

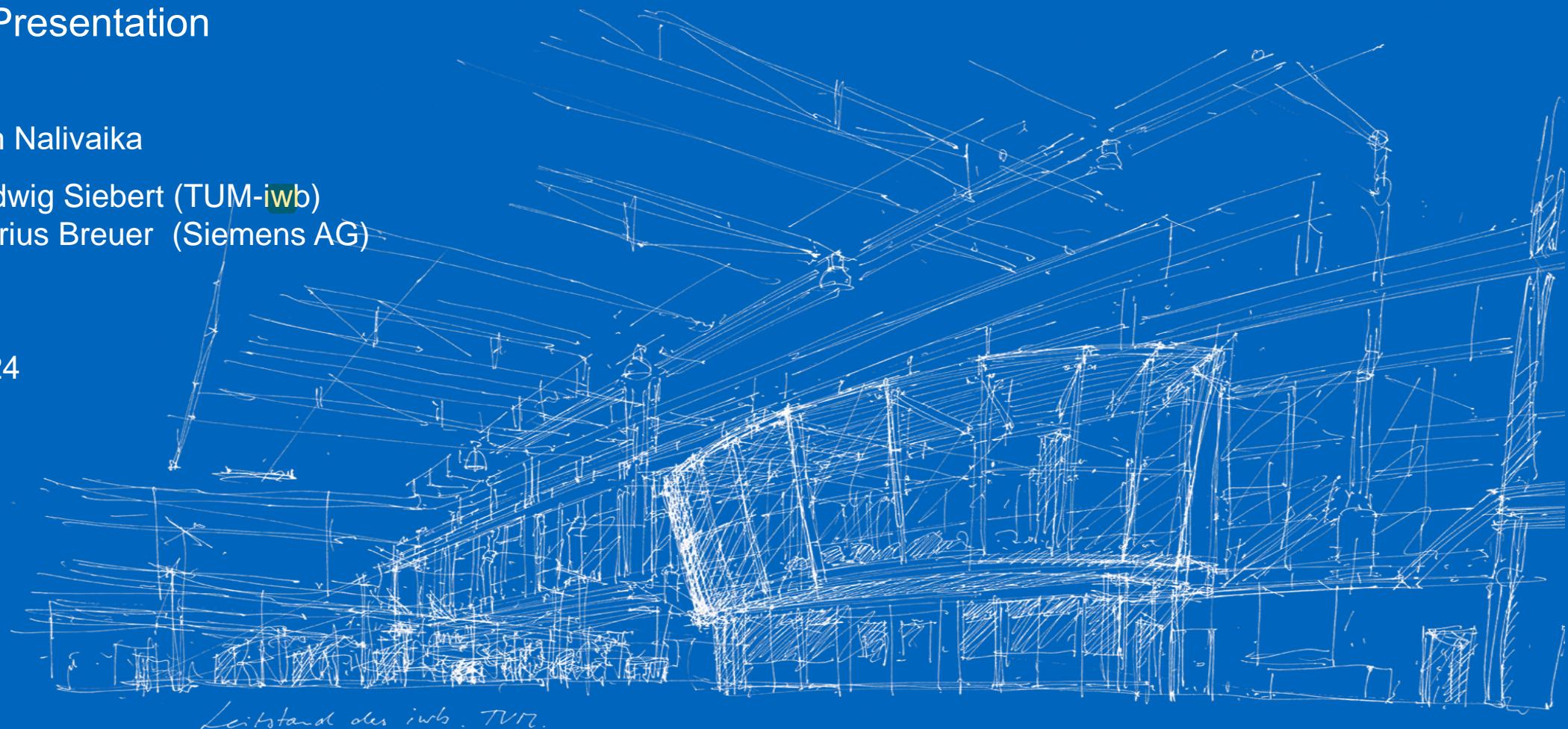
# Methodical Approach for Analyzing Process Variables and Optimizing Boundary Conditions in Multi-Axis Robot Programs

Master-Thesis Presentation

Submitted by: Jan Nalivaika

Supervised by: Ludwig Siebert (TUM-iwb)  
Marius Breuer (Siemens AG)

Garching 04.03.2024

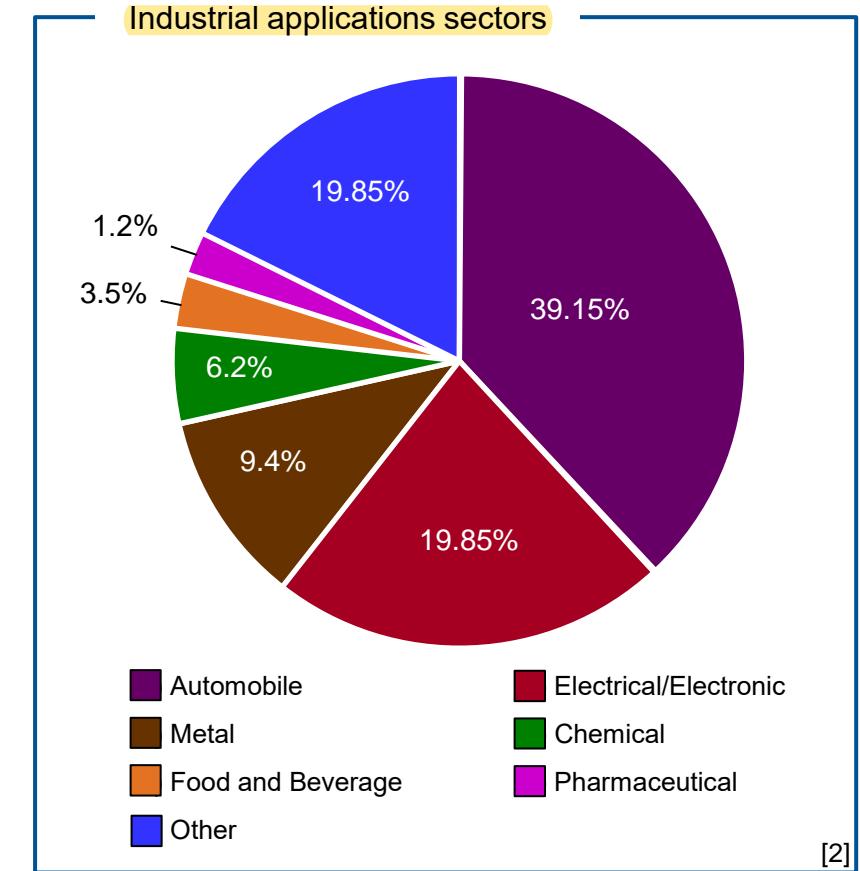
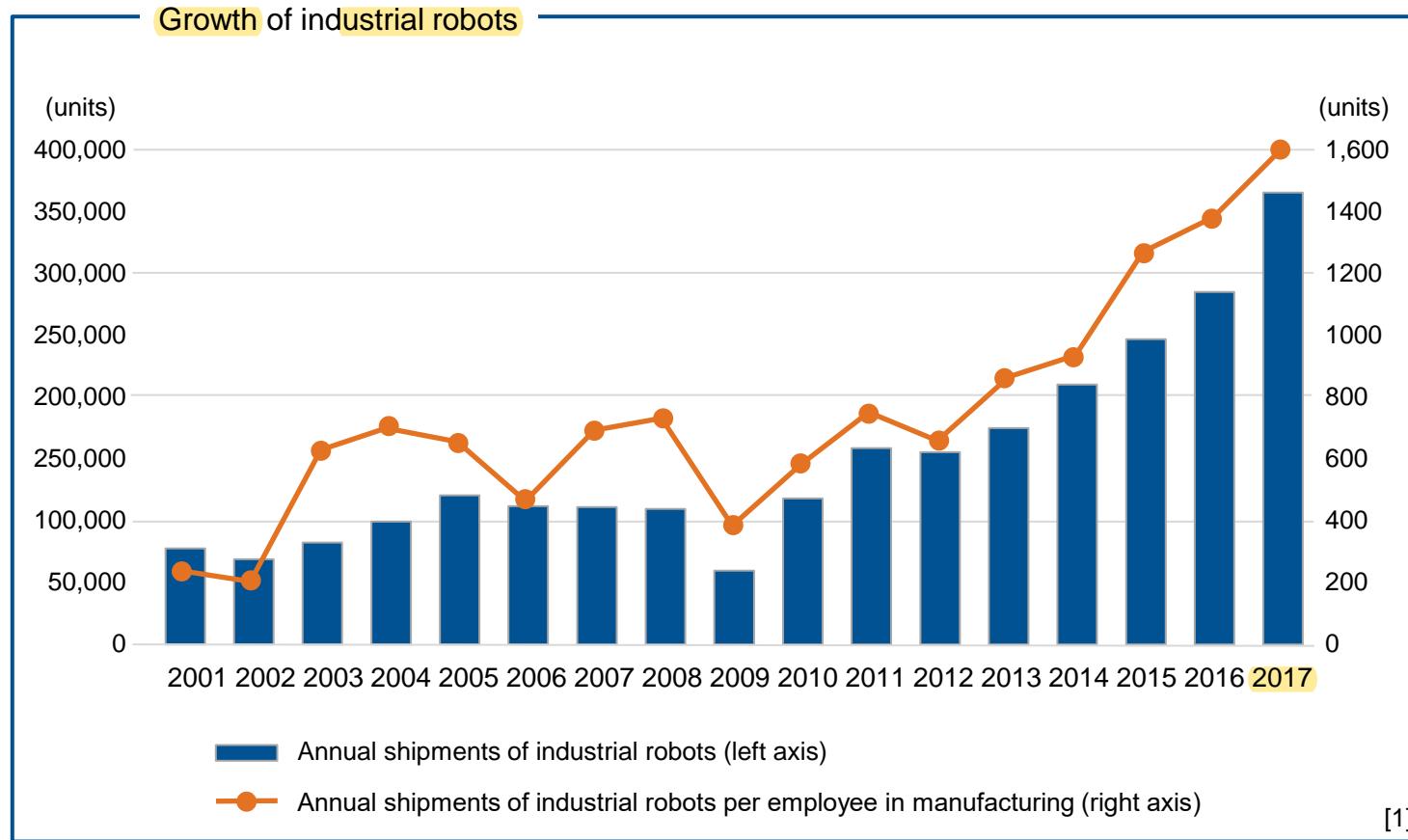


# Table of Contents

- 1 Motivation
- 2 Problem Formulation and Aim
- 3 State of the Art
- 4 Methodology
- 5 Implementation and Results
- 6 Summary and Outlook

