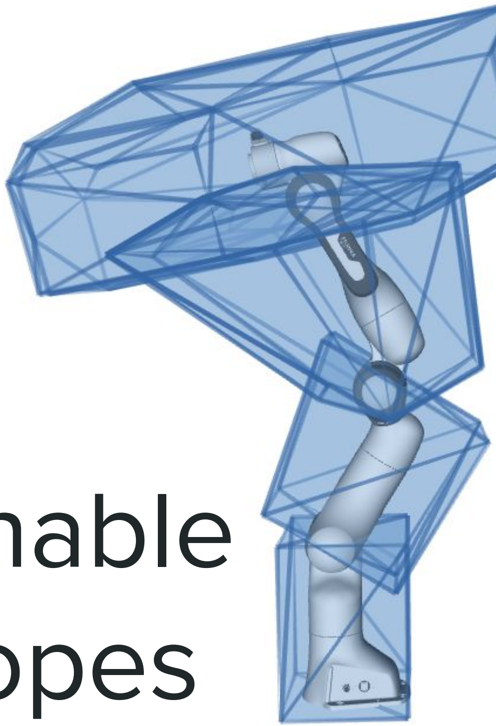


Approximating robot reachable space using convex polytopes

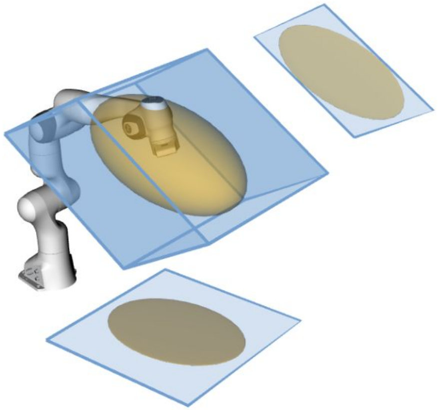
Antun Skuric, Vincent Padois, David Daney
INRIA Bordeaux, team AUCTUS
web: team.inria.fr/auctus
email: firstname.lastname@inria.fr

Human robot interaction perspective

- Human and robots in close proximity
  - Robot programming
  - Interactive tasks

- To go a step further
  - Design tools for safe and capable robots
  - Providing operators with more information about robot’ s current state and its physical abilities
    - In real-time
    - Ideally: in future

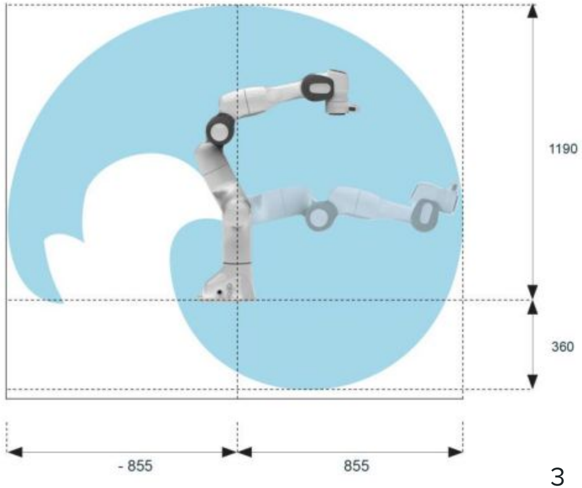
Robot’s physical abilities



- What can we quantify and predict
  - Force capacity
  - Velocity capacity
  - Acceleration capacity, …

- Reachable space

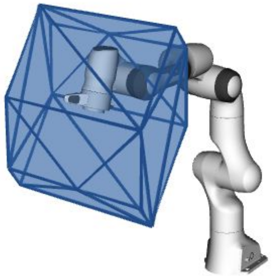
  “*Space of reachable cartesian positions within a horizon*”
  - Important for safety and performance
  - Complex geometry
    - Hard to characterise
    - Usually done in advance for the whole workspace
  - Not considering robot’s dynamics

