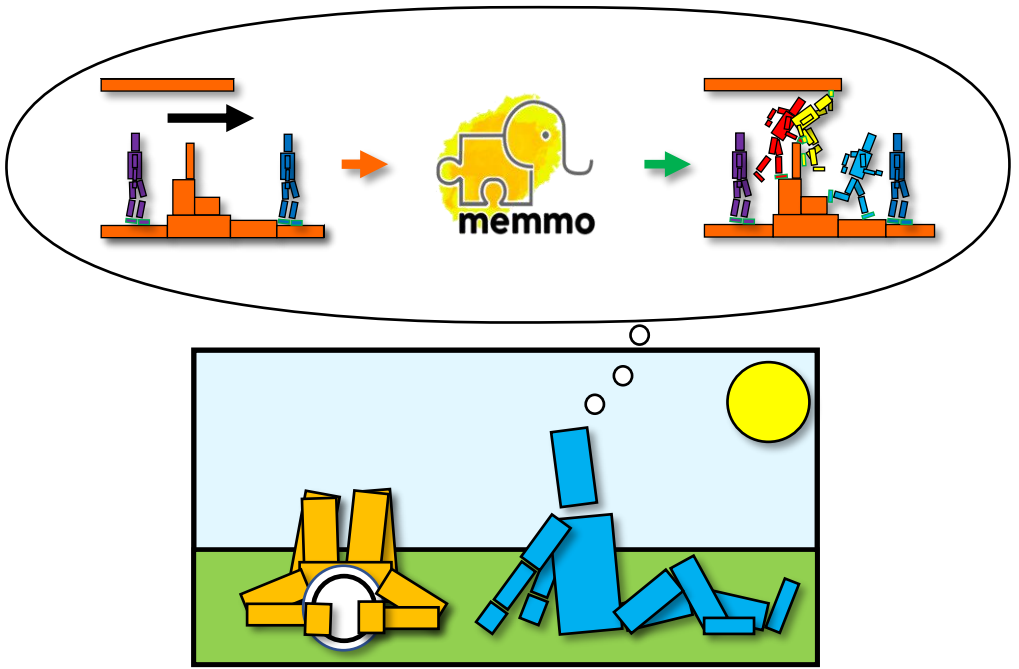
Memmo Summer school 2020 - Introduction



Steve Tonneau

Before going further: install the software !

Instructions:

https://memory-of-motion.github.io/summer-school/participate
short url: http://cpc.cx/rWV

4 options:

Docker / Virtual machine (Recommended)
Binary / source installation (ubuntu 18.04)

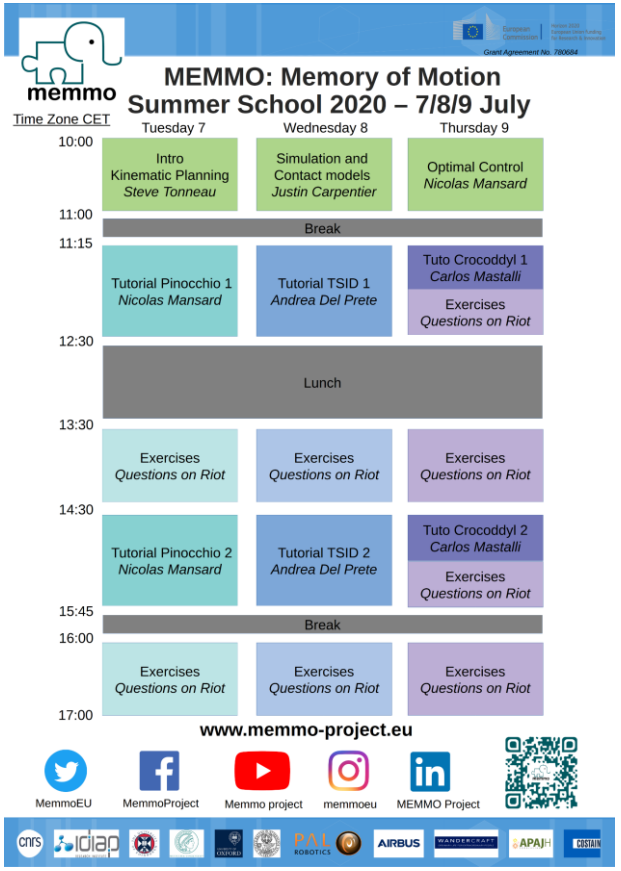
Before going further: get to the chat rooms !