# Growth of Industrial Robots and Applications

## Motivation

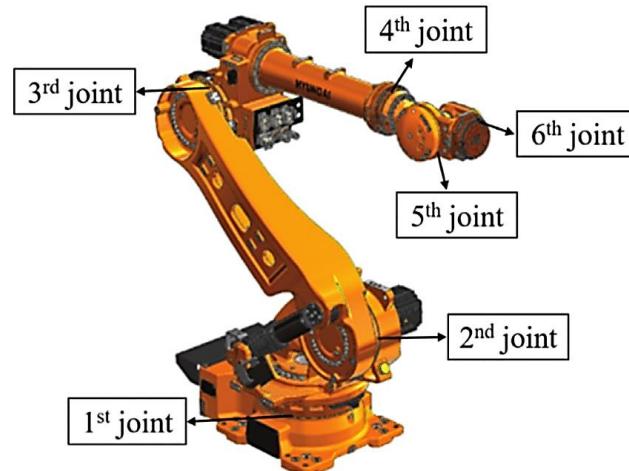


## Motivation



### CNC machines

- Designed for milling
- High precision
- Toolpaths in 5 DoFs
- X, Y, Z (translation)
- A, B (rotation)



### Industrial robots

- 6 DoFs
- Used for WAAM
- Can be used for milling
- Highly versatile
- Sub-optimal stiffness

### Milling with robots

- 5-DoF toolpath does not fully specify the robots' configuration
- Only X, Y, Z, A and B are defined
- Rotation C is not specified and **can be chosen freely**
- Rotation C is affecting how a toolpath is traversed
- Rotation C is considered **a boundary condition**

### Influence of boundary condition

- ➔ Redundancy in the system when performing a 5-DoF task in a 6-DoF system
- ➔ How does the boundary condition affect the robot?
- ➔ Goal: Optimizing the robots' **behavior** by **optimally** setting the boundary condition

# Example: Traversing a Toolpath With Different C-Rotation

## Problem Formulation and Aim

Rotation C = 0°

**THIS IS A VIDEO**

Add video of robot here

Add video of NX here (if robot not available)

Rotation C = -30°

**THIS IS A VIDEO**

Add video of robot here

Add video of NX here (if robot not available)

- ➔ Rotation C can be chosen freely
- ➔ The rotation C significantly influences the behavior of the robot

### Influenced process variables

Direction changes in the joints; Accelerations;  
Energy consumption; Cable positioning; Stiffness; etc.

# Shortcomings in Current Methodologies

## Problem Formulation and Aim

### Problem 1

No currently published method allows the user to select specific process variables to evaluate a process

### Problem 2

As of now, it is not possible to weigh individual process parameters and thus describe the manufacturing process as a singular scalar value

### Problem 3

No available method provides a solution to optimize boundary conditions based on a user defined goal

### Aim 1

Provide a method that can work with specifically selected process variables and rate a manufacturing process

### Aim 2

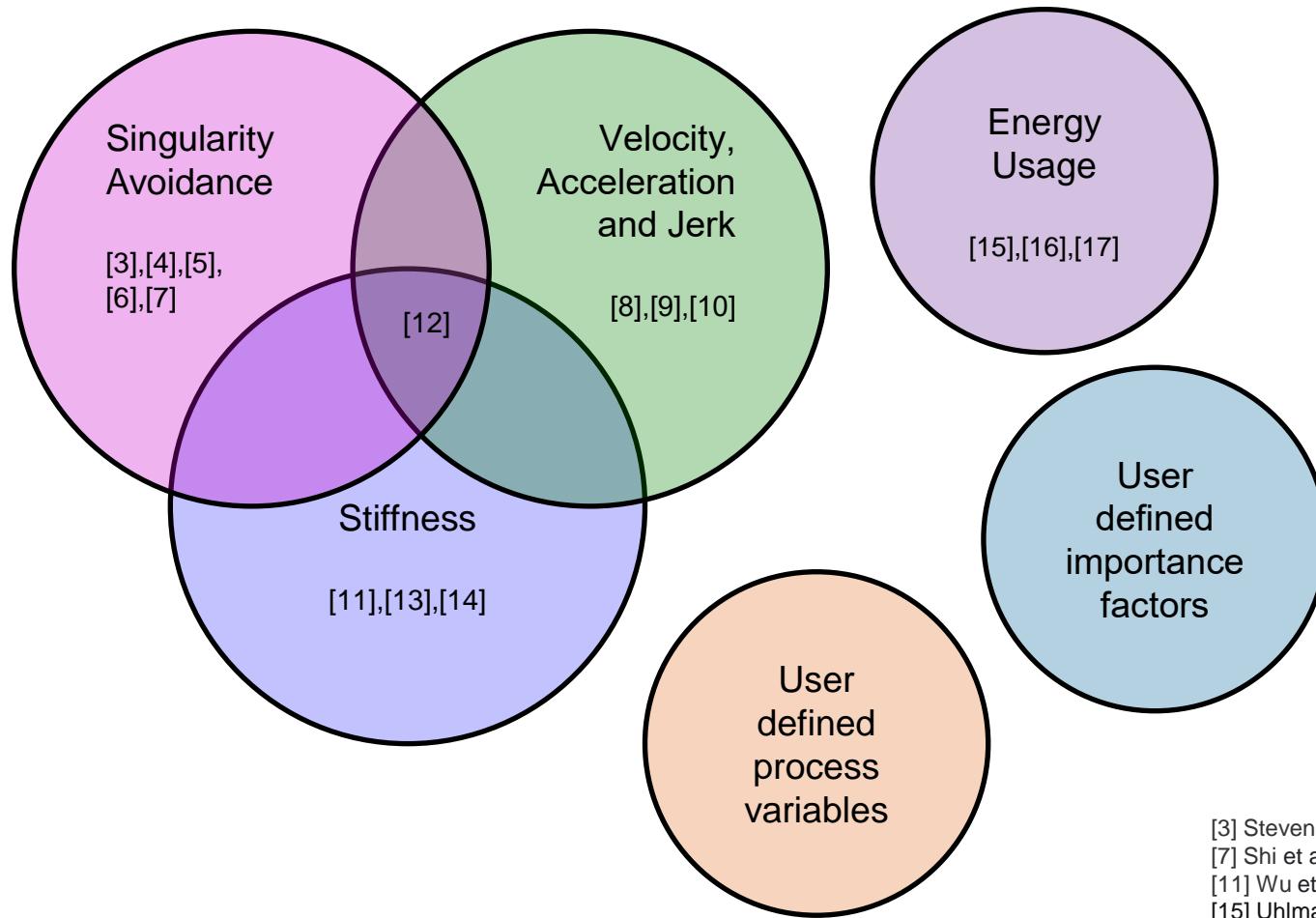
Extend the developed method with the option to add user-defined importance factor (weights) for the process variables

### Aim 3

Provide system that can optimize the boundary conditions while considering the user defined goal

# Elaboration of the Research Gap

## State of the Art



## Research Gap

- marginal overlap in the publications
- no method takes a holistic approach

## Missing elements

- No possibility for the user to select process variables
- No possibility for the user **to weigh** the individual process variables
- No possibility to optimize the setting of redundant DoFs towards a user-defined goal

[3] Stevenson et al., 2002  
[7] Shi et al., 2021  
[11] Wu et al., 2021  
[15] Uhlmann et al., 2016

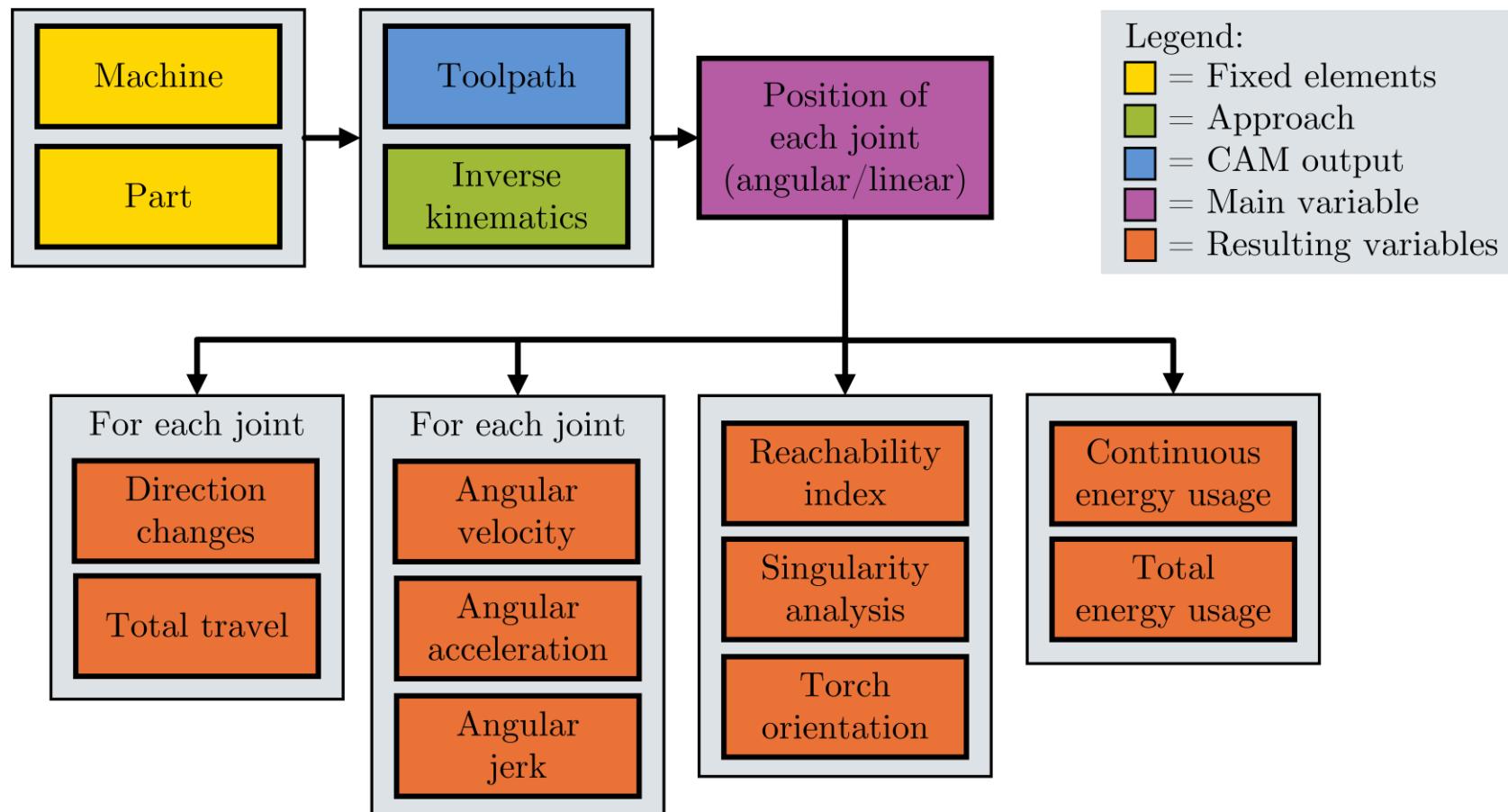
[4] Bedrossian, 1990  
[8] Dai et al., 2020  
[12] Xiong et al., 2019  
[16] Boscaroli et al., 2020

