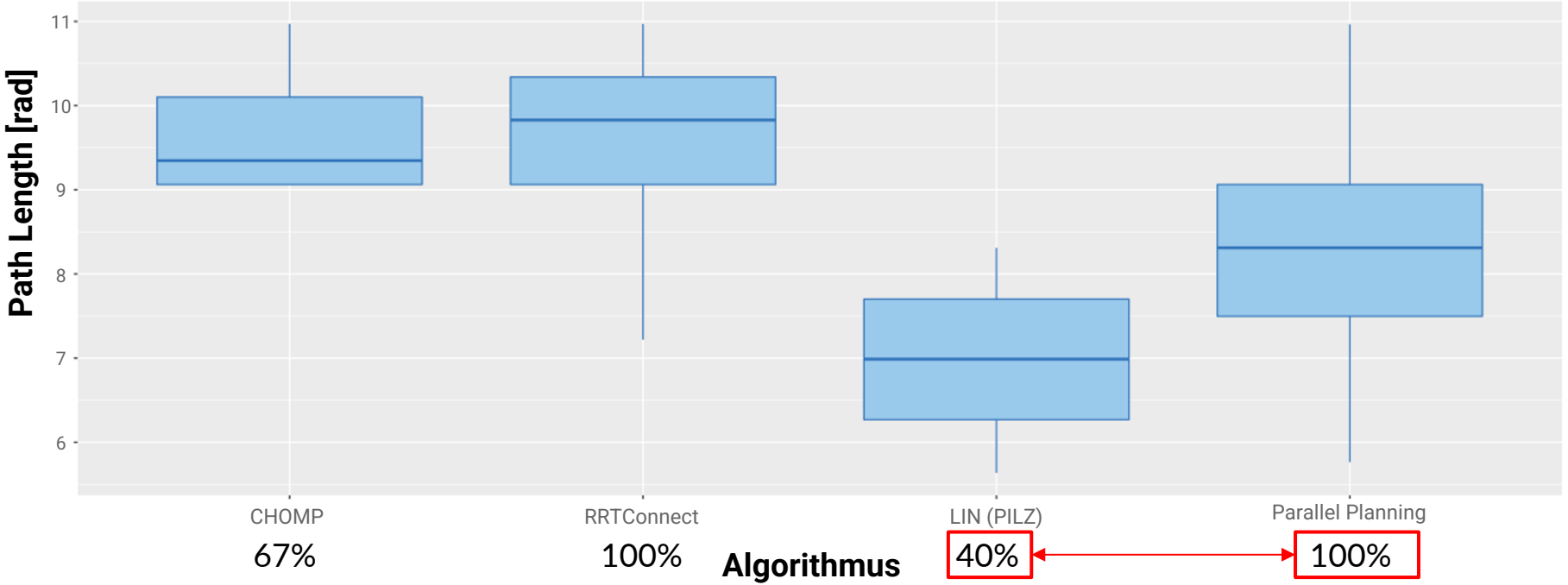

Query 9

- 10 Benchmark Probleme (Queries)
- 50 Iterationen pro Problem
- 1,5s max. Planungszeit
- Kürzeste (L1 Norm Joint Space) Lösung gilt als optimal
- Planerportfolio:
  - *RRTConnect* (*OMPL*)
  - *LIN* (*PILZ*)
  - *CHOMP* (*CHOMP*)
- S