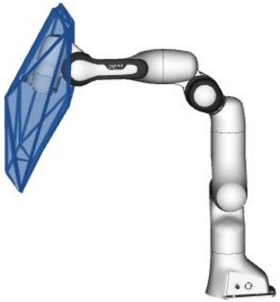


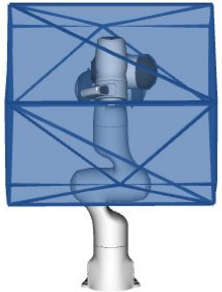
Proposed reachable space method

- An approximation approach
  - Considering robot's dynamics and actuation constraints
  - Using convex polytope based description

- Convex polytopes
  - Efficient to calculate - Real-time capable
  - Set of constraints - could be used for optimisation
  - Easy to visualise - polytopes = 3d mesh







Problem definition

- A set of all achievable positions $x_{k+1} \in R^m$ at the end of the horizon t_h
  - Given current robot state $\dot{q}, q$

- Respecting the actuator constraints

$$\tau \in [\tau_{min}, \tau_{max}] \quad \dot{q} \in [\dot{q}_{min}, \dot{q}_{max}], \quad q \in [q_{min}, q_{max}]$$

- And robot's dynamics and kinematics

$$M(q)\ddot{q} + \underbrace{c(q, \dot{q}) + g(q)}_{\tau_d} = \tau$$

$$\ddot{q} = M(q)^{-1}(\tau - \tau_d)$$

$$\ddot{x} = J(q)\ddot{q} + \dot{J}(q, \dot{q})\dot{q}$$

$$\dot{x} = J(q)\dot{q}$$

Problem definition

- A set of all achievable positions $x_{k+1} \in R^m$ at the end of the horizon t_h
  - Given current robot state $\dot{q}, q$
- Respecting the actuator constraints

$$\tau \in [\tau_{min}, \tau_{max}] \quad \dot{q} \in [\dot{q}_{min}, \dot{q}_{max}], \quad q \in [q_{min}, q_{max}]$$

- And robot's dynamics and kinematics

$$M_k \ddot{q} + \tau_d = \tau$$

$$\ddot{q} = M_k^{-1}(\tau - \tau_d)$$
$$\ddot{x} = J_k \ddot{q} + \dot{J}_k \dot{q}$$
$$\dot{x} = J_k \dot{q}$$

➔ Assuming robot's dynamics and kinematics constant and linear in horizon t_h
  - Reasonable for short horizon lengths

Polytope definition

- Propagating robots dynamics using Euler backward integration

$$x_{k+1} = \ddot{x}_k \frac{t_h^2}{2} + \dot{x} t_h + x_k$$

- Polytope form

$$\begin{aligned} \mathcal{P}_x = \{x_{k+1} \in \mathbf{R}^m \quad | \quad x_{k+1} = J_k M_k^{-1} \frac{t_h^2}{2} \tau + x_{k+1}^*, \\ \tau \in [\tau_{min}, \tau_{max}], \\ M_k^{-1} t_h (\tau - \tau_d) + \dot{q}_k \in [\dot{q}_{min}, \dot{q}_{max}], \\ M_k^{-1} \frac{t_h^2}{2} (\tau - \tau_d) + \dot{q}_k t_h + q_k \in [q_{min}, q_{max}]\} \end{aligned}$$

Polytope definition

- Propagating robots dynamics using Euler backward integration

$$x_{k+1} = \ddot{x}_k \frac{t_h^2}{2} + \dot{x} t_h + x_k$$

- Polytope form

$$\mathcal{P}_x = \{x_{k+1} \in \mathbf{R}^m \quad | \quad \boxed{x_{k+1}} = J_k M_k^{-1} \frac{t_h^2}{2} \boxed{\tau} + x_{k+1}^*,$$
$$\tau \in [\tau_{min}, \tau_{max}],$$
$$M_k^{-1} t_h (\tau - \tau_d) + \dot{q}_k \in [\dot{q}_{min}, \dot{q}_{max}],$$
$$M_k^{-1} \frac{t_h^2}{2} (\tau - \tau_d) + \dot{q}_k t_h + q_k \in [q_{min}, q_{max}]\}$$