[5] Huo and Baron, 2008  
[9] Jiang et al., 2017  
[13] Wang et al., 2022  
[17] Soori et al., 2015

[6] Milenkovic, 2021  
[10] Bien Duong 2021  
[14] Cvitanic et al., 2020

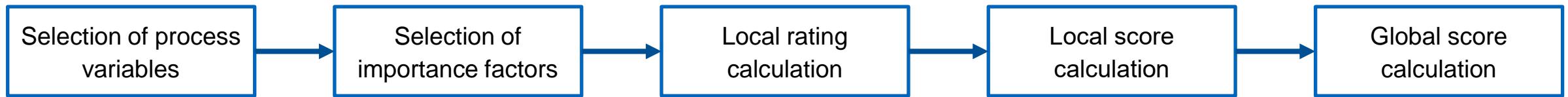
# Extraction of Process Variables

## Methodology

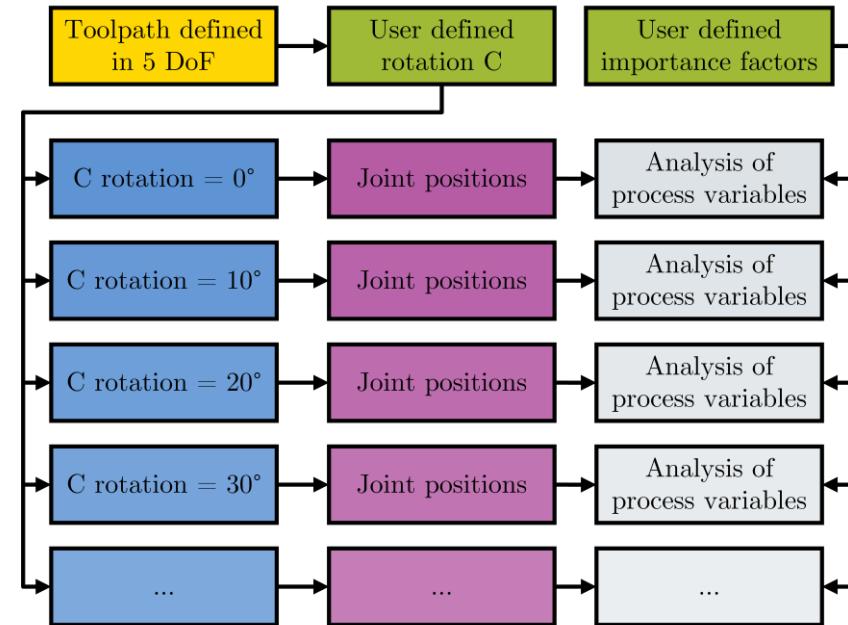
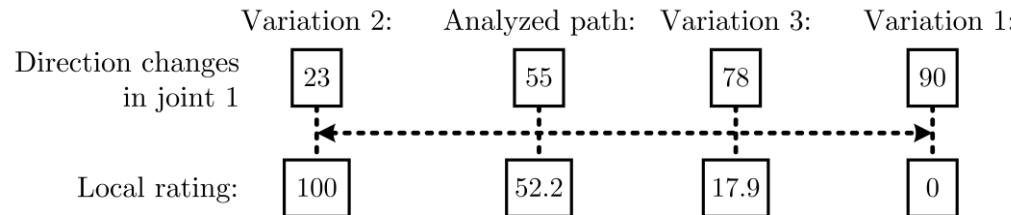


# User Defined Importance Factors and Evaluation

## Methodology

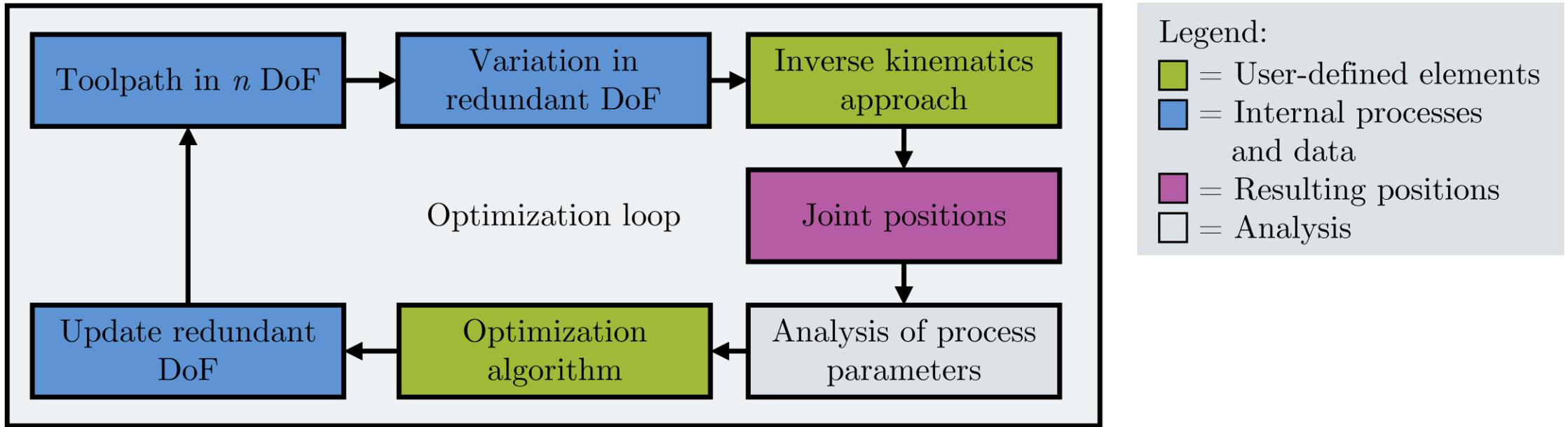


Process variable	Importance factor	Local rating	Local score
Process variable Nr. 1	0.5	74	37
Process variable Nr. 2	0.1	34	3.4
Process variable Nr. 3	0.1	65	6.5
Process variable Nr. 4	0.3	22	6.6
Global score			53.5



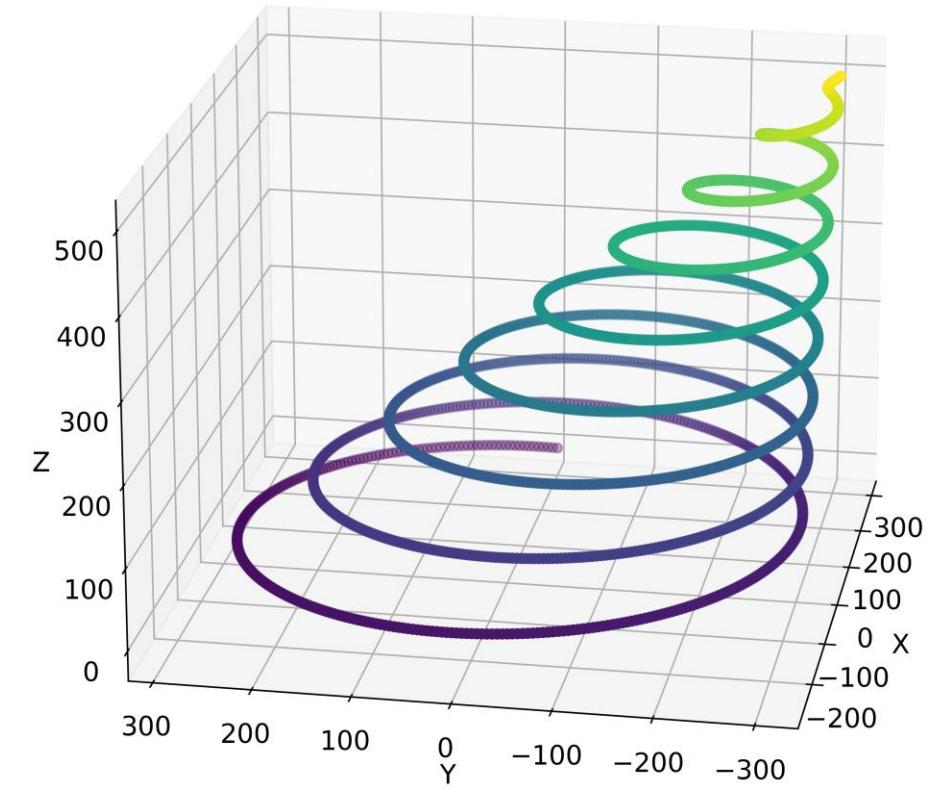
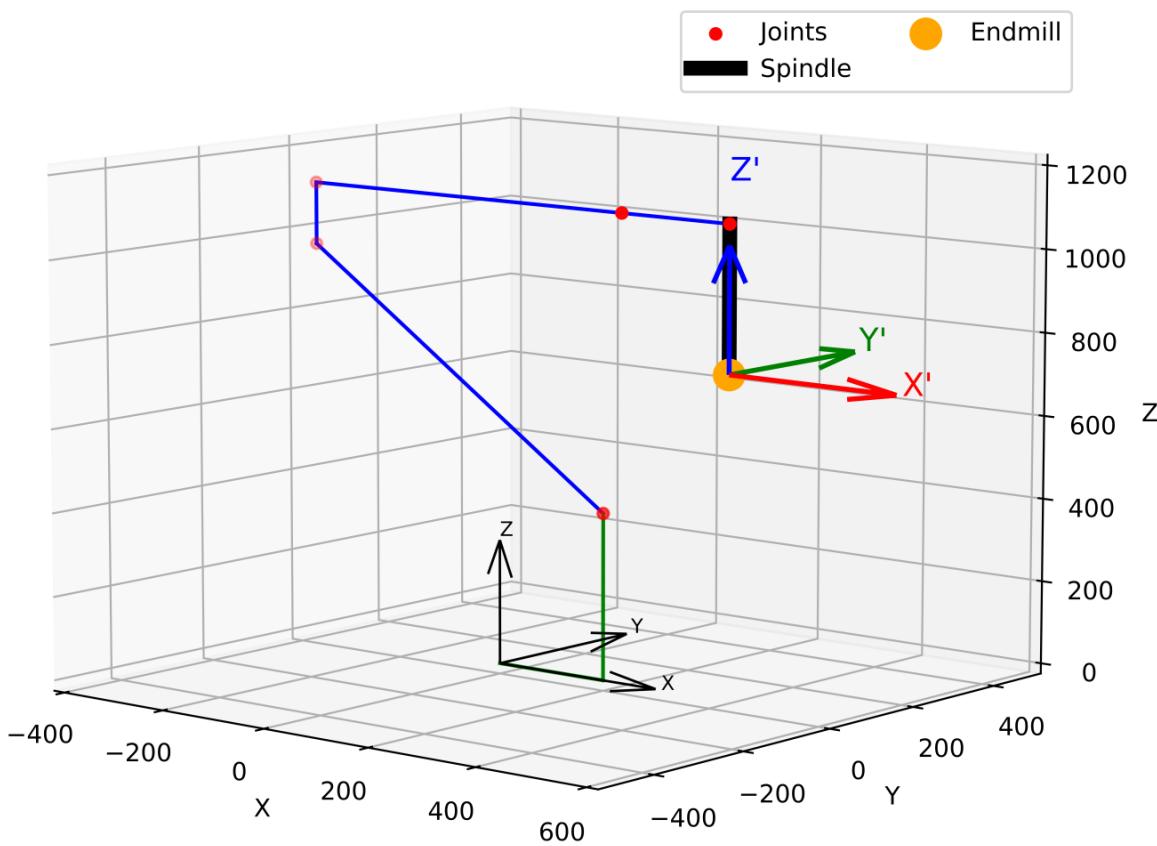
# Boundary Condition Optimization