Please join the main room #memmo-ss20:matrix.org

You will need a riot account

More details at the end of the talk

Class Schedule



Plan for the introduction

Objectives for the school / requirements

Context:

- Legged locomotion through the loco3D project
- The memmo project

Syllabus and practical information

Objectives for the summer school

Hands tutorial for (part of) the Memmo software suite:

- Model the kinematic chain of a robot
- Use rigid body algorithms for posture optimisation
- Design / resolve optimisation problems for motion control involving contacts

Audience and requirements

Researchers / engineers working on robot control
    mainly legged robots, but also manipulators

Theoretical requirements:
    Basic calculus (limit, derivative, gradient, Jacobian)
    Basic algebra (matrix operations, eigenvalues)

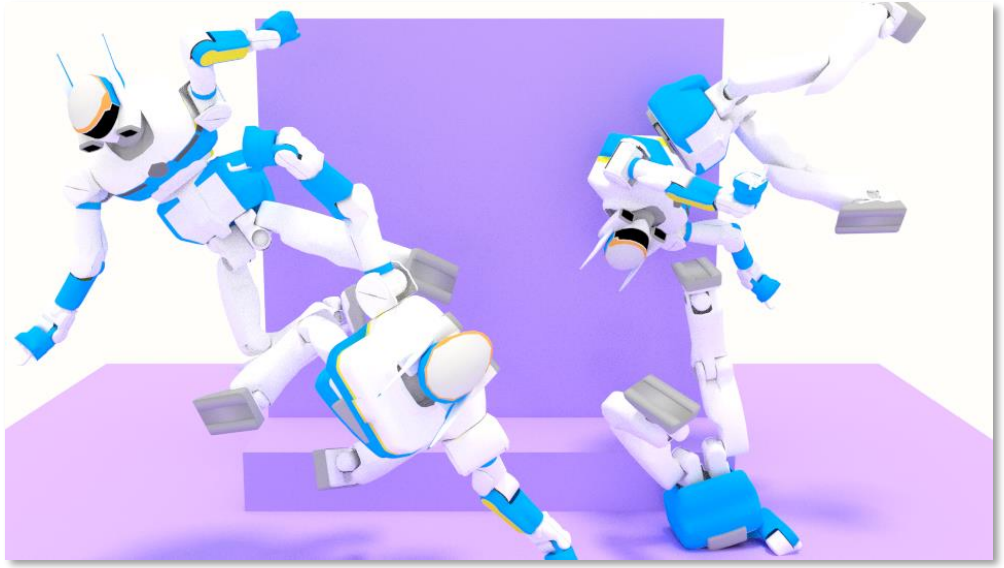
Technical requirements:
    How to implement those concepts in Python 3