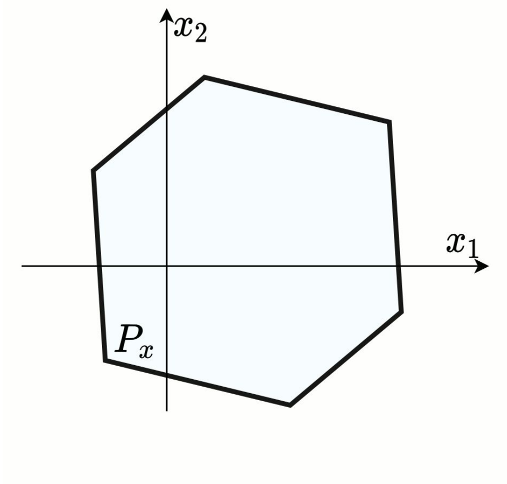
Torque

Velocity

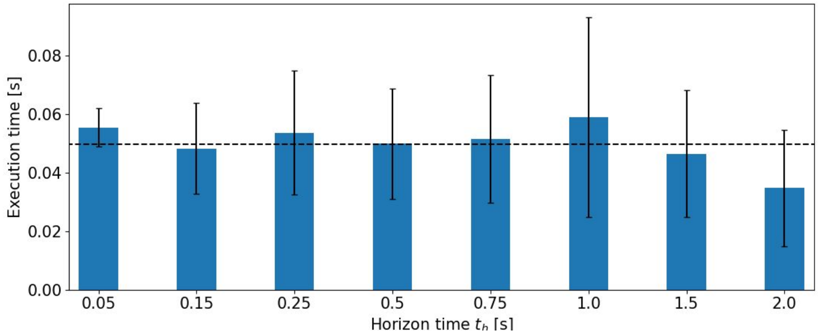
Position

Polytope resolution

- Finding vertices and faces
  Iterative convex hull method
  - Iterations of linear programming and convex hull
  - Approximating polytope with user define accuracy



- Execution time of around 50ms
  - Constant execution time horizon independent
  - Franka Emika Panda robot

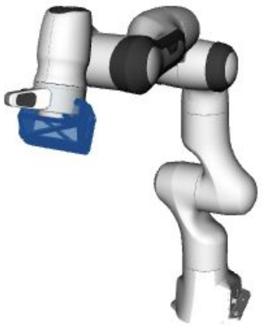


- Enabling interactive applications
  - Real-time capable

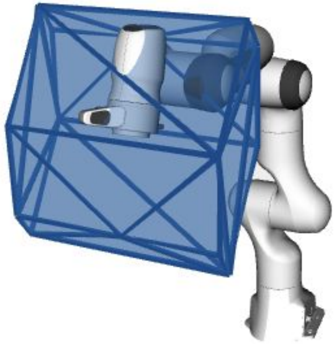
Skuric, Antun, et al. "On-line feasible wrench polytope evaluation based on human musculoskeletal models: an iterative convex hull method." *IEEE Robotics and Automation Letters* 7.2 (2022): 5206-5213.