## Methodology



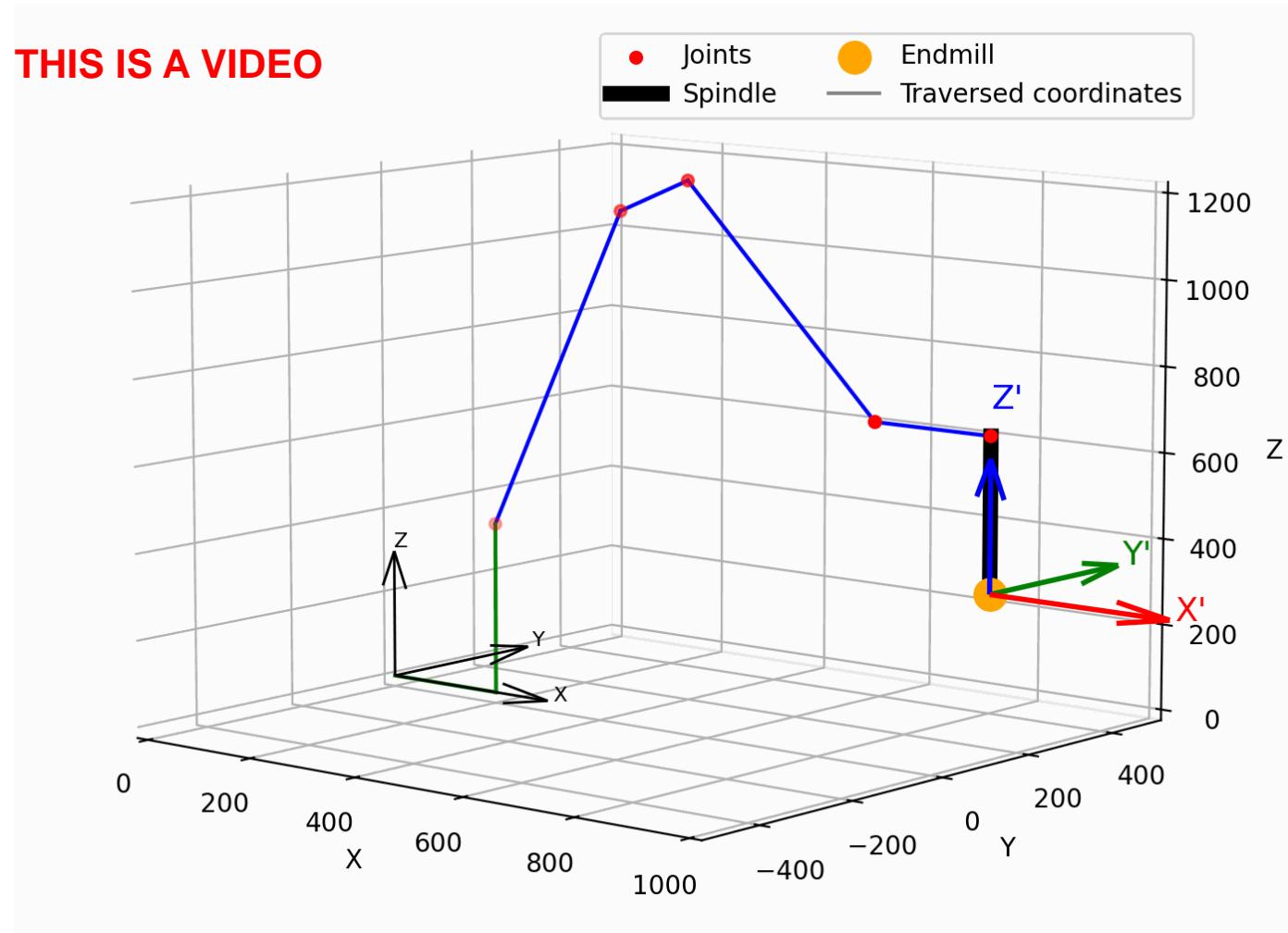
# Modeled Robot and Analyzed Toolpath

## Implementation and Results



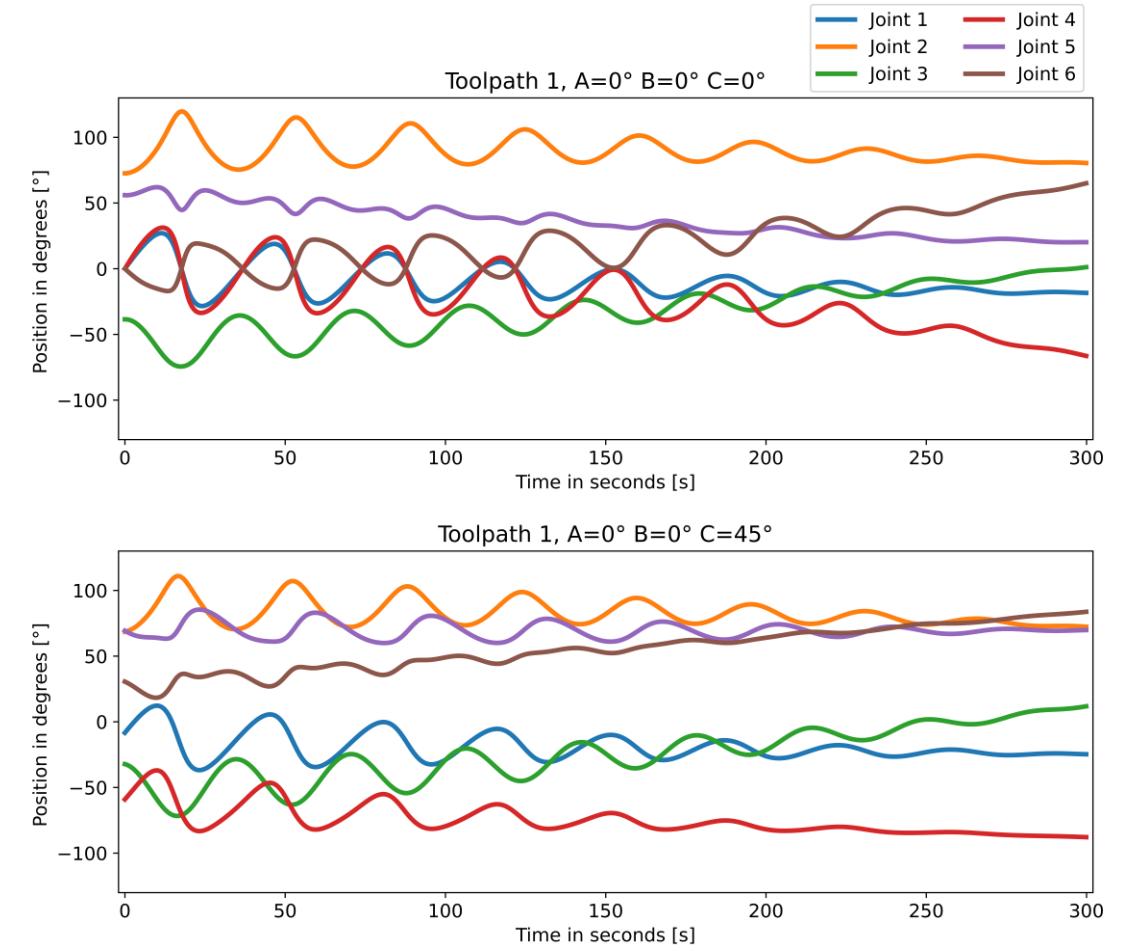
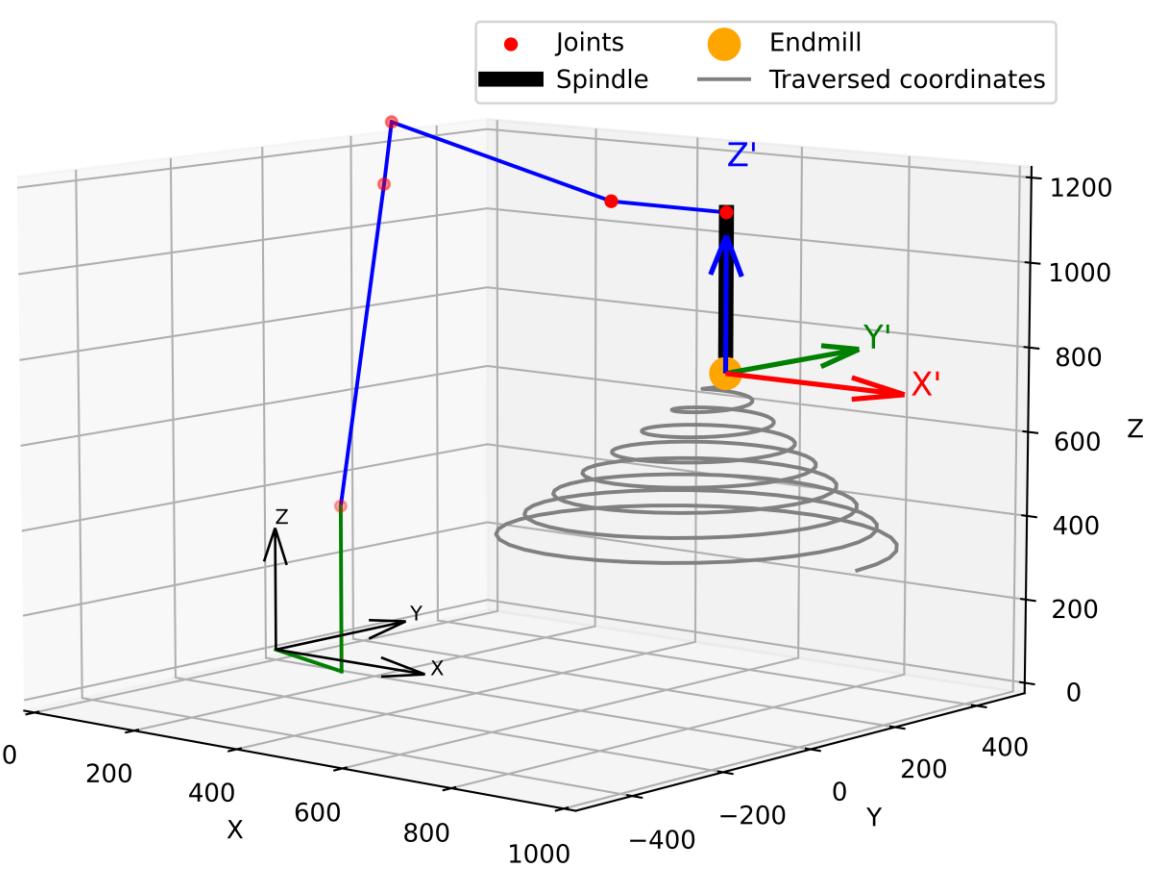
# Traversing the Toolpath

