Controls
Electronics &
Mechatronics

BEAMS

CERN

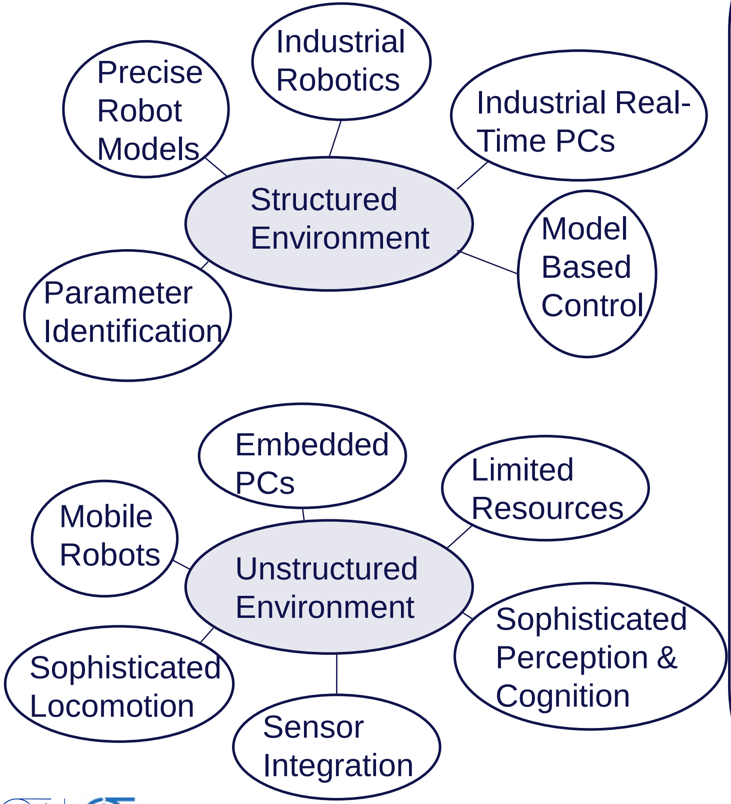
Design Optimization, Kinematics & Dynamics

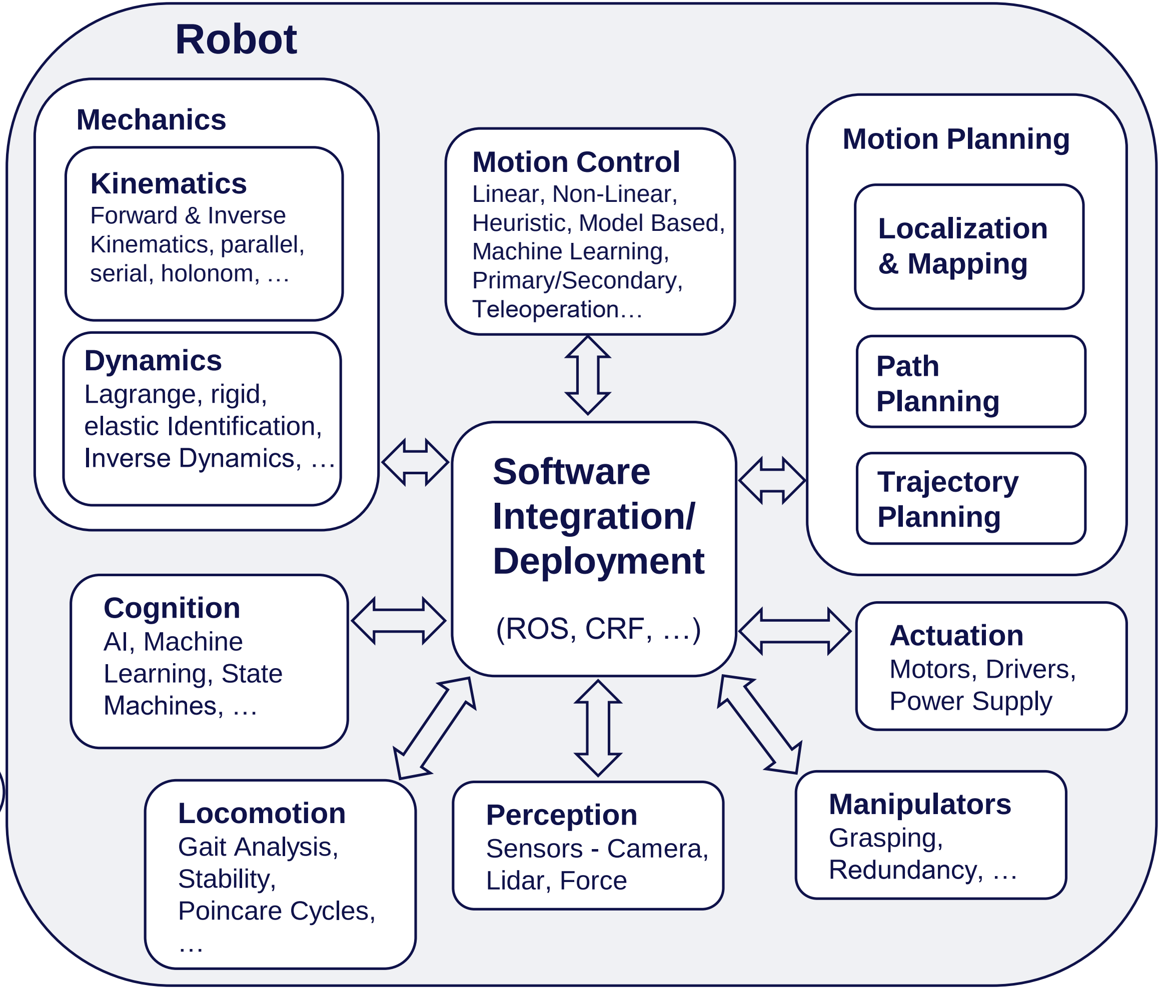
Hannes Gamper

CERN Academic Training: Robotics

Content

What is Robotics?





What is Robotics?

Mechanical Robot Design

Actuators
Motors, Drivers, Power Supply

Manipulators
Grasping, Redundancy, …

Locomotion
Gait Analysis, Stability, Poincare Cycles, …

Optimal Design
Topology, Light Weight, FEA, …

Manufacturing

CAD Design

In House Prototyping

External Manufacturing

Robot

Mechanics

Kinematics
Forward & Inverse Kinematics, parallel, serial, holonom, …

Dynamics
Lagrange, rigid, elastic, Identification, Inverse Dynamics, …

Motion Control
Linear, Non-Linear, Heuristic, Model Based, Machine Learning, Primary/Secondary, Teleoperation…

Motion Planning

Localization & Mapping

Path Planning

Trajectory Planning

Software Integration/ Deployment
(ROS, CRF, …)

Cognition
AI, Machine Learning, State Machines, …

Actuation
Motors, Drivers, Power Supply

Locomotion
Gait Analysis, Stability, Poincare Cycles, …

Perception
Sensors - Camera, Lidar, Force

Manipulators
Grasping, Redundancy, …

Content

Dynamics

- The equation of motion

$$\mathcal{L} = -\frac{1}{4}F_{\mu\nu}F^{\mu\nu} + i\bar{\psi}\not{D}\psi + h.c + \psi_i y_{ij}\psi_j\phi + h.c + |D_\mu\phi|^2 - V(\phi)$$

$$L = T - V$$

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\mathbf{q}}} - \frac{\partial L}{\partial \mathbf{q}} + \frac{\partial R}{\partial \dot{\mathbf{q}}} = \mathbf{Q}$$

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{g}(\mathbf{q},\dot{\mathbf{q}}) = \mathbf{Q}$$

$\dot{z}_l$ z_l

nonholonomic system

z_t

$\dot{z}_t$

mechanical solution

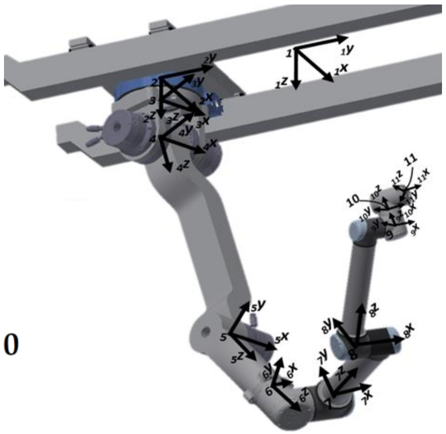
RP measurements on old LHC TDE w/ CERNbot