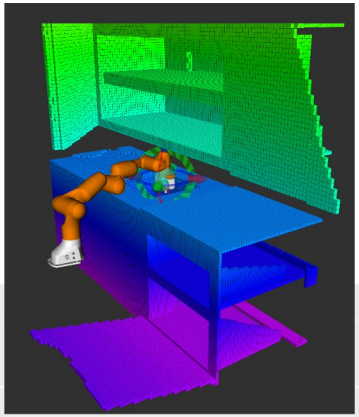
PICKNIK

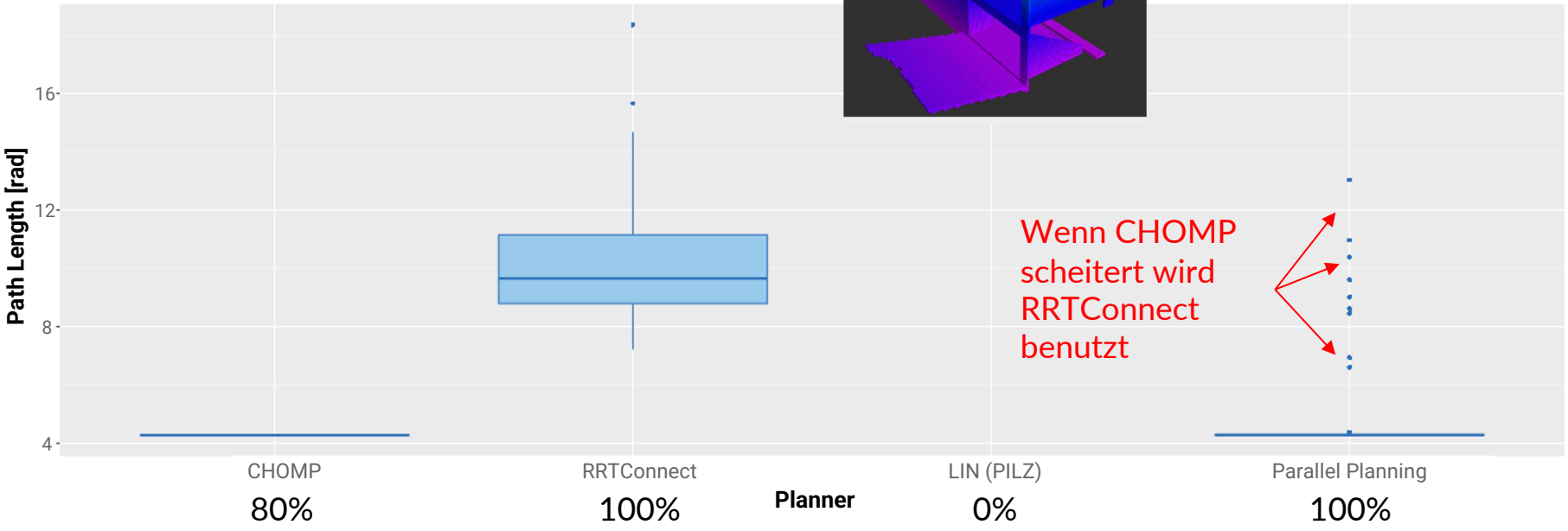
Allgemeines Ergebnis



PICKNIK

Problem 7

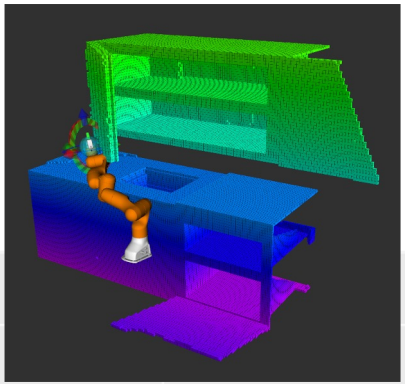


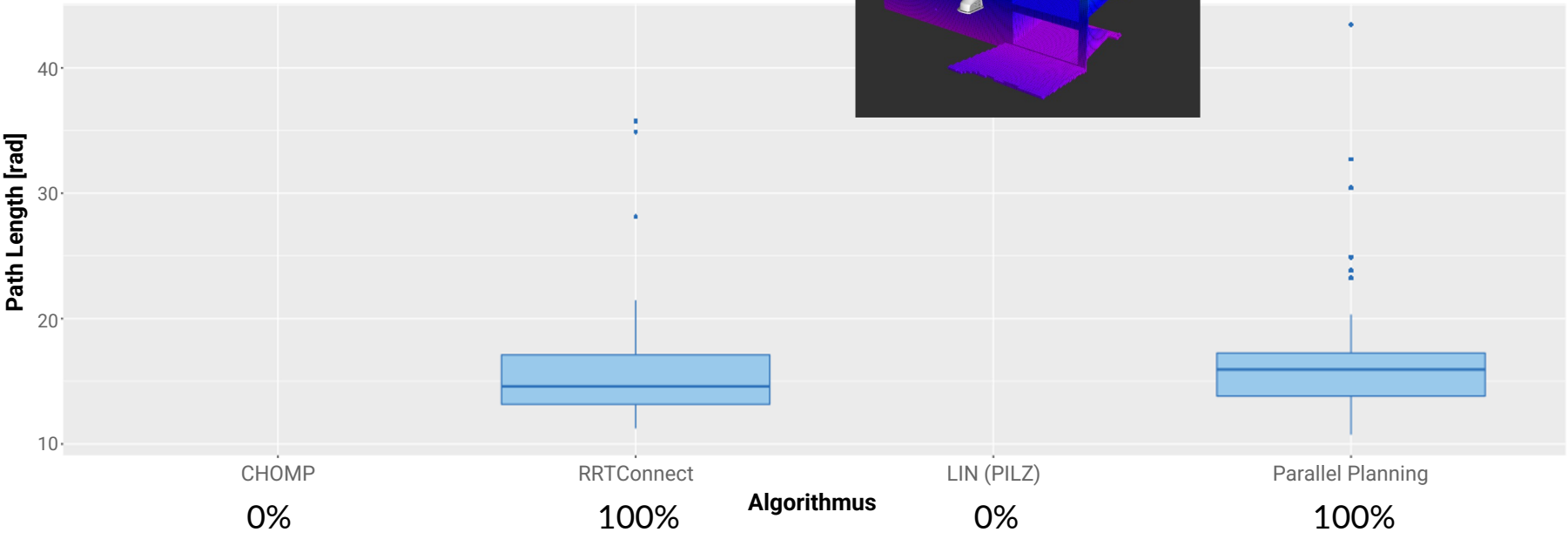


80% 100% 0% 100%

PICKNIK

Problem 9





→ Parallel Planning niemals besser als der beste Planungsalgorithmus!

PICKNIK

MoveIt 2 Update

PICKNIK

Moving MoveIt 2

Python Bindings!
(Peter David Fagan)
C++ MoveIt 2 Funktionalität nutzbar in Python

IK cost function API
(Wyatt Rees)
Benutzerdefinierte Kostenfunktionen für IK Algorithmen

Und mehr in Entwicklung

TRAC-IK Kinematics Plugin
(TRACLabs)
Newton und SQP Verfahren basierter IK Algorithmus

Isaac Integration
(Marq Rasmussen & Jafar Abdi)
NVIDIA Isaac Simulationsumgebung nutzbar mit MoveIt 2

PICKNIK

Hier gehts weiter:

PICKNIK

MoveIt Studio

DevTool and SDK

MoveIt 2

Documentation

Danke!