## Implementation and Results

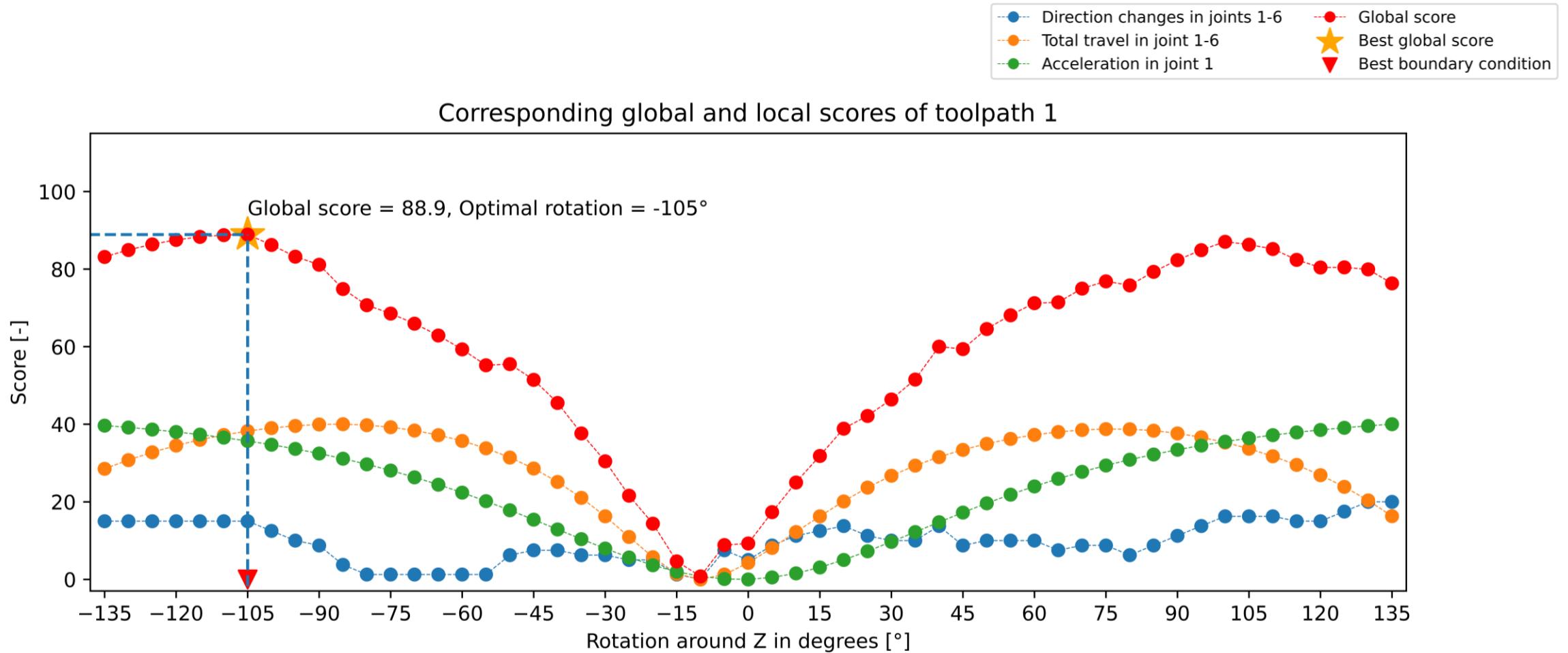


# Extracting Joint Positions

## Implementation and Results



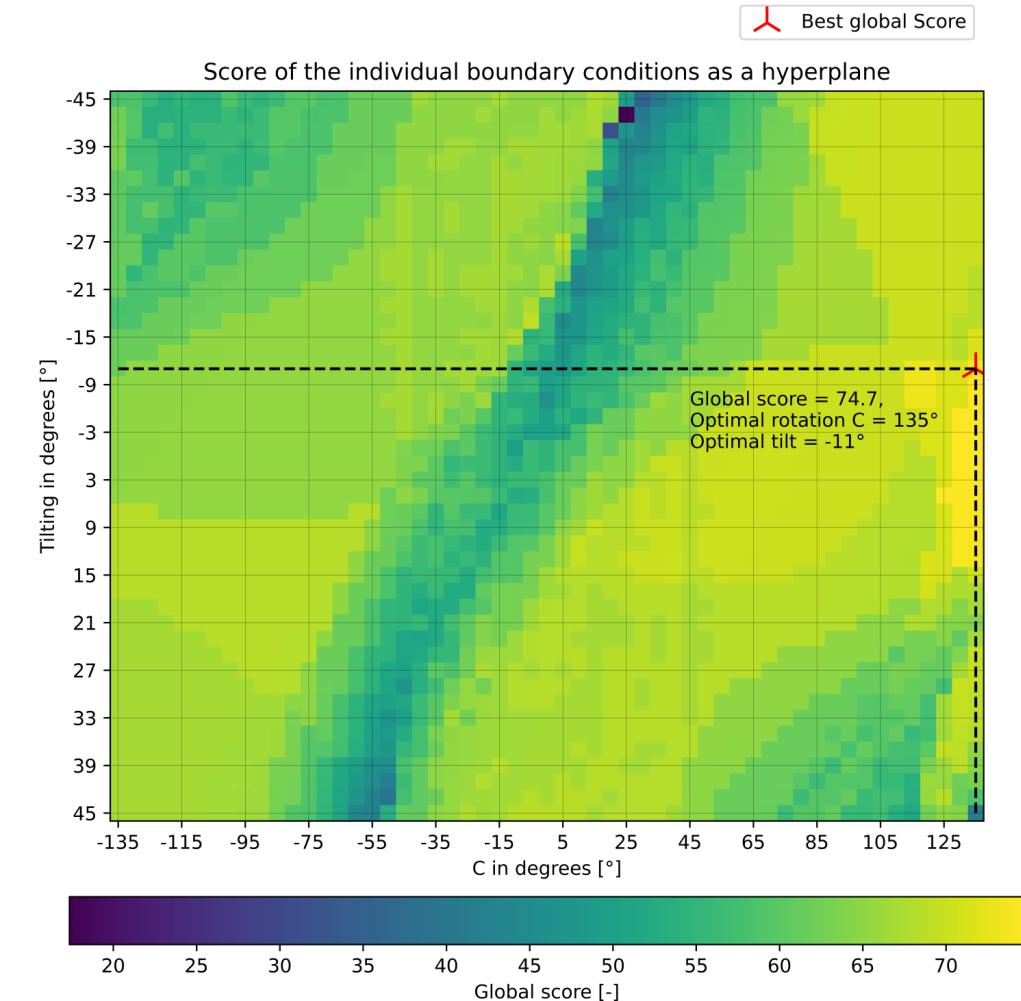
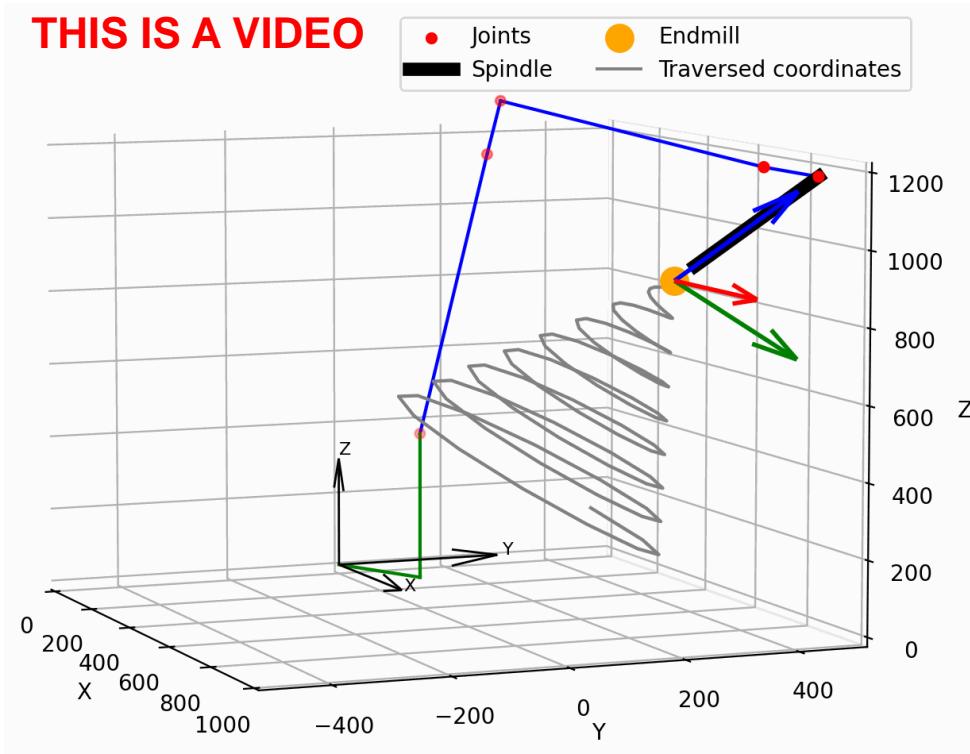
## Implementation and Results