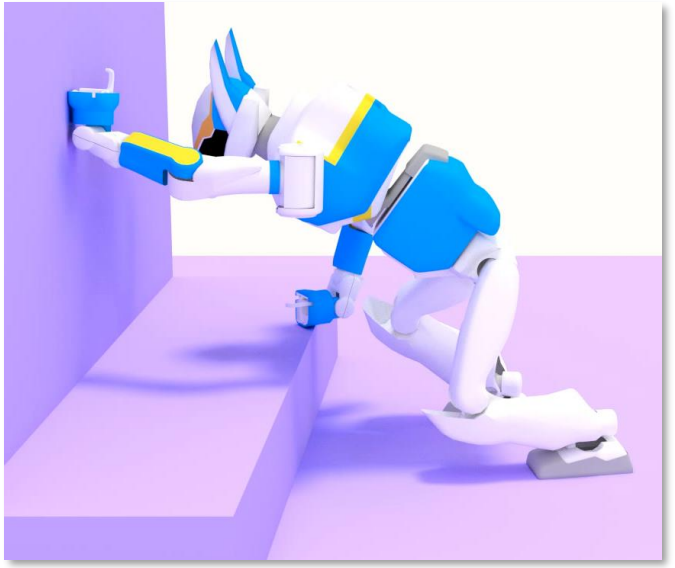
Context:
From loco3D to Memmo

What we do with our software

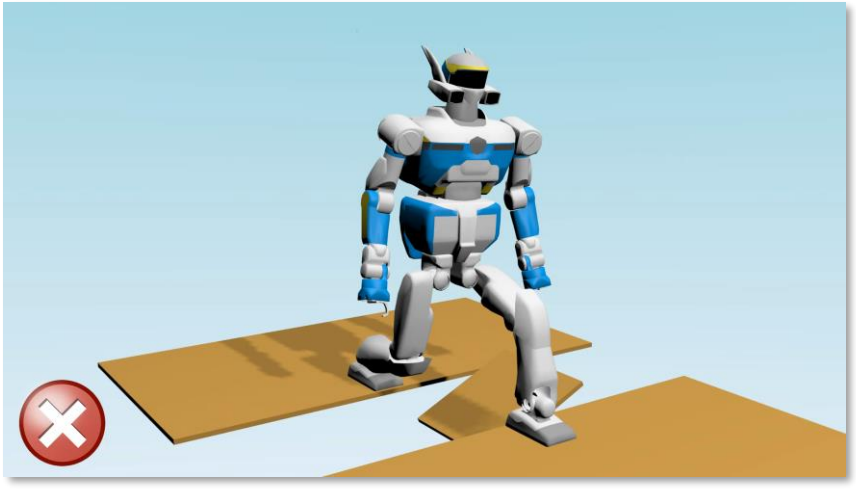
The complexity of legged locomotion

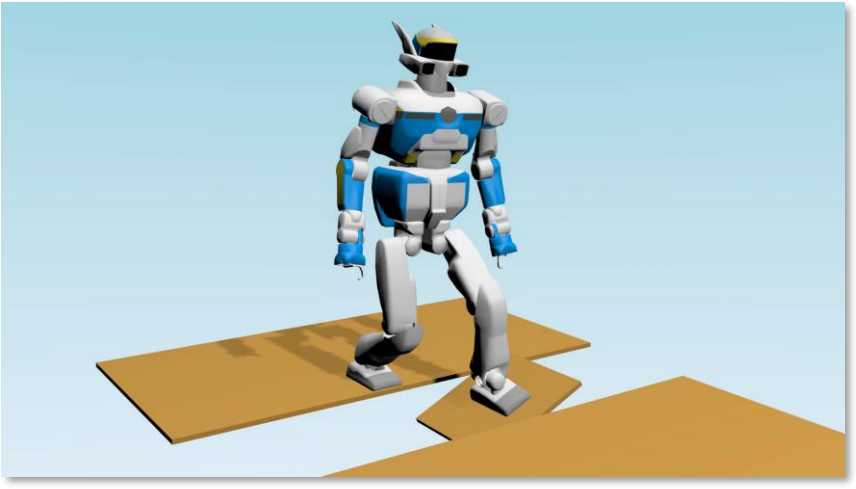
Contact postures in high-dimensional space ?