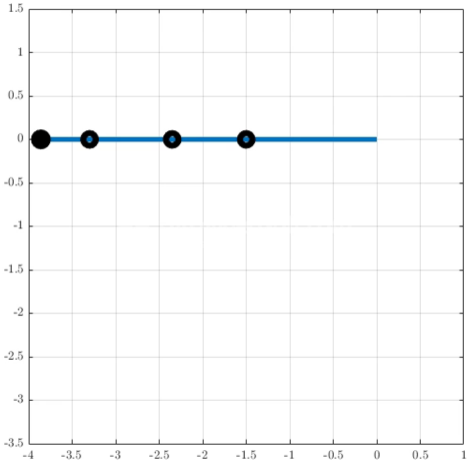
Dynamics

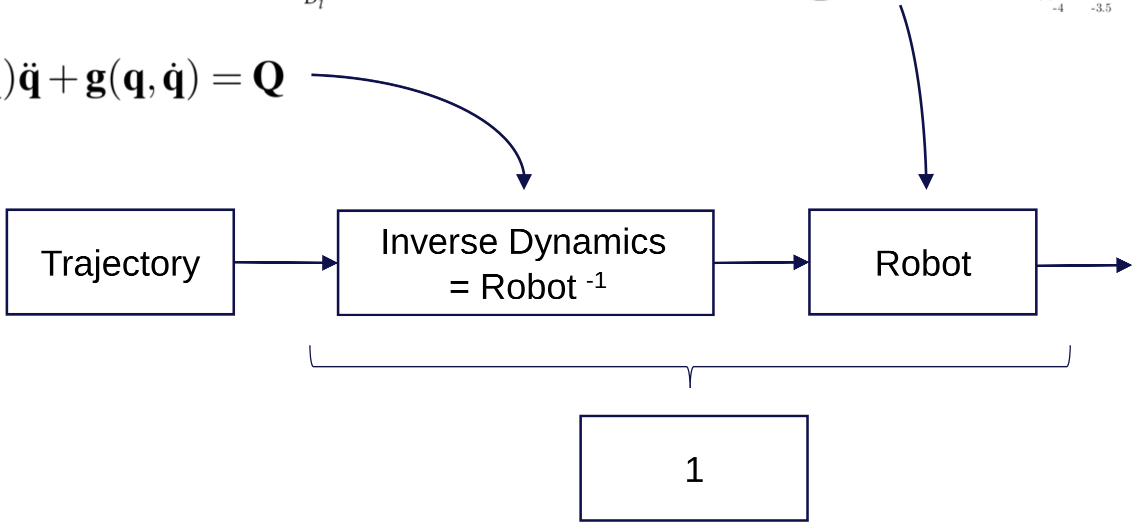
- Projection equation for more complex system

$$\sum_{i=1}^{N} \overbrace{\left[\left(\frac{\partial_R \mathbf{v}_{IS}}{\partial \dot{\mathbf{q}}} \right)^T \left(\frac{\partial_R \boldsymbol{\omega}_{IS}}{\partial \dot{\mathbf{q}}} \right)^T \right]_i}^{A_i} \underbrace{\begin{bmatrix} {}_R\dot{\mathbf{p}} + {}_R\widetilde{\boldsymbol{\omega}}_{IR}\, {}_R\mathbf{p} - {}_R\mathbf{f}^e \\ {}_R\dot{\mathbf{L}} + {}_R\widetilde{\boldsymbol{\omega}}_{IR}\, {}_R\mathbf{L} - {}_R\mathbf{M}^e \end{bmatrix}_i}_{B_i} = \mathbf{0}$$

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{g}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{Q}$$

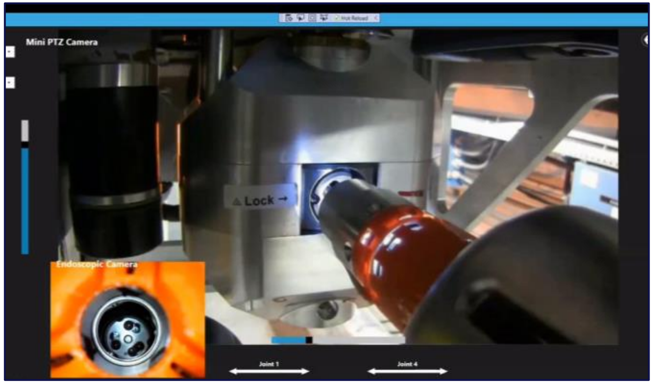






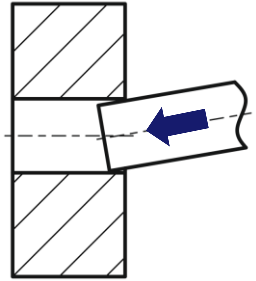
Dynamics

Example: Compliant Control

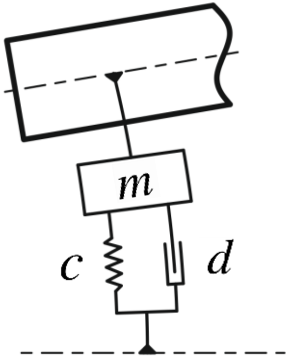


TIM Handling Radioactive Source for BLM Tests

Problem:



Solution:



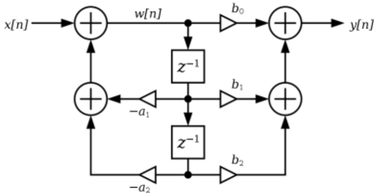
1. Mass spring damper system

$$\mathbf{M}\ddot{\mathbf{x}} = \mathbf{h} - \mathbf{C}\mathbf{x} - \mathbf{D}\dot{\mathbf{x}}$$

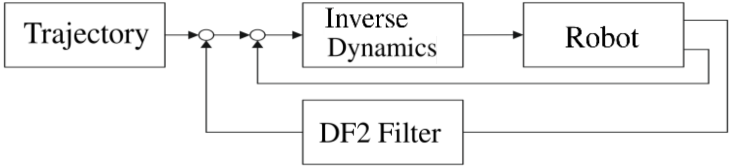
2. Discrete Frequency domain

$$\mathbf{G}(z) = \frac{\mathbf{b}_0}{1 + \mathbf{a}_1 z^{-1} + \mathbf{a}_2 z^{-2}}$$

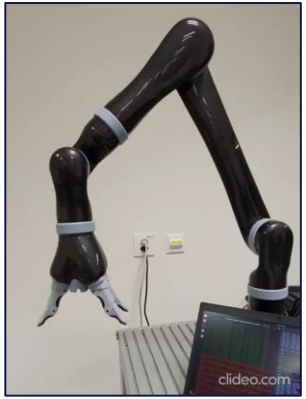
3. Implementation in C++ Framework in Direct Form II



4. Controller



Low Damping:



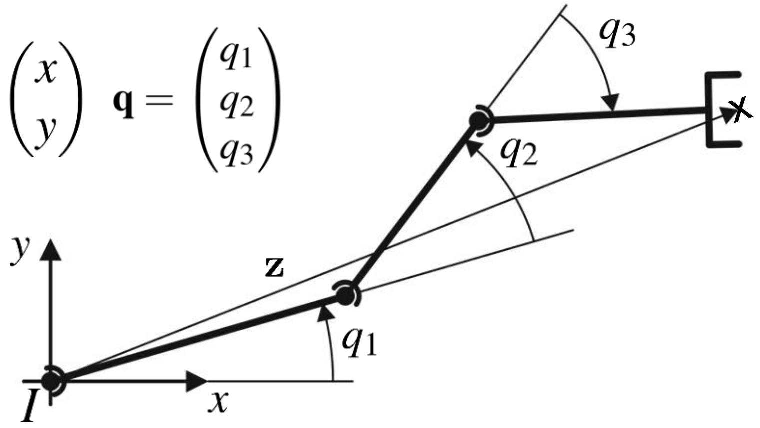
High Damping:

Content

1. **What is Robotics?**
2. **Dynamics**
3. **Kinematics**
4. **Design Optimization**

Kinematics

$$\mathbf{z} = \begin{pmatrix} x \\ y \end{pmatrix} \quad \mathbf{q} = \begin{pmatrix} q_1 \\ q_2 \\ q_3 \end{pmatrix}$$



Direct Kinematics

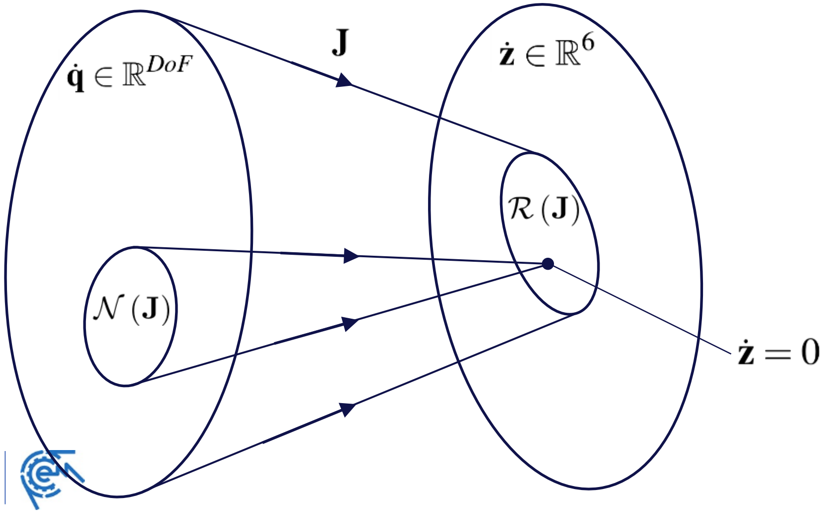
$$\mathbf{z} = \mathbf{f}(\mathbf{q})$$

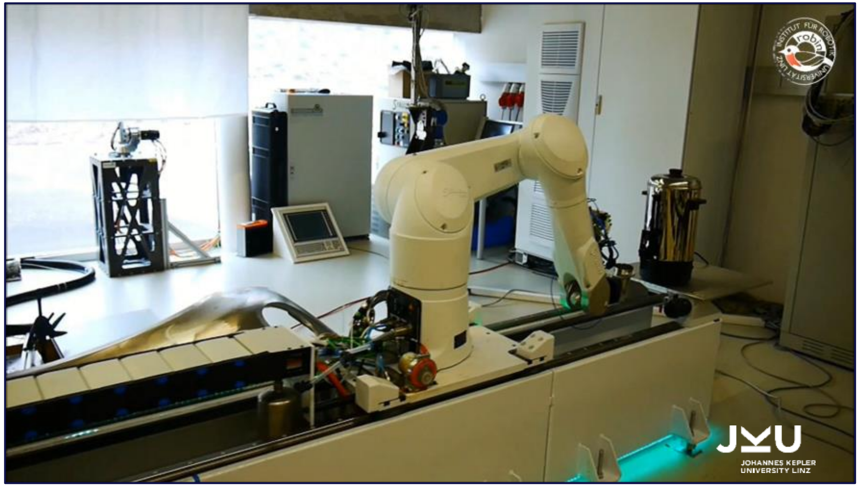
$$\dot{\mathbf{z}} = \frac{\partial \mathbf{f}(\mathbf{q})}{\partial \mathbf{q}} \frac{\partial \mathbf{q}}{\partial t} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}}$$