# Addition of a Second Redundant DoF in Form of a Rotary-Tilt-Table

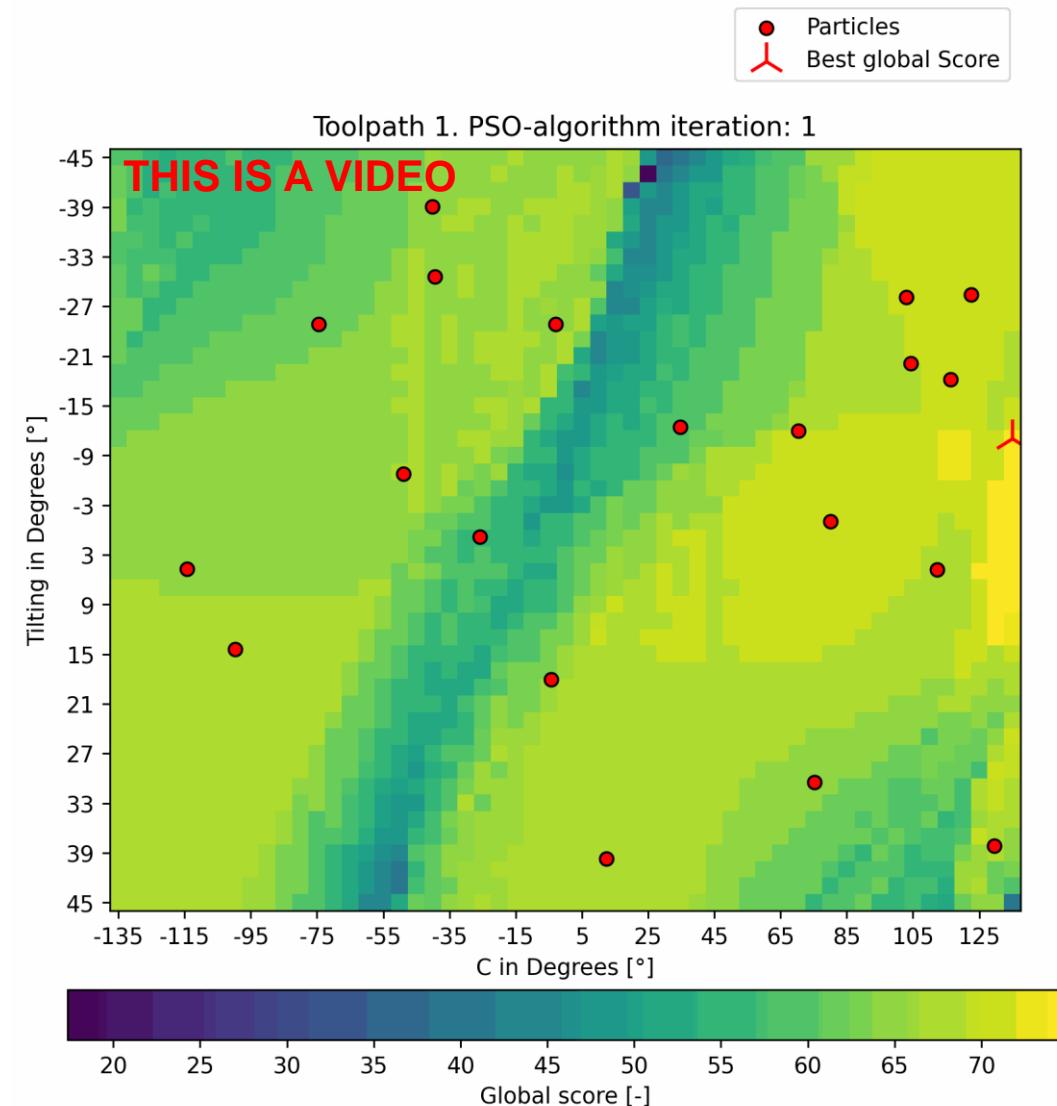
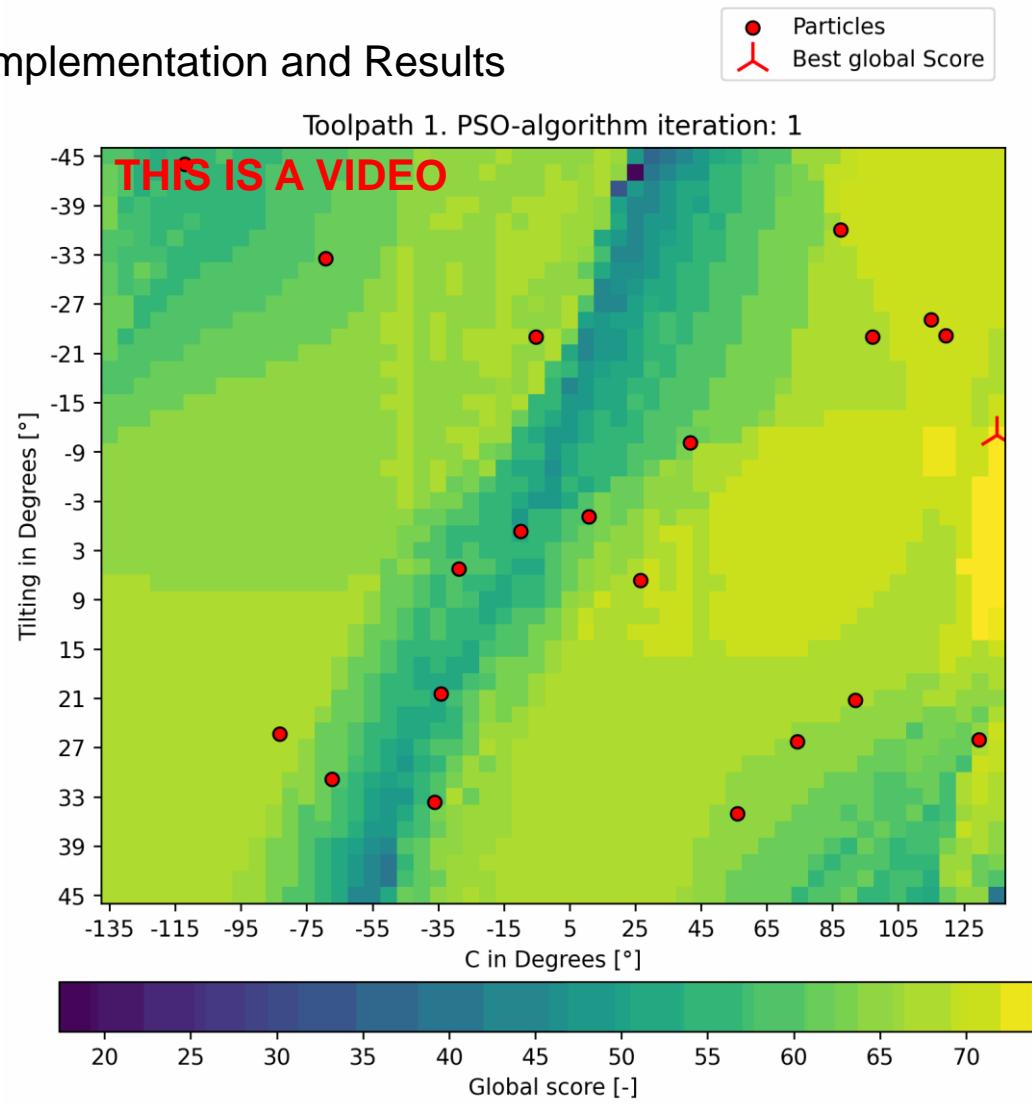
## Implementation and Results

- Two redundant DoFs: 1) Rotation C (tool symmetry axis)  
2) Tilting of the base-plate



# Visualization of the PSO-Algorithm

## Implementation and Results



## Summary and Outlook

### Summary

- The validation provides a solid proof-of-concept
- Redundant DoFs can offer significant potential for improvement in the robot's movement.
- A PSO-algorithm is reasonable choice for finding the optimal boundary conditions
- The method's adaptability allows for wide application to a broad spectrum of robotic systems
- For a more detailed validation, longer production G-codes and multi-axis operations with more than two redundant DoFs are necessary

### Outlook

- A combination with CAM-software can significantly reduce computation time
- Implementation of stiffness analysis to reduce chatter
- Piecewise optimization of the toolpath instead of whole toolpath