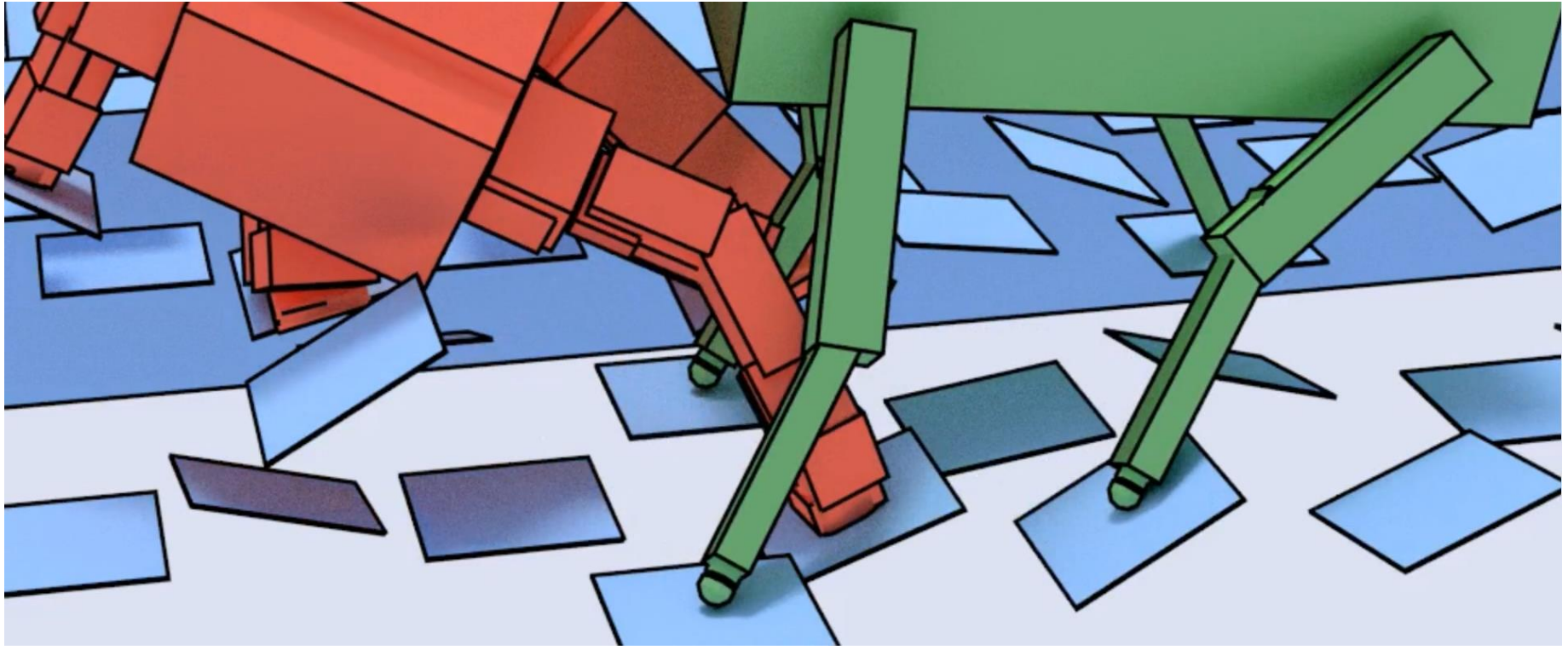


Contact-dependent, discontinuous, non-linear dynamics / geometric constraints





Contact interactions without collisions ?