→ Jacobian

Inverse Kinematics

$$\dot{\mathbf{q}} = \mathbf{J}(\mathbf{q})^{-1}\dot{\mathbf{z}}$$





Inverse Kinematics

Conventions

$$\mathbf{z} = \mathbf{f}(\mathbf{q})$$

↓

$$\dot{\mathbf{z}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}}$$

↓

$$\dot{\mathbf{q}} = \mathbf{J}(\mathbf{q})^{-1}\dot{\mathbf{z}}$$

↓

Solution for $\dot{\mathbf{q}}$

↓

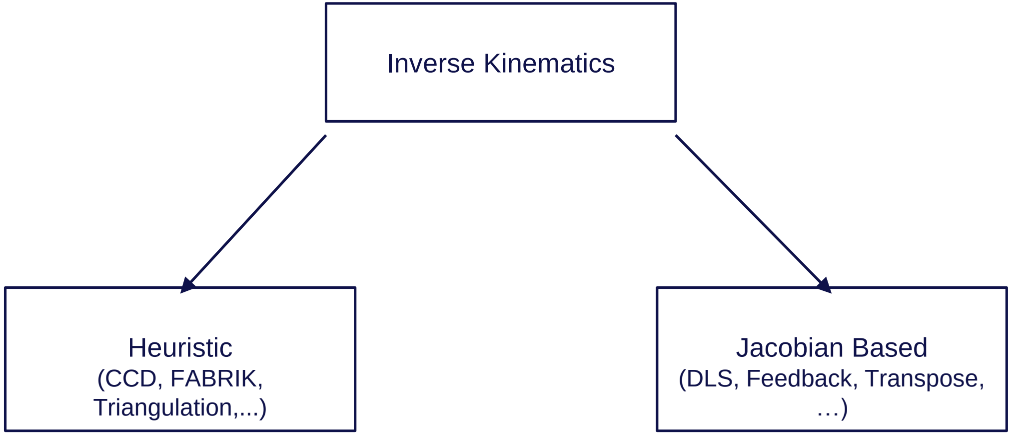
Solution for $\mathbf{q}$

↓

Def. of Error

↓

Final Algorithm



Other: Monte Carlo, Mesh-based, Model Predictive Control,…

Inverse Kinematics

Conventions

$$\mathbf{z} = \mathbf{f}(\mathbf{q})$$

↓

$$\dot{\mathbf{z}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}}$$

↓

$$\dot{\mathbf{q}} = \mathbf{J}(\mathbf{q})^{-1}\dot{\mathbf{z}}$$

↓

Solution for $\dot{\mathbf{q}}$

↓

Solution for $\mathbf{q}$

↓

Def. of Error

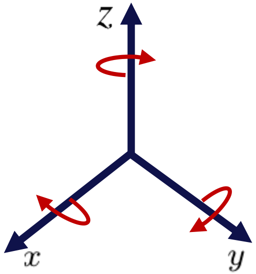
↓

Final Algorithm

Orientation

2 **Difficulties**:

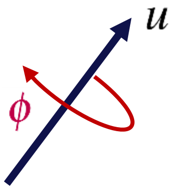
1. Rotations are not commutative!
2. Representation Singularities!



Conventions:

- **Euler**: z -> x -> z
- **Kardan**: z -> y -> x (yaw – pitch – roll)
- **Axis-Angle**: unit vector defines axis + angle
- **Quaternions:**

$$\mathrm{i}^2 = \mathrm{j}^2 = \mathrm{k}^2 = \mathrm{ijk} = -1$$



Inverse Kinematics

Conventions

$\mathbf{z} = \mathbf{f}(\mathbf{q})$

↓

$\dot{\mathbf{z}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}}$

↓

$\dot{\mathbf{q}} = \mathbf{J}(\mathbf{q})^{-1}\dot{\mathbf{z}}$

↓

Solution for $\dot{\mathbf{q}}$

↓

Solution for $\mathbf{q}$

↓

Def. of Error

↓

Final Algorithm

Jacobian Matrix

$$\mathbf{z} = \mathbf{f}(\mathbf{q})$$

$$\dot{\mathbf{z}} = \frac{\partial \mathbf{f}(\mathbf{q})}{\partial \mathbf{q}} \frac{\partial \mathbf{q}}{\partial t} = \left(\frac{\partial \dot{\mathbf{f}}(\mathbf{q})}{\partial \dot{\mathbf{q}}} \right) \dot{\mathbf{q}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}}$$

- **Analytic Jacobian**
- Contains representation singularities!

- **Geometric Jacobian**
- No representation singularities!
- Derivative can be avoided, by deriving the Jacobian via the kinematic chain!

Inverse Kinematics

Conventions

$\mathbf{z} = \mathbf{f}(\mathbf{q})$

↓

$\dot{\mathbf{z}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}}$

↓

$\dot{\mathbf{q}} = \mathbf{J}(\mathbf{q})^{-1}\dot{\mathbf{z}}$

↓

Solution for $\dot{\mathbf{q}}$

↓

Solution for $\mathbf{q}$

↓

Def. of Error

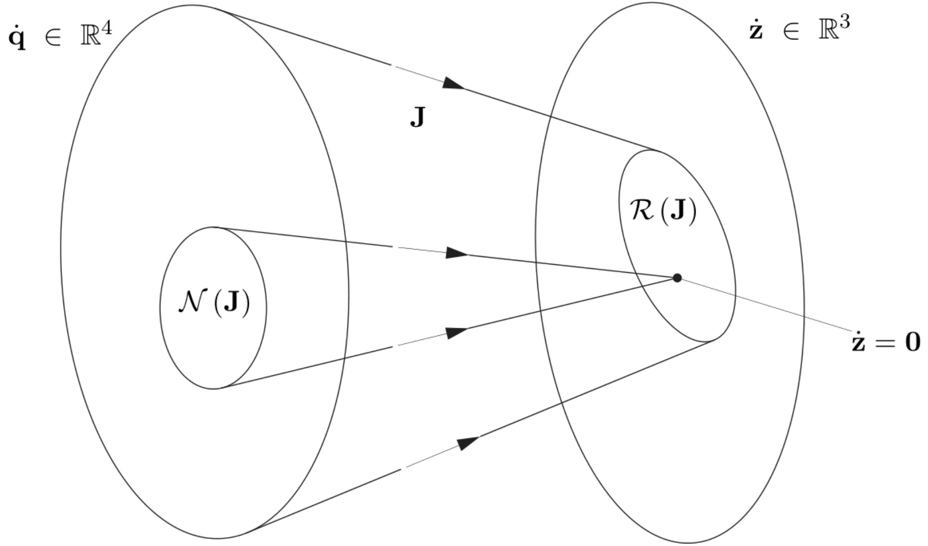
↓

Final Algorithm

Jacobian Inverse

$\mathbf{J}(\mathbf{q})^{-1}$

Redundancy



$\mathbf{J}(\mathbf{q})^{-1}$

↓

$\mathbf{J}^T(\mathbf{J}\mathbf{J}^T)^{-1}$

- Moore-Penrose pseudo inverse
- Minimizing velocities

Inverse Kinematics

Conventions

$\mathbf{z} = \mathbf{f}(\mathbf{q})$

↓

$\dot{\mathbf{z}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}}$

↓

$\dot{\mathbf{q}} = \mathbf{J}(\mathbf{q})^{-1}\dot{\mathbf{z}}$

↓

Solution for $\dot{\mathbf{q}}$

↓

Solution for $\mathbf{q}$

↓

Def. of Error

↓

Final Algorithm

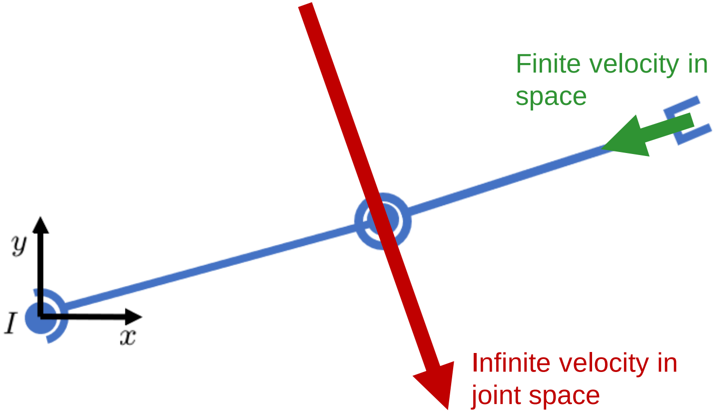
Jacobian Inverse

$$\mathbf{J}(\mathbf{q})^{-1}$$

Redundancy → $\mathbf{J}^T(\mathbf{J}\mathbf{J}^T)^{-1}$

Singularities

$$\mathbf{J}^{\dagger} = \mathbf{J}^T(\mathbf{J}\mathbf{J}^T + \alpha\mathbf{I})^{-1}$$



$$\alpha = \begin{cases} \alpha_0 \left(1 - \frac{w_{kin}}{w_0}\right)^2 & w_{kin} < w_0 \\ 0 & w_{kin} \geq w_0 \end{cases}$$

$$w_{kin}(\mathbf{q}) = \sqrt{\det\left\{\mathbf{J}(\mathbf{q})\mathbf{J}^T(\mathbf{q})\right\}},$$