THX



## Jan Nalivaika

Student at Siemens



+49 163 7180148



nalivaika@outlook.de



## Ludwig Siebert

Supervisor at TUM



+49 89 289 15578



ludwig.siebert@iwb.tum.de



## Marius Breuer

Supervisor at Siemens



+49 172 8396287

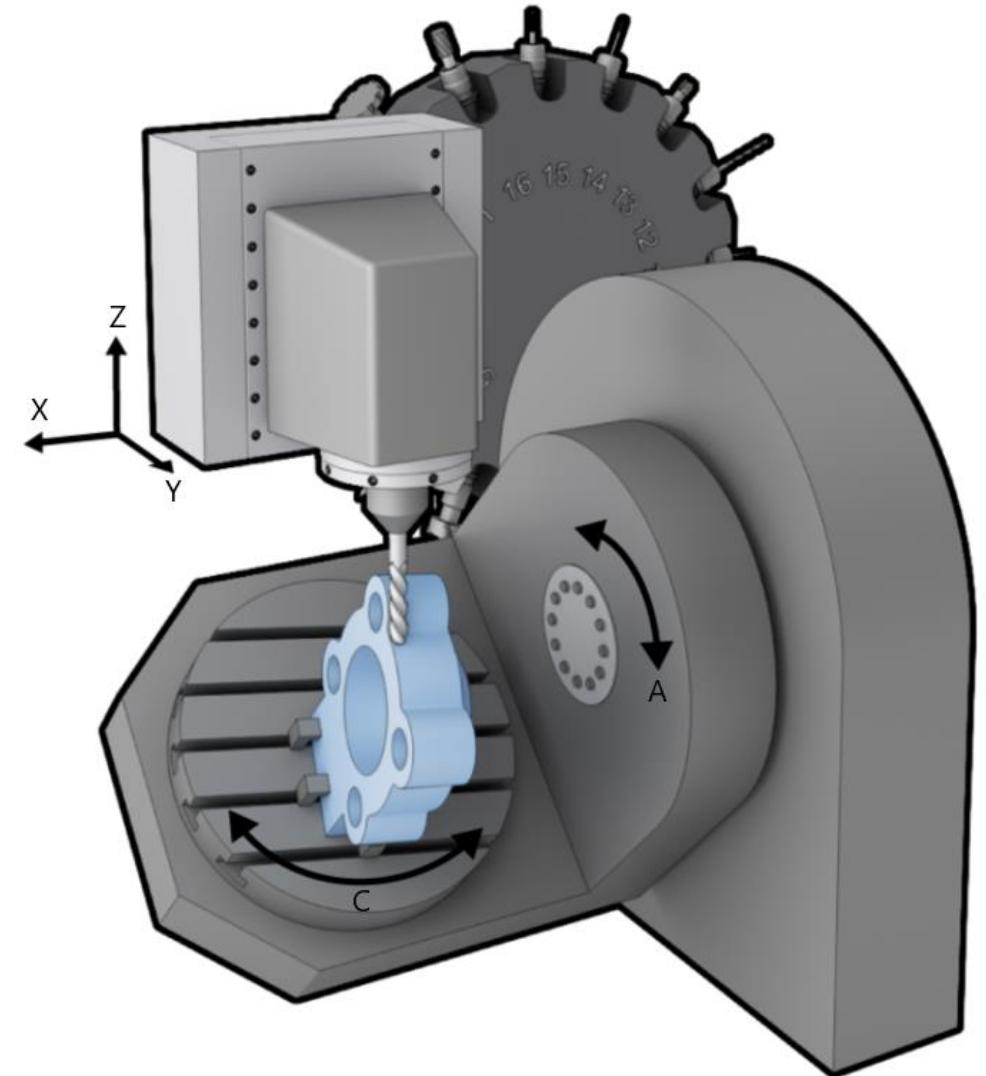
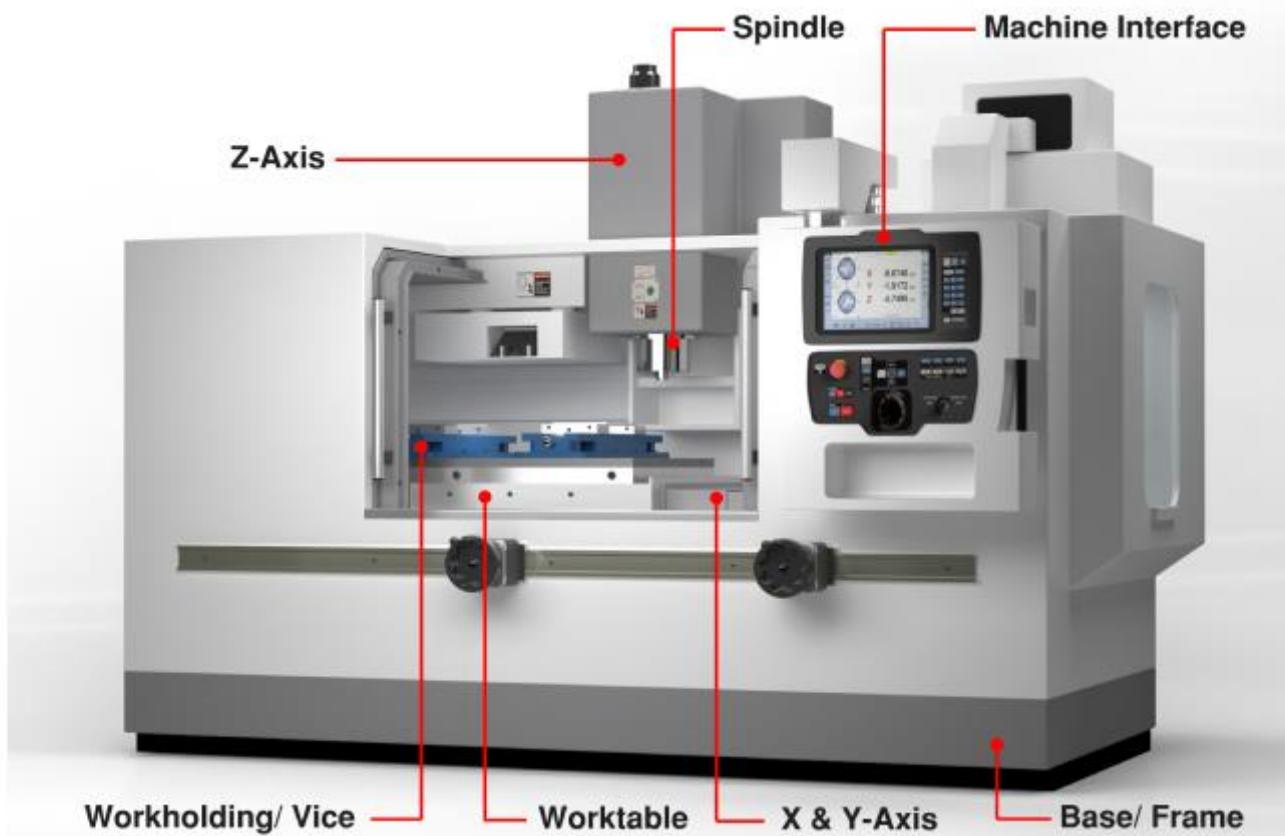