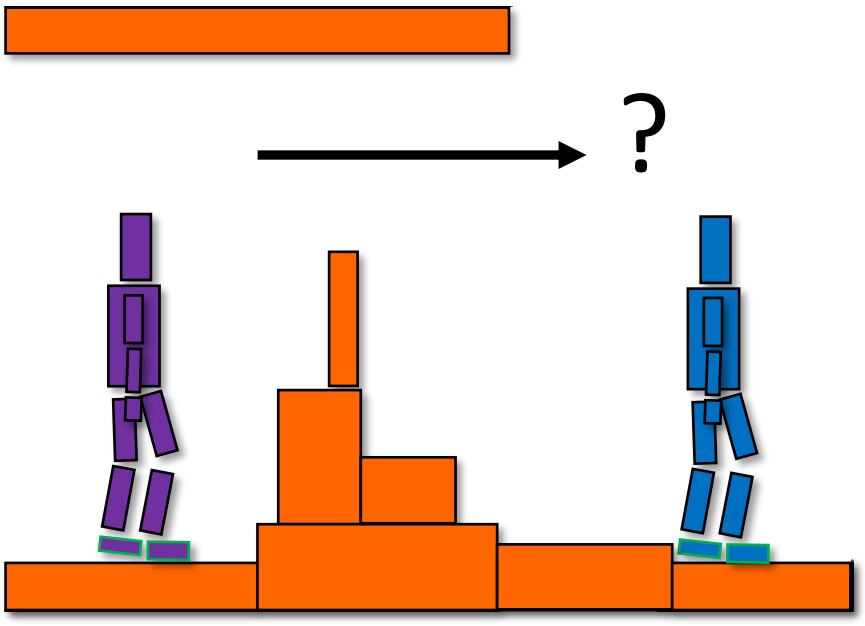


Legged locomotion is too hard

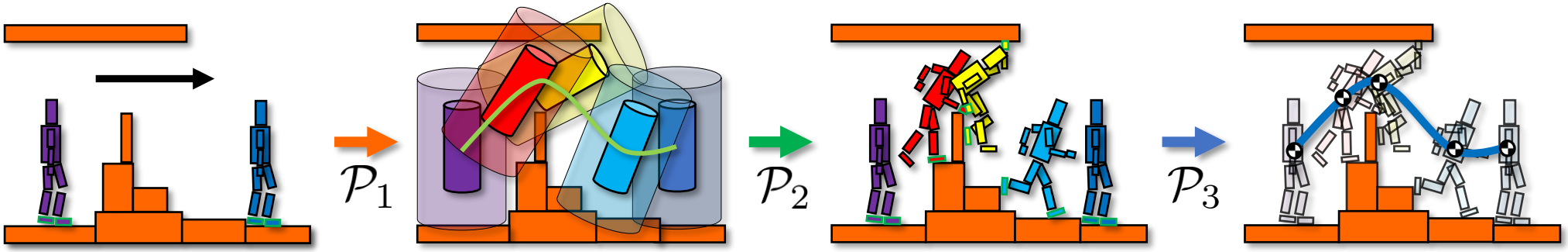
We need to cheat !