Effects of horizon length

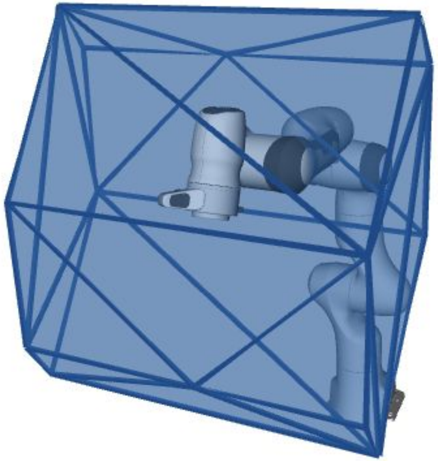
https://youtu.be/JwZgrUp095Y



$t_h = 50ms$



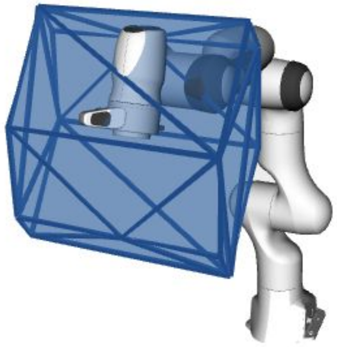
$t_h = 150ms$

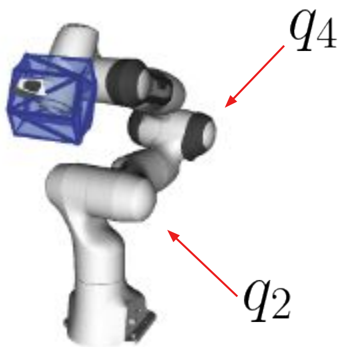


$t_h = 250ms$

Effects of joint limits

https://youtu.be/JwZgrUp095Y

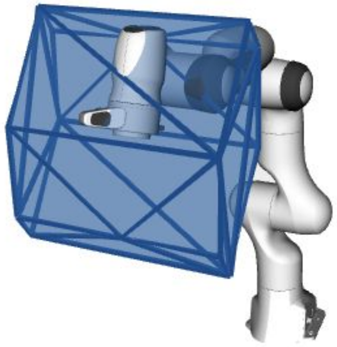


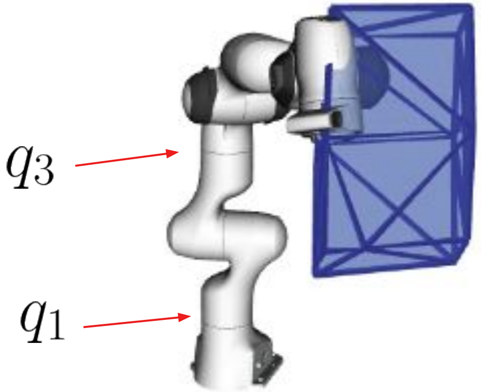


$$t_h = 150ms$$

Effects of joint limits

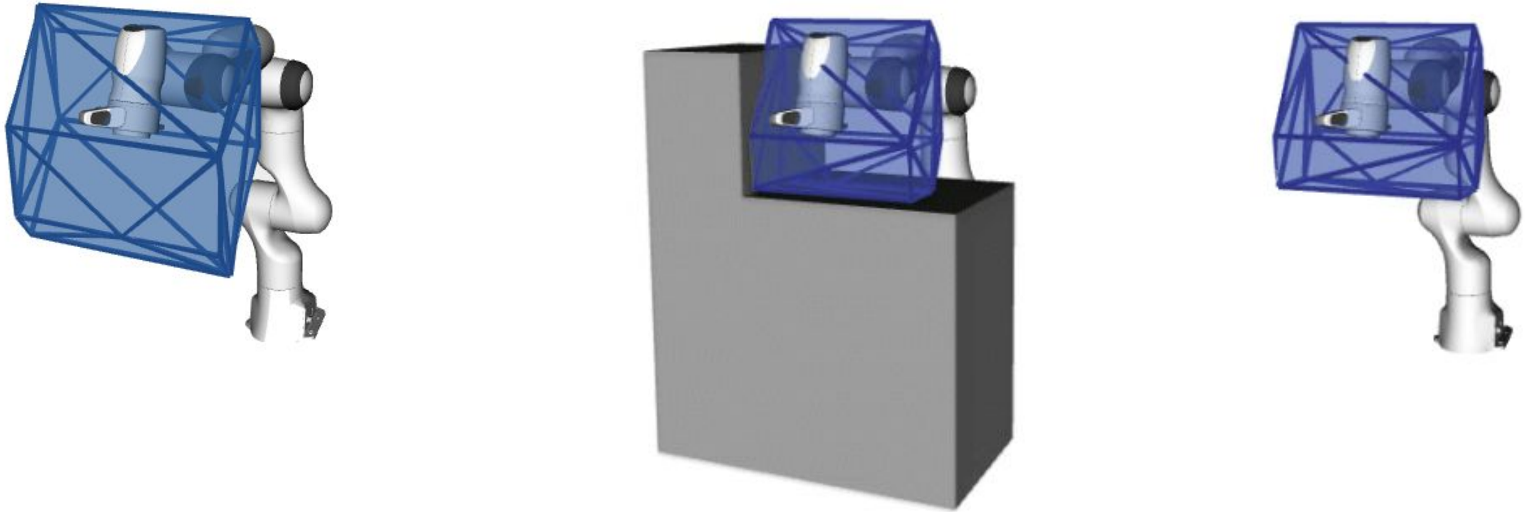
https://youtu.be/JwZgrUp095Y





$$t_h = 150ms$$

Environment constraints

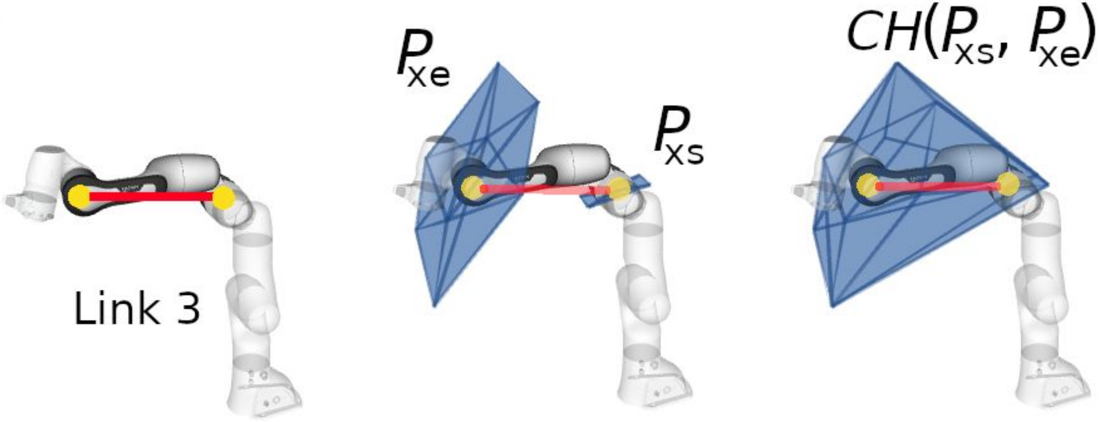


- If the environment can be expressed as set of constraints