Damped least-squares pseudo-inverse

Inverse Kinematics

Conventions

$\mathbf{z} = \mathbf{f}(\mathbf{q})$

↓

$\dot{\mathbf{z}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}}$

↓

$\dot{\mathbf{q}} = \mathbf{J}(\mathbf{q})^{-1}\dot{\mathbf{z}}$

↓

Solution for $\dot{\mathbf{q}}$

↓

Solution for $\mathbf{q}$

↓

Def. of Error

↓

Final Algorithm

Optimization Based 1st Order Redundancy Resolution

$$\min_{\dot{\mathbf{q}}} \quad \frac{1}{2}\dot{\mathbf{q}}^T \mathbf{W} \dot{\mathbf{q}}$$
$$\text{s.t.} \quad \mathbf{J}(\mathbf{q})\dot{\mathbf{q}} - \dot{\mathbf{z}} = 0$$

↓

$$\dot{\mathbf{q}} = \mathbf{W}^{-1}\mathbf{J}^T \left(\mathbf{J}\mathbf{W}^{-1}\mathbf{J}^T\right)^{-1} \dot{\mathbf{z}}$$

- Minimizing joint velocities
- Yields feasible solutions
- Corresponds to Moore-Penrose pseudo-inverse

Inverse Kinematics

Conventions

$\mathbf{z} = \mathbf{f}(\mathbf{q})$

↓

$\dot{\mathbf{z}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}}$

↓

$\dot{\mathbf{q}} = \mathbf{J}(\mathbf{q})^{-1}\dot{\mathbf{z}}$

↓

Solution for $\dot{\mathbf{q}}$

↓

Solution for $\mathbf{q}$

↓

Def. of Error

↓

Final Algorithm

Optimization Based 1st Order Redundancy Resolution

$$\begin{aligned} \min_{\dot{\mathbf{q}}} \quad & \tfrac{1}{2}\dot{\mathbf{q}}^T\mathbf{W}\dot{\mathbf{q}} + \nabla h(\mathbf{q})^T\dot{\mathbf{q}} \\ \text{s.t.} \quad & \mathbf{J}(\mathbf{q})\dot{\mathbf{q}} - \dot{\mathbf{z}} = 0 \end{aligned}$$

Artificial potential energy:

$$\frac{d}{dt}h(\mathbf{q}) = \frac{dh(\mathbf{q})}{d\mathbf{q}}\frac{d\mathbf{q}}{dt} = \nabla h(\mathbf{q})^T\dot{\mathbf{q}}$$

$$\dot{\mathbf{q}} = \mathbf{J}^\dagger\dot{\mathbf{z}} + \underbrace{\left(\mathbf{I} - \mathbf{J}^\dagger\mathbf{J}\right)}_{\mathbf{N}}\mathbf{W}^{-1}\nabla h(\mathbf{q})$$

- Minimizing joint velocities
- Yields feasible solutions
- Possibility of Different Objectives

Inverse Kinematics

Conventions

$\mathbf{z} = \mathbf{f}(\mathbf{q})$

↓

$\dot{\mathbf{z}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}}$

↓

$\dot{\mathbf{q}} = \mathbf{J}(\mathbf{q})^{-1}\dot{\mathbf{z}}$

↓

Solution for $\dot{\mathbf{q}}$

↓

Solution for $\mathbf{q}$

↓

Def. of Error

↓

Final Algorithm

Optimization Based 1st Order Redundancy Resolution

$$\dot{\mathbf{q}} = \mathbf{J}^{\dagger}\dot{\mathbf{z}} + \underbrace{\left(\mathbf{I} - \mathbf{J}^{\dagger}\mathbf{J}\right)}_{\mathbf{N}} \mathbf{W}^{-1}\nabla h(\mathbf{q})$$

Objective	$h(\mathbf{q})$	$\nabla h(\mathbf{q})$
Joint Limits Max	$-\mathbf{p}_L e^{c_L(q-q_{max})}$	$-\mathbf{p}_L c_L e^{c_L(q-q_{max})}$
Desired Joint Position	$-\frac{1}{2}\mathbf{c}_P\left(q - q_{des}\right)^2$	$\mathbf{c}_p\left(q_{des} - q\right)$
Distance from Singularity	$c_S\sqrt{\det\left\{\mathbf{J}(\mathbf{q})\mathbf{J}^T(\mathbf{q})\right\}}$	numerical
Collision Avoidance	$-e^{-c_C\left(dist(\mathbf{q})-d_{min}\right)}$	numerical
Torque Optimization	$\boldsymbol{\tau}^T\mathbf{T}\boldsymbol{\tau}$	numerical

Inverse Kinematics

Conventions

$\mathbf{z} = \mathbf{f}(\mathbf{q})$

↓

$\dot{\mathbf{z}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}}$

↓

$\dot{\mathbf{q}} = \mathbf{J}(\mathbf{q})^{-1}\dot{\mathbf{z}}$

↓

Solution for $\dot{\mathbf{q}}$

↓

Solution for $\mathbf{q}$

↓

Def. of Error

↓

Final Algorithm

Numeric Stabilization

$$\mathbf{q}(t) = \int_0^t \dot{\mathbf{q}}(\tau)\,\mathrm{d}\tau + \mathbf{q}(0)$$

Integration Methods, e.g. Euler will drift

$$\dot{\mathbf{e}} = \dot{\mathbf{z}}_d - \mathbf{J}\dot{\mathbf{q}}$$

Define an Error

$$\dot{\mathbf{q}} = \mathbf{J}^\dagger(\dot{\mathbf{z}}_d + \mathbf{K}\mathbf{e}) + (\mathbf{I} - \mathbf{J}^\dagger\mathbf{J})\nabla h(\mathbf{q})$$

$$\mathbf{J}\dot{\mathbf{q}} = \dot{\mathbf{z}}_d + \mathbf{K}\mathbf{e} + \underbrace{\mathbf{J}(\mathbf{I} - \mathbf{J}^\dagger\mathbf{J})}_{\mathbf{0}}\nabla h(\mathbf{q})$$

Deriving the differential Equation for e

$$\mathbf{0} = \dot{\mathbf{e}} + \mathbf{K}\mathbf{e}.$$

Diff. Equation asymptotically stable iff K > 0!

Leads to Closed Loop Inverse Kinematics - CLIK

Inverse Kinematics

Conventions

$\mathbf{z} = \mathbf{f}(\mathbf{q})$

↓

$\dot{\mathbf{z}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}}$

↓

$\dot{\mathbf{q}} = \mathbf{J}(\mathbf{q})^{-1}\dot{\mathbf{z}}$

↓

Solution for $\dot{\mathbf{q}}$

↓

Solution for $\mathbf{q}$

↓

Def. of Error

↓

Final Algorithm

Definition of an Error

Position

$$\mathbf{e}_p = \mathbf{r}_{Ed} - \mathbf{r}_E(\mathbf{q})$$

Orientation

$$\Delta\mathbf{R} = \mathbf{R}_{IE,d}\mathbf{R}_{EI}(\mathbf{q})$$

$$\mathbf{e}_o = \frac{1}{2}\left[\widetilde{\mathbf{n}}(\mathbf{q})\mathbf{n}_d + \widetilde{\mathbf{s}}(\mathbf{q})\mathbf{s}_d + \widetilde{\mathbf{a}}(\mathbf{q})\mathbf{a}_d\right]$$

with

$$\mathbf{R} = [\mathbf{n}\ \mathbf{s}\ \mathbf{a}]$$

- Orientation Error derived from Axis/Angle representation
- Using this error directly in feedback-loop is a simplification, but stability can be easily proven with Ljapunov