This has a cost …

loco3D: A divide and conquer approach

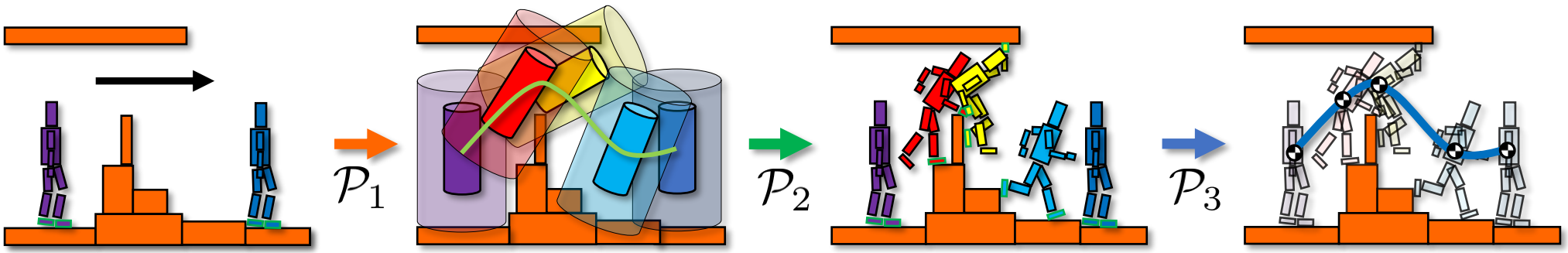


loco3D: A divide and conquer approach



loco3D: A divide and conquer approach

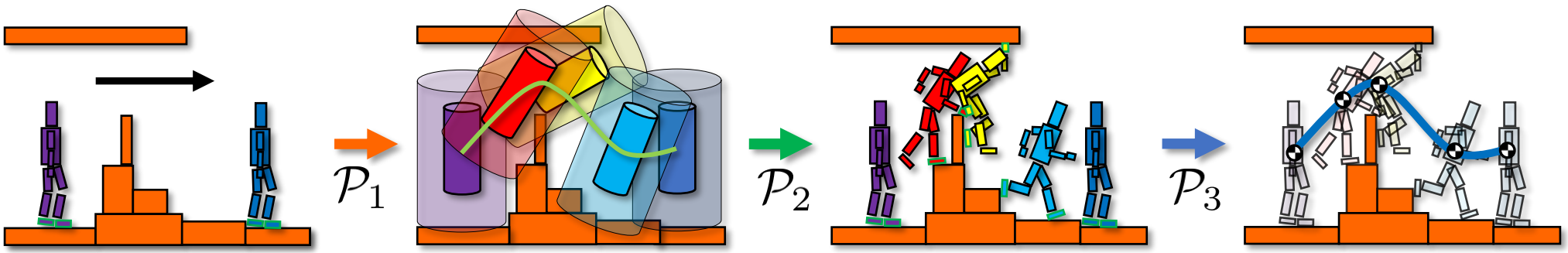
Global path planner



loco3D: A divide and conquer approach

Global path planner

Contact Planner

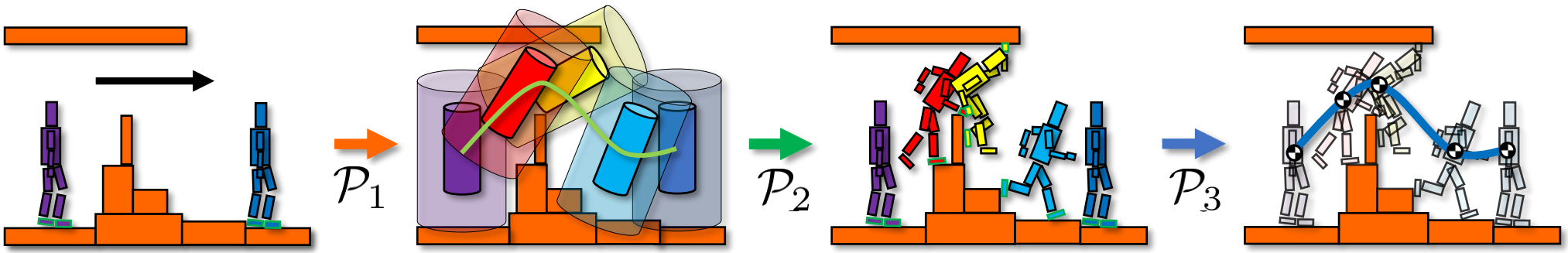


loco3D: A divide and conquer approach

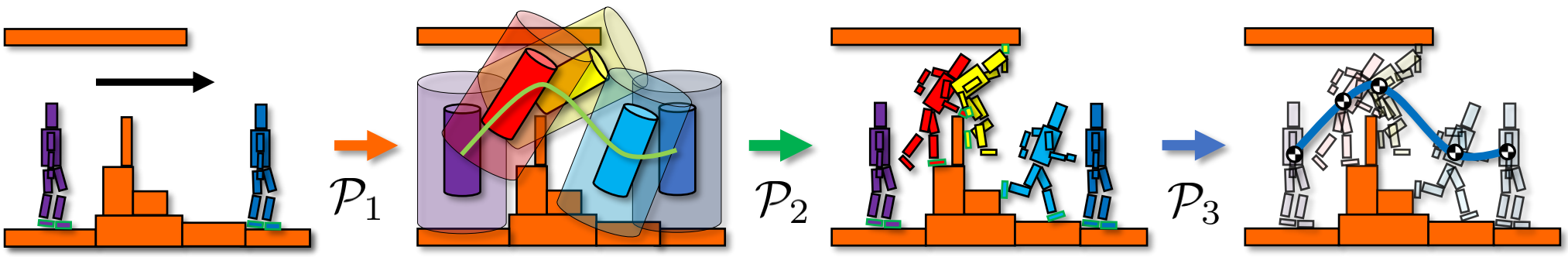
Global path planner

Contact Planner

CoM + Whole body motion generator



The importance of having good models

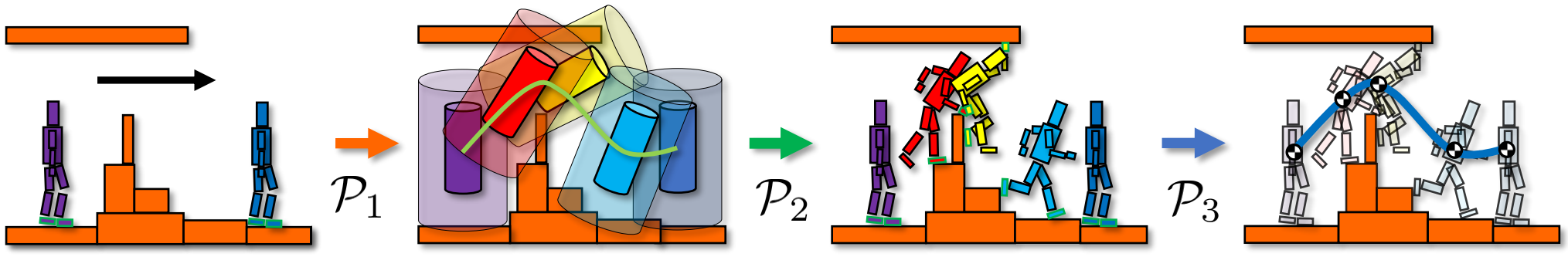


Solve $\mathcal{P}_i$ in the feasibility domain of $\mathcal{P}_j, \forall j < i$

$\Leftrightarrow$

A good initial guess for $\mathcal{P}_3$?

The loco 3D project – software*



*references given on last slide

What does TSID do in this pipeline ?

Given:

- Reference COM trajectory
- Reference end-effector trajectories
- Contact locations and timings

Compute:

Whole body motion that (locally) follows at best reference

TSID Pros and cons

TSID is really good for control:

- Computationally efficient
- Robust (well-established theoretical background)
- Easy to use

What if the initial guess is not so good / not feasible ?