$$Ax \leq b$$

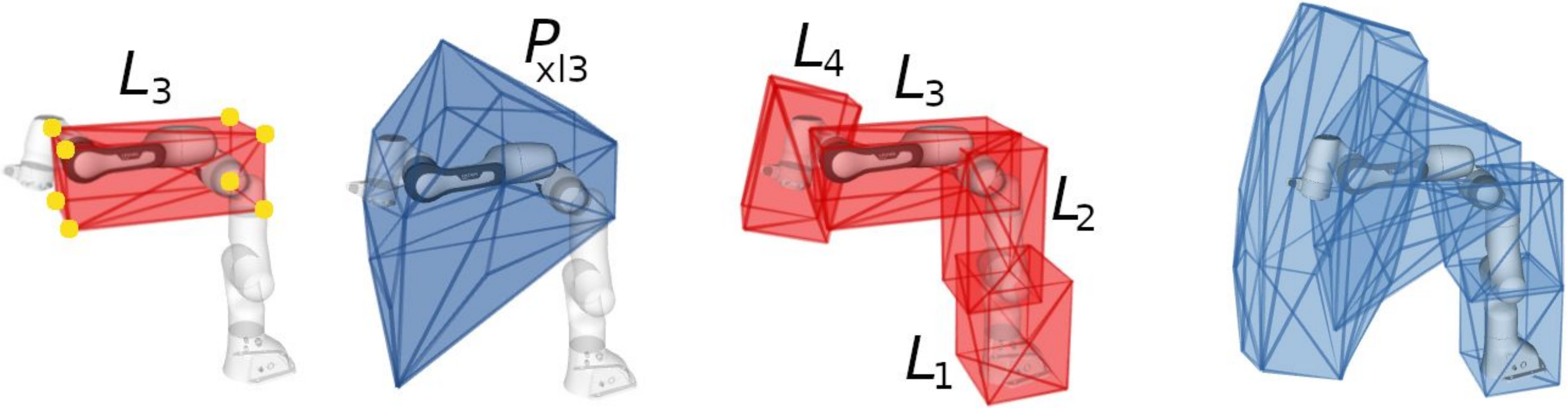
Robot link geometry

https://youtu.be/1J2UrMC2uP0



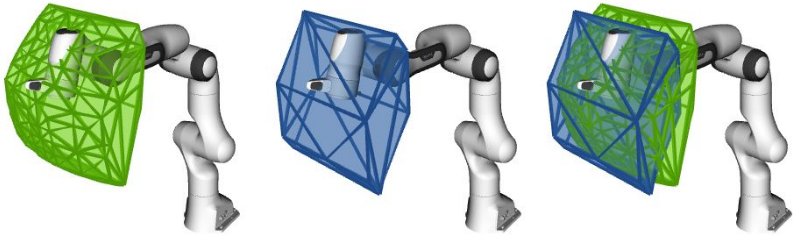
Robot link geometry

https://youtu.be/1J2UrMC2uP0



Accuracy analysis

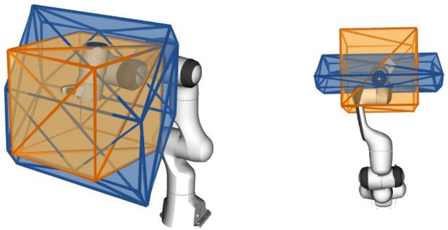
Accuracy analysis



- Simulation of robot's reachable space
  - Robot's nonlinear dynamics
  - Long execution time

- Proposed metric has good performance for short horizon times $t_h \leq 250ms$
  - Capturing most of the simulated reachable space (> 80%)
  - Having comparable volume

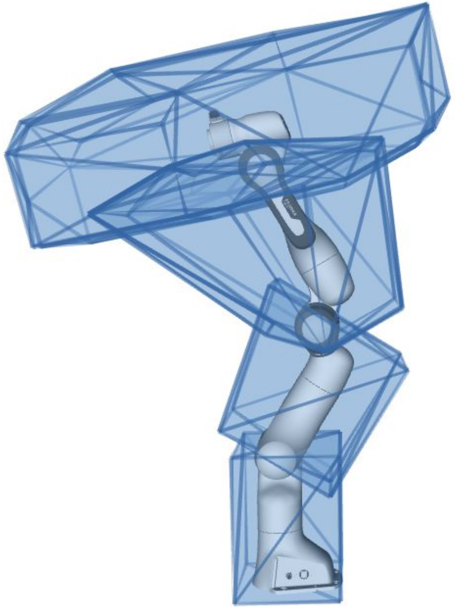
- Benchmarking against fixed task-space limits
  - Manufacturer's datasheet
  - Proposed method more precise for all horizon lengths



Conclusion

- New convex polytope based reachable space approximation
- Good accuracy for horizon times $t_h \leq 250ms$
  - Reasonable assumptions on constant robot's dynamics
- Real-time capable
  - 50ms execution time for Franka Emika Panda robot

- Yet, not under or over approximation



Email: *antun.skuric@inria.fr*
GitLab: *https://gitlab.inria.fr/auctus-team/people/antunskuric/papers/reachable_space*
HAL: *https://hal.archives-ouvertes.fr/hal-03719885*

This work has been funded by the BPI France Lichie project