Inverse Kinematics

Conventions

$\mathbf{z} = \mathbf{f}(\mathbf{q})$

↓

$\dot{\mathbf{z}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}}$

↓

$\dot{\mathbf{q}} = \mathbf{J}(\mathbf{q})^{-1}\dot{\mathbf{z}}$

↓

Solution for $\dot{\mathbf{q}}$

↓

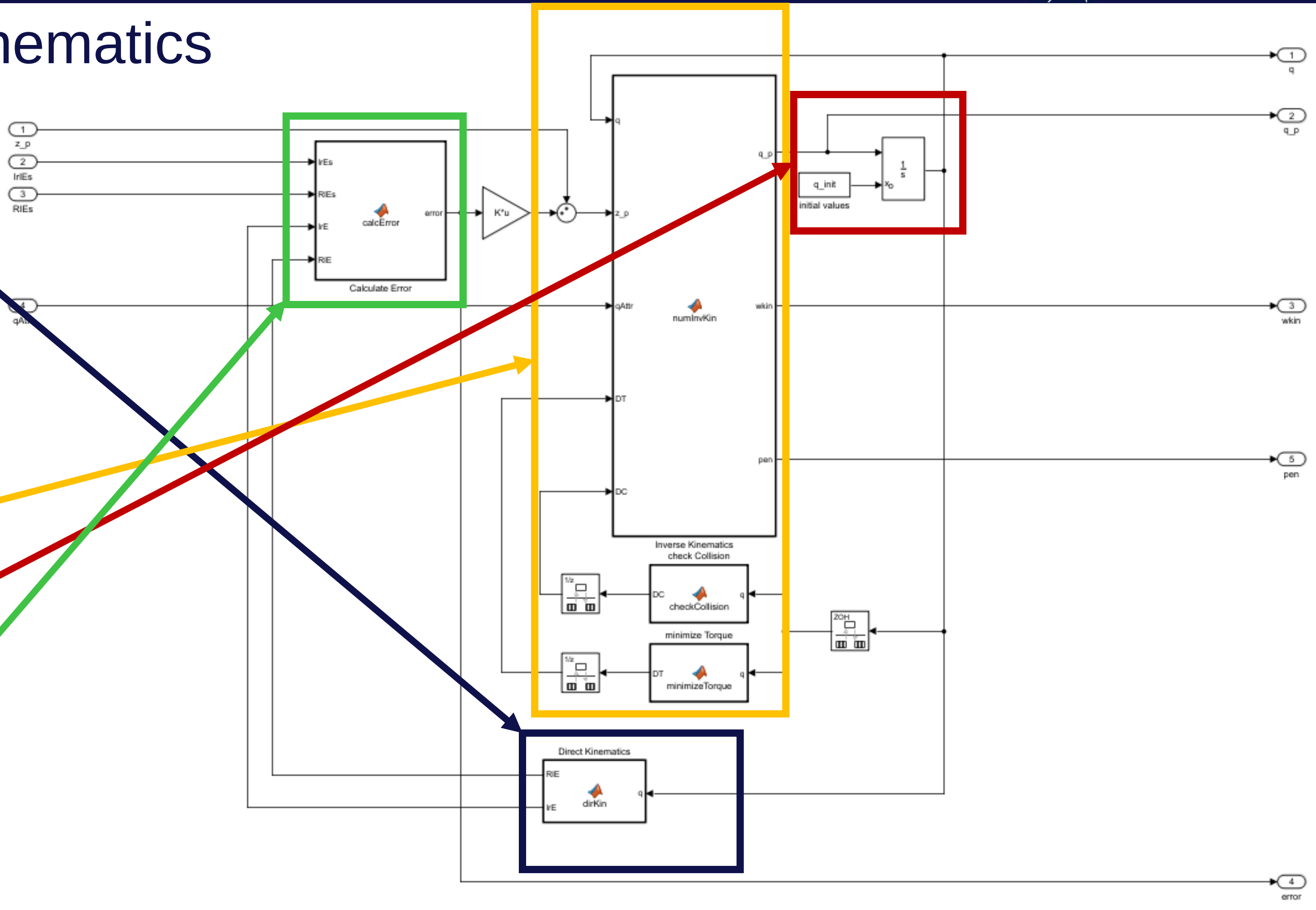
Solution for $\mathbf{q}$

↓

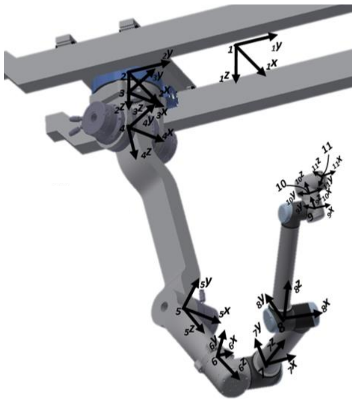
Def. of Error

↓

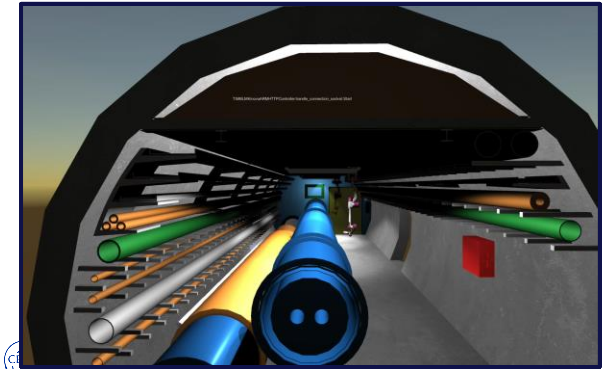
Final Algorithm



Inverse Kinematics

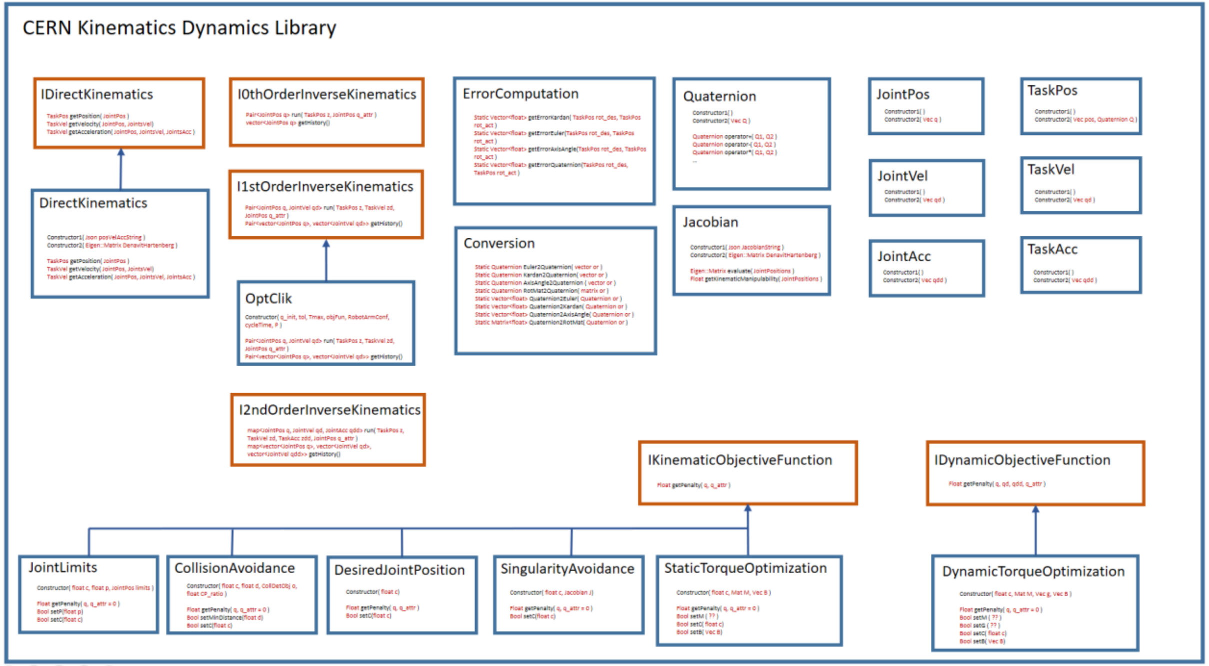


Robotic Manipulator for FCC



Kinematics Library in C++ (IK for highly redundant systems):

- Redundancy resolution for higher dexterity
- Null space projections for task priority assignment