We want more flexibility

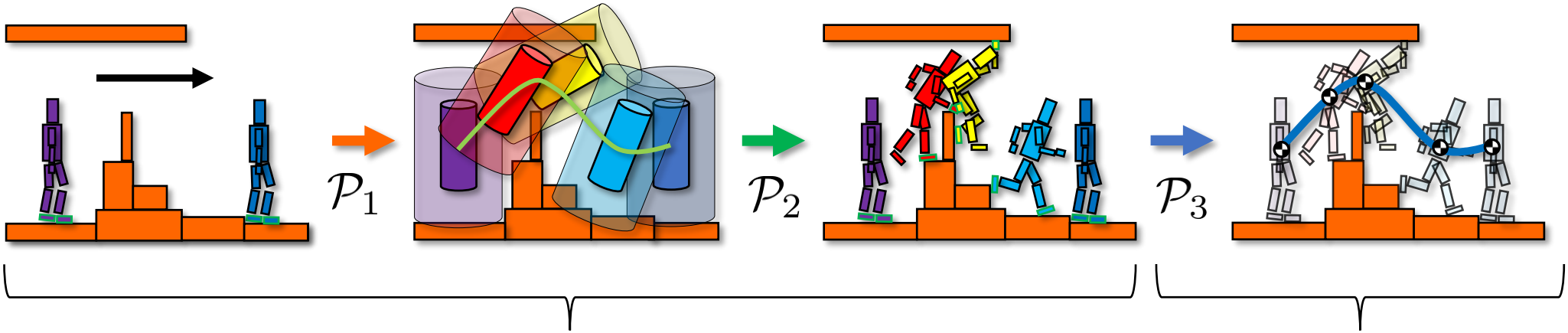
Croccodyl: specifications

Motion synthesis framework able to initial guess neighbourhood:

- Repositions contacts as needed
- Handles centroidal dynamics and contact timings

For efficiency, still requires initial guess

The Memmo project – ideal software architecture*



"Memory of motion"
(Machine learning)

Croccodyl

Pinocchio

*references given on last slide

The Memmo consortium

Class details

Pinocchio - Today

Implementation of rigid body algorithms and their derivatives

Target demo:

Posture optimisation for a humanoid in contact, integrated in a first prototype of a randomised contact planner

Schedule:

Geometry and dynamic algorithms (1:15 hour) + implementation (1 hour)
Deriving algorithms (1:15 hour) + implementation (1 hour)

Nicolas Mansard

Rohan Budhiraja

Carlos Mastalli

TSID - Wednesday

Hands-on introduction to optimization-based control framework

Target demo:

Balance control of humanoid subject to external pushes

Schedule:

Joint Space Control (1:15 hour) + Implementation (1 hour)
Task Space Control (1:15 hour) + Implementation (1 hour)

Staff:

Andrea Del Prete

Noelie Ramuzat

Sanghyun Kim

Croccodyl - Friday

Introduce Crocoddyl API and its main design choice

Target demo:

Various optimisation problems (unicycle, cart-pole, whole-body manipulation / locomotion)

Schedule:

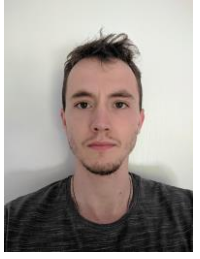
2 * 30 min tutorials

2 * 1.45 hours exercises

Staff:

Carlos Mastalli

Wolfgang Merkt



Gabriele Fadini

Teguh Santoso Lembono

Class details

Tutorials:

Everything happens here (Big Blue Button)

Exercices:

Presentation and oral discussion here

Text-based chat on dedicated riot channels

Technical support:

Guilhem Saurel

Software references

HPP:

https://humanoid-path-planner.github.io/hpp-doc/

Mirabel et al. “HPP: A new software for constrained motion planning” - *IROS 16.*

RBPRM:

https://github.com/humanoid-path-planner/hpp-rbprm

Tonneau et al. “An efficient acyclic contact planner for multiped robots” - *TRO 19*

SL1M:

https://github.com/loco-3d/sl1m

SL1M: Sparse L1-norm Minimization for contact planning on uneven terrain
Tonneau et al. – *ICRA 20*

Momentum opt:

https://github.com/machines-in-motion

Ponton et al. “On Time Optimization of Centroidal Momentum Dynamics” – *ICRA 18*