Alejandro Diaz Rosales, Laura Rodrigo Perez

Content

Design Process

Initial Study
- Requirements
- Restrictions

↓

Integration & Logistics

↓

Design Optimization

	Task							Derived Requirements	
	Nr	Name	Description: When			Description			
			Repetitive	Periodic	T	How	Why	Description	Detailed
Maintenance	1	Measuring Radiation	yes	yes	wks - mts	Industrial Radiation Sensor measures radiation at beam level. Sensor is carried by extendable arm of TIM.	To know the risk of sending people to the tunnel or use this information as indication for other problems.	Reach Beam level in front and behind pipe; Position accuracy for repeatable measurements;	WSP: 3.2x3.6m
	2	Take Pictures	yes	yes	wks - mts	Cameras and 3D Cameras mounted on mobile platforms are taking pictures and mapping the tunnel. 3D Cameras check the geometry of the tunnel and RNN detect optical changes at the tunnel walls (e.g. new cracks, ...)	To monitor health of the tunnel; Many new cracks could indicate other problems	Stable movement; carry a camera array	
	3	Test BLM Sensors	yes	yes	wks - mts	Rough map of sensors exists, go to rough sensor position and find exact position with RNN, scan environment with depth camera to find allowed operating space, plan path with these restrictions (random points constrained by ellipse, RRT or PRM), plan smooth trajectory, bring radioactive probe to sensor, precise distance from BLM measured with additional sensor, thus approaching sensor slowly and precision of robot is not a problem	Test if Sensors are working normal, sensors are measuring radiation, higher radiation indicates beam loss which implies bigger problems and force a shutdown, should be done by robot because of radioactive sample	Reach BLM Sensors in front and behind pipe; texture of robot must allow nullspace movement to provide collision avoidance while maintaining probe position	WSP: 3.2x3.6m (1,5x1,5m -> every orientation)
	4	Measure Oxygen	yes	yes	wks - mts	Industrial Oxygen measurement sensor; measured throughout the whole tunnel;	Make sure its save for people to work down there	Reach different hights to maesure oxygen	WSP: 3.2x3.6m
	5	Measure Alignment	yes	yes	wks - mts	Strings are placed by STI on fixed mounted sockets in tunnel, TIM goes to strings as reference and measures some indicators on the Collimators & Dipoles, same distance => align! New method will automatically place the strings: outter robots hold the string and inner robots does the alignment measurement, same procedure for horizontal distance, for vertical distance new ultrasonic sensor is used	With non-align tubes, beam would get lost.	Version1 : (manually placed string) reach string with existing technology; stable movement; Version2: (automotively placed string) need of two outter robots and one inner robot with existing technology	
	6	Audio Inspection	yes	yes	wks - mts	Microphone is carried through whole tunnel, detect unusual noise (e.g. frequency domain -> 100Hz peak	To detect unusual noise which can indicate other problems (e.g.		

Design Process

Importance: 100% / 80% / 60%

Rating: 4 very positive for solution / 3 positiv for solution / 2 negativ for solution / 1 very negativ for solution

Restrictions: Tunnel & Environment / Tasks / General

Initial Study
- Requirements
- Restrictions

↓

Integration & Logistics

↓

Design Optimization

Task							Derived Requirements	
Nr	Name	Description					Description	Detailed
		When			How	Why		
		Repetitive	Periodic	T				

REQ / SOL	Geometry Req.: Workspace ~3200x3600	Geometry Req.: Go through fire doors	Geometry Req.: Collission avoidance	Power Supply: Maintenance	Power Supply: Reliability	Power Supply: Costs	Power Supply: Consumption	Communication: Easy Teleoperation	Communication: Robust	Maneuverability: Stable Movement	Maneuverability: Turn within 1500mm	Maneuverability: Go under or behind pipe	Radiation: No further cont. by rob	Radiation: Radiation Tolerant	Control: Simple	Control: Robust	Control: proofen Stability	Control: Autonomous	Emergency: Not Blocking Exit	Emergency: Move in harsh env.	Emergency: Modular	Emergency: Intervention	Summed up Rating in %
	1	0.8	1	0.8	1	1	0.6	1	1	0.8	1	0.8	1	0.8	0.8	1	1	0.8	1	0.8	0.6		
Mobile Robot (holonomic)	2	3	2	2	2	3	4	2	3	1	4	4	3	1	4	4	4	1	1	1	2		65.7
Mobile Robot (non-holonomic)	2	3	2	2	2	3	4	2	3	3	2	2	3	1	4	4	4	1	1	1	2		64.8
Rail Guided Robot (ceiling)	3	3	3	3	3	1	2	3	3	4	4	1	4	1	3	4	4	2	4	3	2		74.3
Drone	4	3	1	2	2	3	3	2	2	1	4	3	1	1	2	2	3	1	3	4	1		58.1
Legged Robot (ANYbotics, ...)	3	2	2	2	2	1	2	2	2	1	4	3	3	1	1	2	2	2	1	3	2		52.0
Legged Robot + Wheels (Boston Dynamics, ...)	3	2	2	2	2	1	2	2	2	2	4	2	3	1	1	2	2	2	1	3	2		53.7
Holonomic Robot travel in Hyperloop	2	3	2	2	2	2	2	2	3	4	4	4	3	1	2	4	4	2	1	1	1		62.0
Rail Guided Robot w. robotic arm & hol. Robot	3	3	3	2	3	1	2	3	3	4	4	4	3	1	3	4	4	2	4	3	2		75.4
Holonomic Robot w. Robotic Arm	3	3	2	2	2	3	4	2	3	1	4	4	3	1	3	4	4	1	1	1	3		63.9
RailGuided Robot w. Snake Robotic Arm	3	3	3	3	3	1	2	3	3	4	4	2	4	1	3	4	4	2	4	3	2		8 71.3

Design Process

Initial Study
- Requirements
- Restrictions

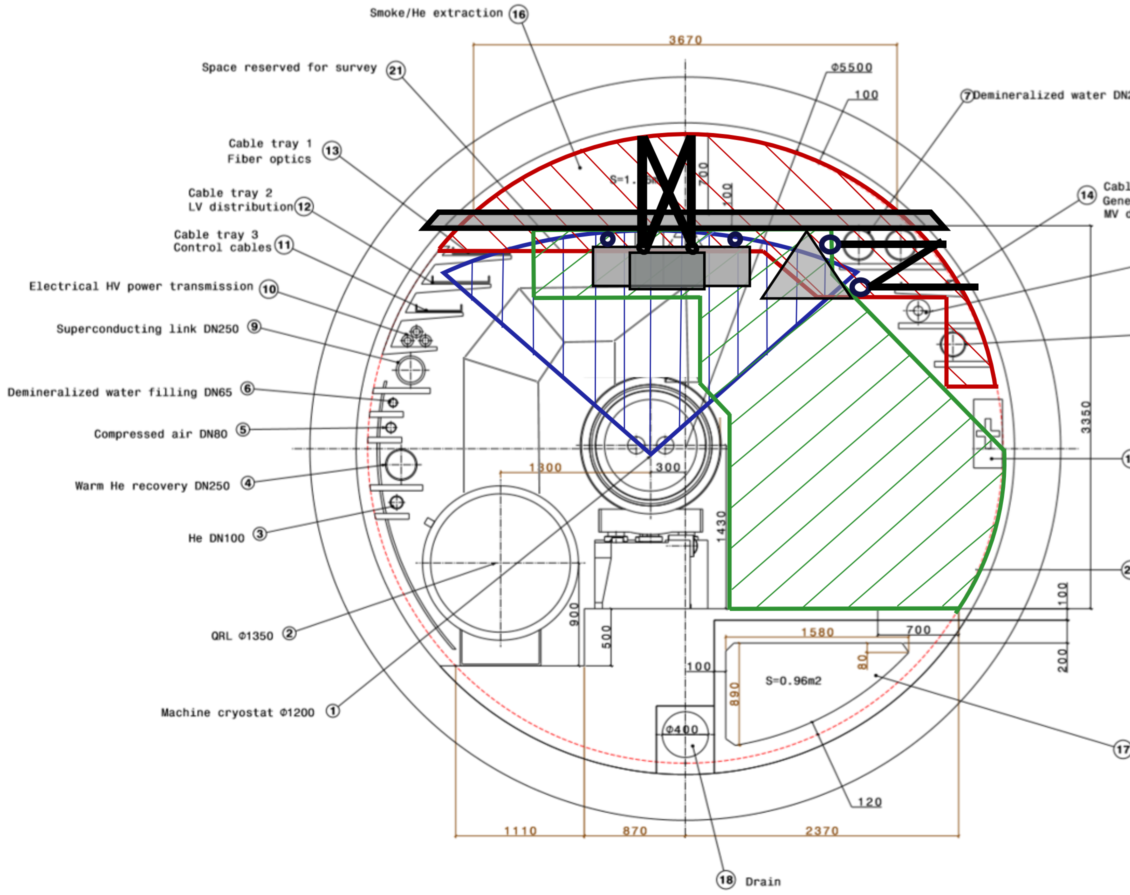
↓

Integration & Logistics

↓

Design Optimization

- Good Access for Inspections
- Allowed Area for moving Robot
- Allowed Area for permanently installed material



Design Process

Initial Study

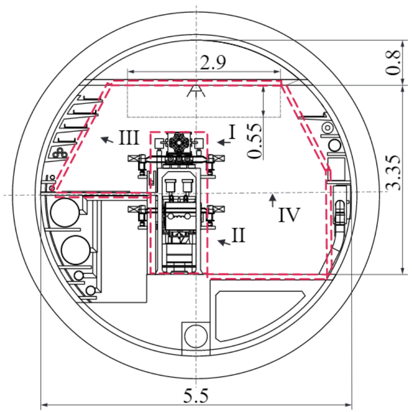
- Requirements
- Restrictions

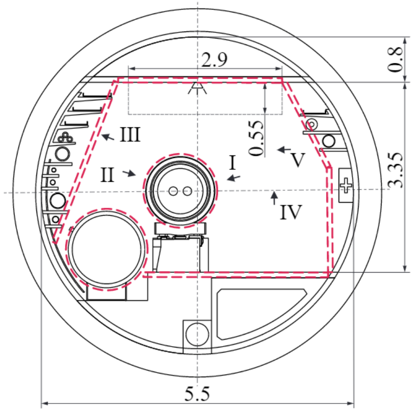
↓

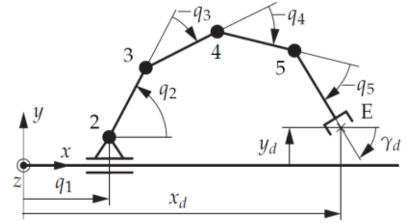
Integration & Logistics

↓

Design Optimization







Requirements:

- Space:
  - Reach points I-IV (workspace of 3.35x5.50x10^5m)
  - Pack up in limited space (2.9x0.55m) while moving along tunnel axis
- Avoid Obstacles
- Mass:
  - Min. payload (~15kg)
  - Max. robot weight (~300kg)

Gamper, H.; Gattringer, H.; Müller, A. and Di Castro, M. (2021). Design Optimization of a Manipulator for CERN's Future Circular Collider (FCC). In Proceedings of the 18th International Conference on Informatics in Control, Automation and Robotics, ISBN 978-989-758-522-7

Design Process

Optimization Goals:

- Min. the Degree of Freedom (DoF)
- Min. the Robot Link Lengths
- Min. kinematic and dynamic perf. criteria

Initial Study

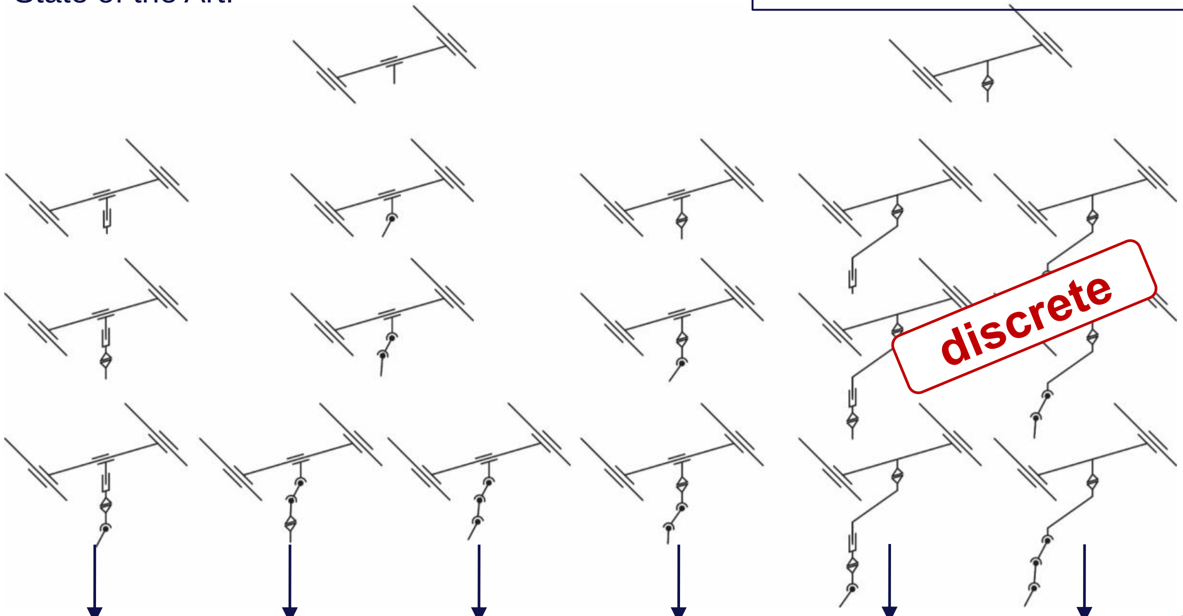
Requirements

Restrictions

Integration & Logistics

Design Optimization

State of the Art:



„Topology Optimization“

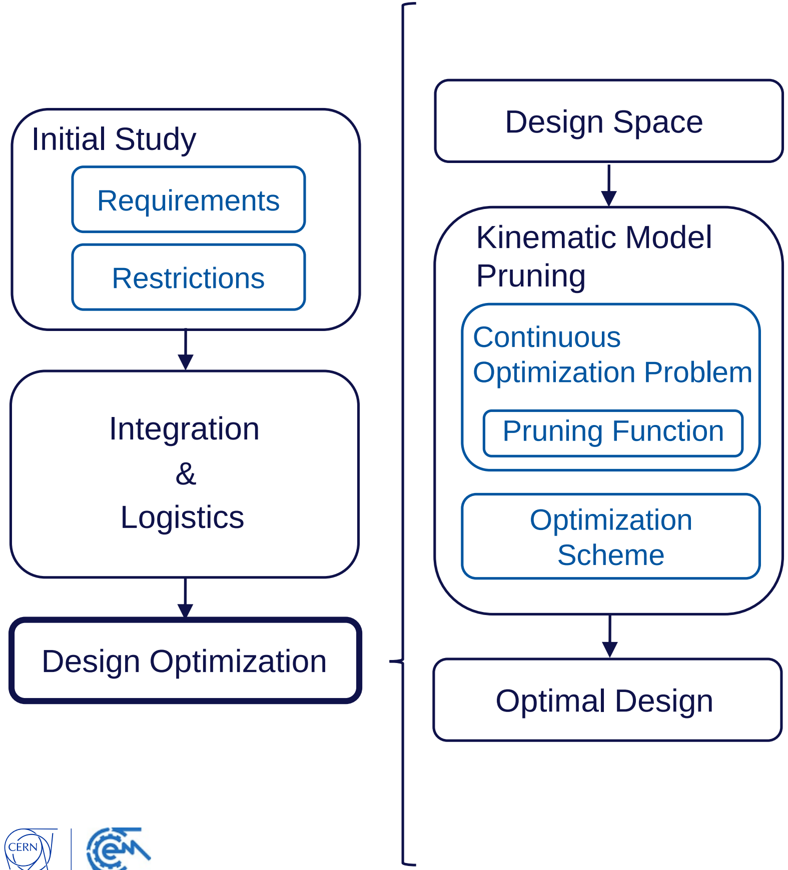
Optimization of Geometry according to Performance Criteria

continuous

„Geometry Optimization“

Optimal Design

Design Process



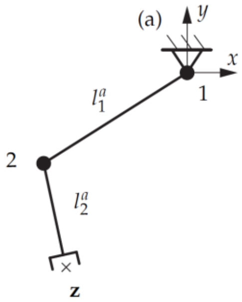
<u>Optimization Goals:</u>

- Min. the Degree of Freedom (DoF)
- Min. the Robot Link Lengths
- Min. kinematic and dynamic perf. criteria

Design Optimization

Optimization Goals:
- Min. the Degree of Freedom (DoF)
- Min. the Robot Link Lengths
- Min. kinematic and dynamic perf. criteria

Design Space
↓
Kinematic Model Pruning
- Continuous Optimization Problem
  - Pruning Function
- Optimization Scheme
↓
Optimal Design



Length	[mm]
l_1	144.6
l_2	80.3

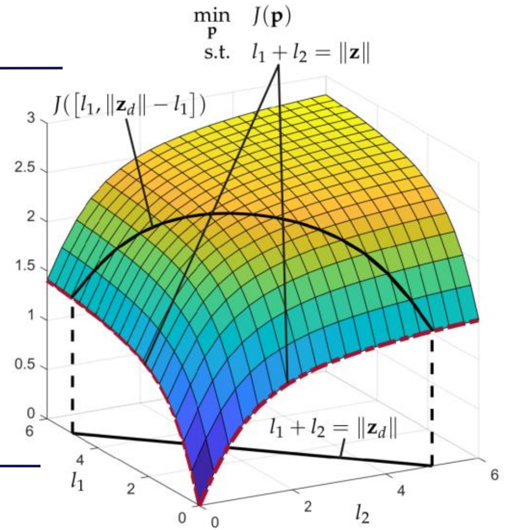
$$\min_{\mathbf{x},\,\mathbf{p}} \; J(\mathbf{p}) \quad \text{s.t.} \quad \mathbf{f}(\mathbf{x},\,\mathbf{p}) - \mathbf{z}_d = \mathbf{0}$$

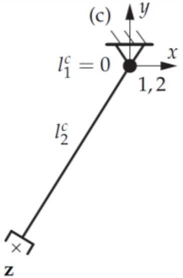
Definition 1 (Pruning Function). *A vector function* $\mathbf{g} = [g_1(l_1) \;\; g_2(l_2) \;\; \dots \;\; g_N(l_N)] : \mathbb{R}^N \to \mathbb{R}^N$ *with argument* $\mathbf{p} = [l_1 \;\; l_2 \;\; \dots \;\; l_N]^T \in \mathbb{R}^N$ *that satisfies*

$$\frac{\partial \mathbf{g}}{\partial \mathbf{p}} > 0 \;\; \forall \;\; l_i > 0, \;\; i \in \{1, 2, \dots, N\} \tag{1}$$

and

$$\frac{\partial^2 \mathbf{g}}{\partial \mathbf{p}^2} < 0 \;\; \forall \;\; l_i > 0, \;\; i \in \{1, 2, \dots, N\}. \tag{2}$$





Length	[mm]
l_1	0
l_2	167

Design Process

Example: FCC Robot

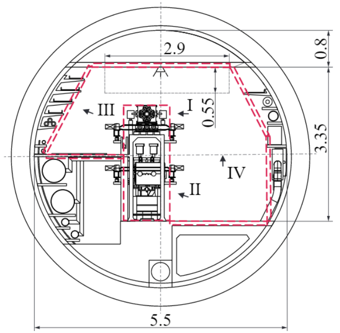
Initial Study
- Requirements
- Restrictions

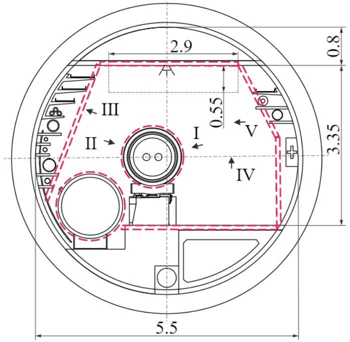
↓

Integration & Logistics

↓

Design Optimization





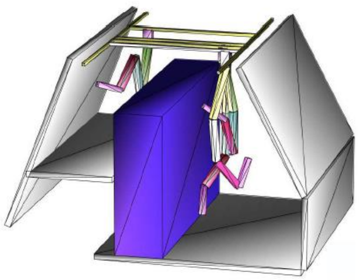


Figure 10: Optimization results FCC-ee (collision objects)

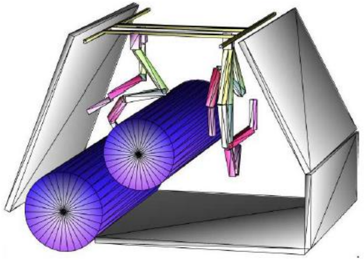


Figure 11: Optimization results FCC-hh (collision objects)

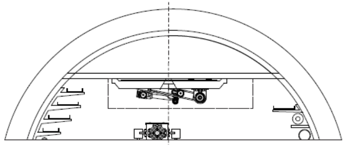


Figure 13: Prototype in FCC-ee - folded configuration

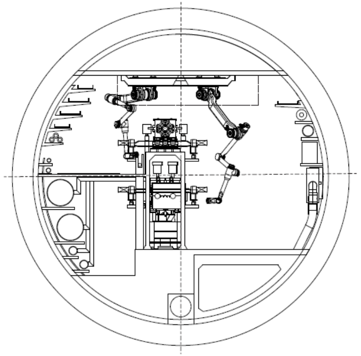


Figure 14: Prototype in FCC-ee cross-section

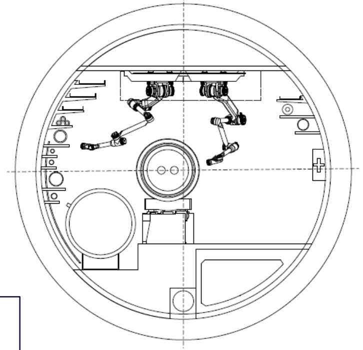


Figure 15: Prototype in FCC-hh cross-section

<u>Optimal Solution:</u>

- Optimal Geometric Parameters (link lengths)
- Optimal Topology (11 DoF, Joint Configuration)

Gamper, H.; Gattringer, H.; Müller, A. and Di Castro, M. (2021). Design Optimization of a Manipulator for CERN's Future Circular Collider (FCC). In Proceedings of the 18th International Conference on Informatics in Control, Automation and Robotics, ISBN 978-989-758-522-7

Design Process

Gamper, H.; Luthi, A.; Gattringer, H.; Müller, A. and Di Castro, M.; Design Optimization of Quality Inspection Robots for Particle Accelerator Components, In Proceedings of the ECCOMAS Multibody Dynamics Conference, 2021

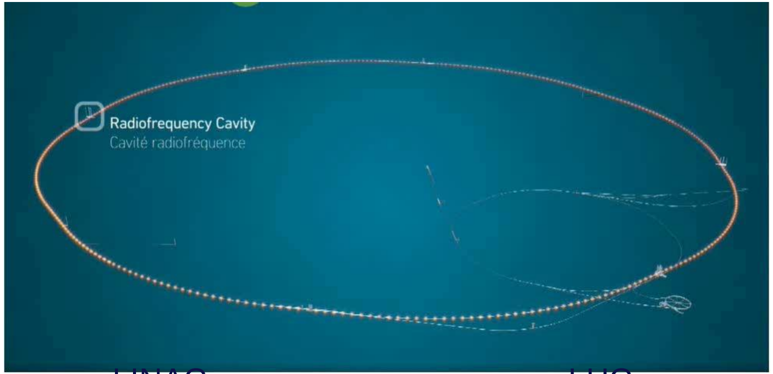
Initial Study
- Requirements
- Restrictions

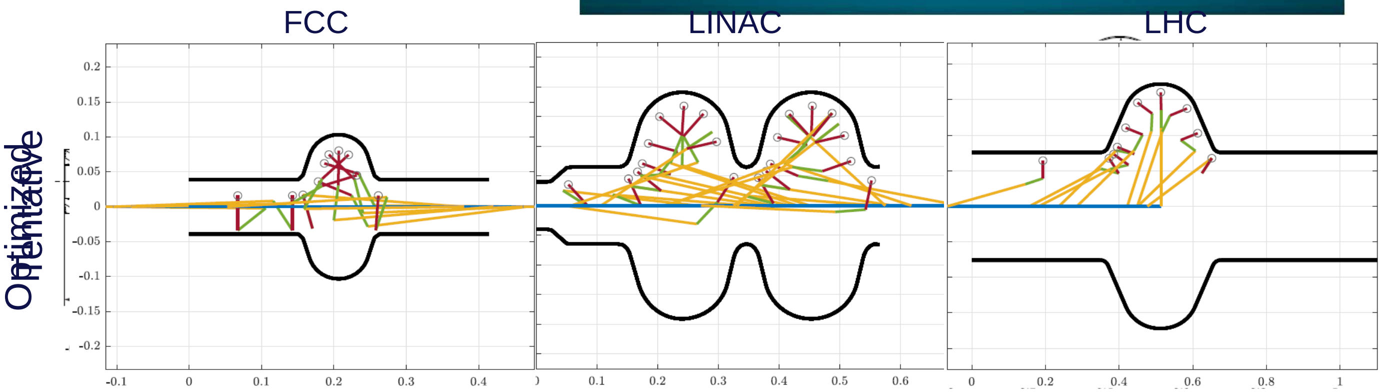
Integration & Logistics

Design Optimization

Example: Cavity Design Optimization

- Visual inspection of inner surface after assembly
- Small allowed robot space
- Big workspace for inspection
- No restrictions on robot design (topology, geometry)





Design Process

Gamper, H.; Luthi, A.; Gattringer, H.; Müller, A. and Di Castro, M.; Design Optimization of Quality Inspection Robots for Particle Accelerator Components, In Proceedings of the ECCOMAS Multibody Dynamics Conference, 2021

Initial Study
- Requirements
- Restrictions

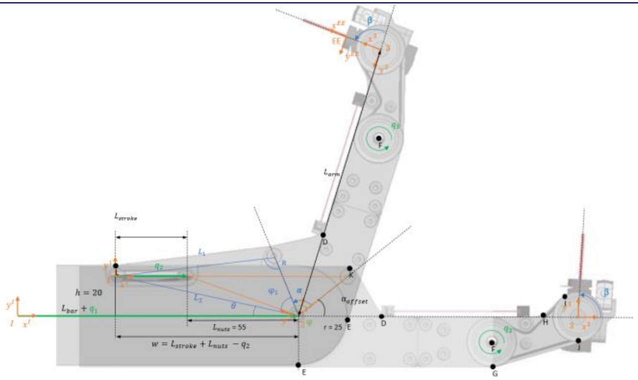
↓

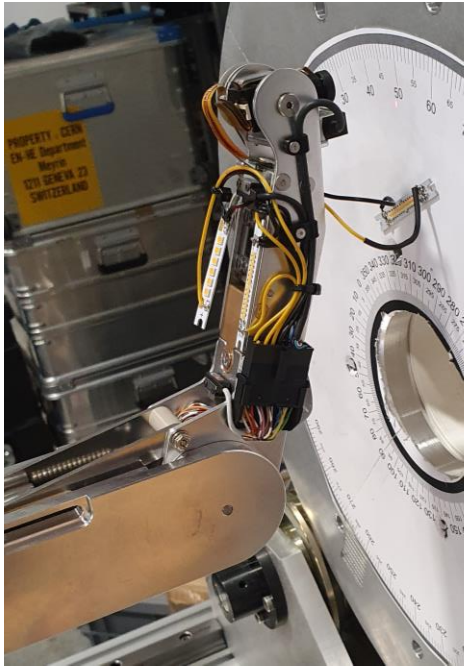
Integration & Logistics

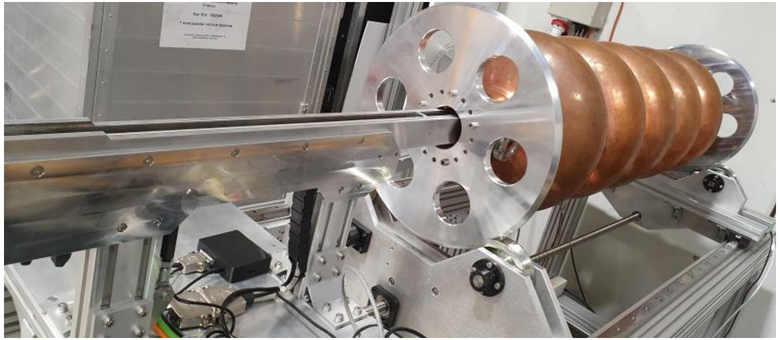
↓

Design Optimization

Example: Cavity Design Optimization







Controls
Electronics &
Mechatronics

Thank you
for your attention!