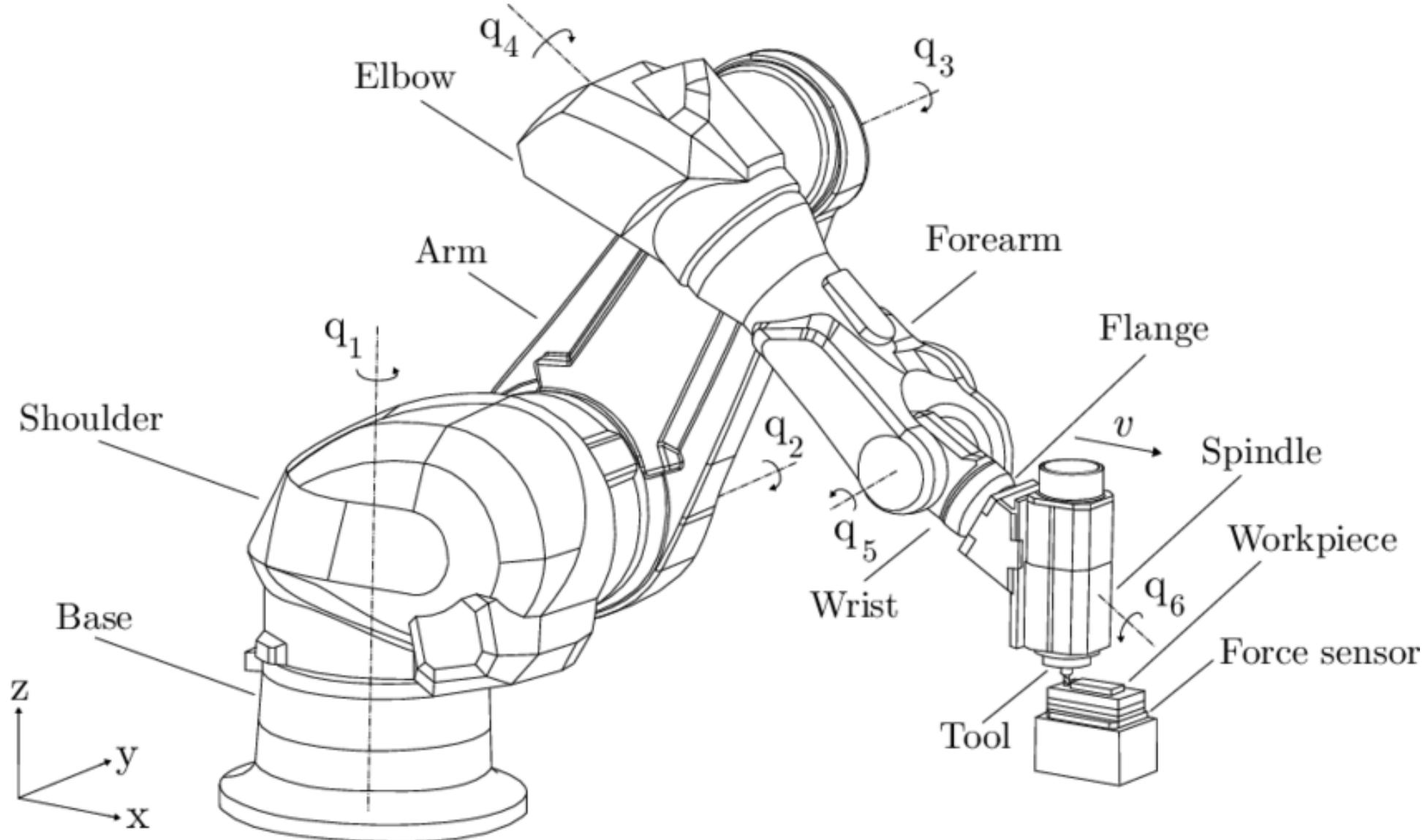
marius.breuer@siemens.com

# Sources

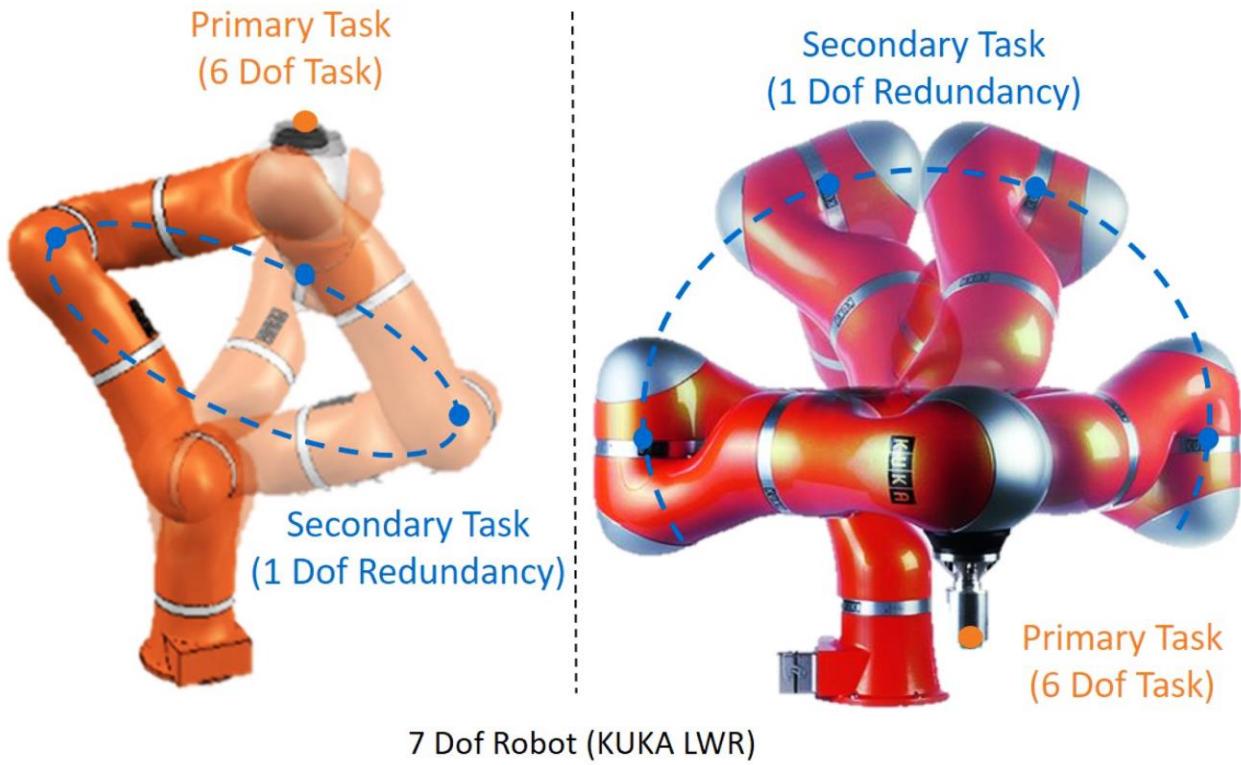
- Ni and Balyan (2021) Research on Mobile User Interface for Robot Arm Remote Control in Industrial Application
- Jung and Lim (2020) Industrial robots, employment growth, and labor cost: A simultaneous equation analysis
- Stevenson et al., (2002) Singularity avoidance and aspect maintenance in redundant manipulators
- Bedrossian,(1990) Classification of singular configurations for redundant manipulators
- Huo and Baron, (2008) The joint-limits and singularity avoidance in robotic welding
- Milenkovic, (2021) Wrist singularity avoidance with a robot end-effector adding an oblique, redundant axis
- Shi et al., (2021) Kinematics and Singularity Analysis of a 7-DOF Redundant Manipulator
- Dai et al., (2020) Planning Jerk-Optimized Trajectory With Discrete Time Constraints for Redundant Robots
- Jiang et al., (2017) Time-Jerk Optimal Trajectory Planning for a 7-DOF Redundant Robot Using the Sequential Quadratic Programming Method
- Bien Duong (2021) On the Effect of the End-effector Point Trajectory on the Joint Jerk of the Redundant Manipulators
- Wu et al., (2021) Review of Industrial Robot Stiffness Identification and Modelling
- Xiong et al., (2019) Stiffness-based Pose Optimization of an Industrial Robot for Five-axis Milling
- Wang et al., (2022) Optimization of redundant degree of freedom in robotic milling considering chatter stability
- Cvitanic et al., (2020) Pose optimization in robotic machining using static and dynamic stiffness models
- Uhlmann et al., (2016) Energy Efficient Usage of Industrial Robots for Machining Processes
- Boscariol et al., (2020) Energy Optimization of Functionally Redundant Robots through Motion Design
- Soori et al., (2015) Optimization of energy consumption in industrial robots, a review

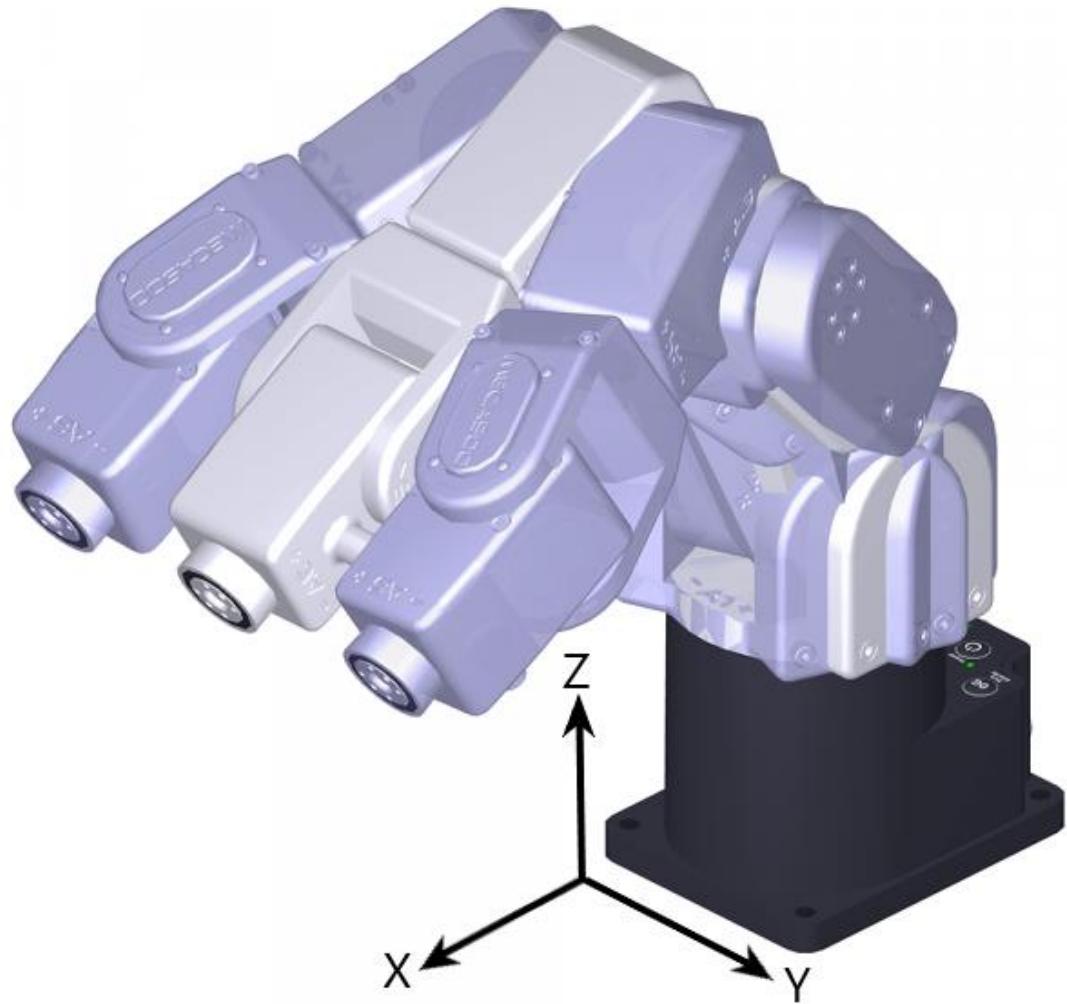
# CNC machines



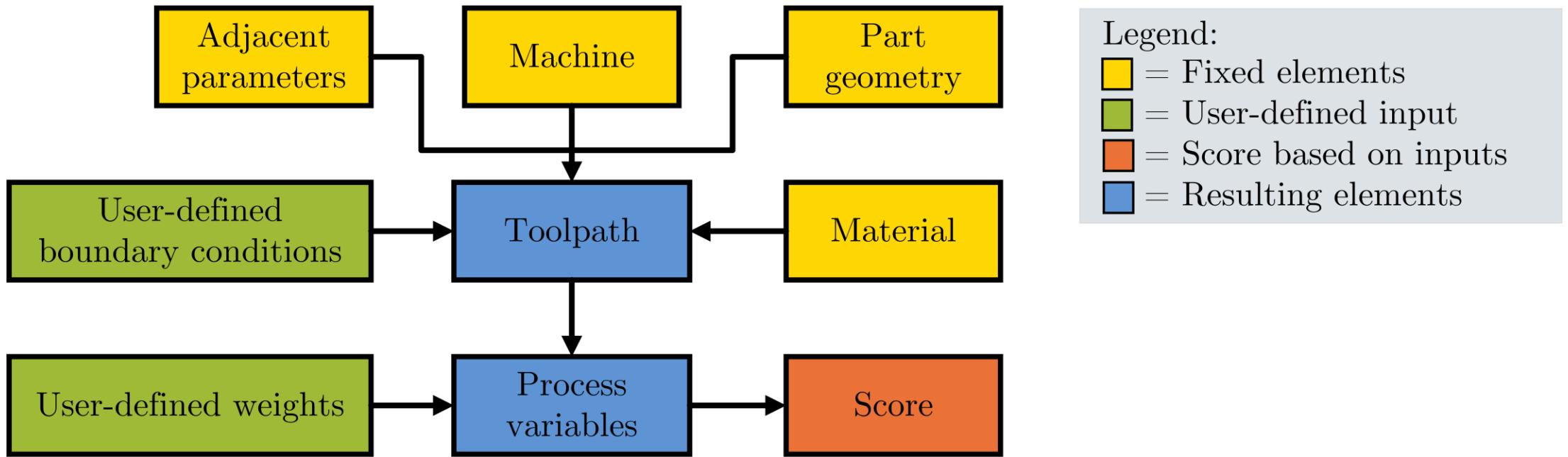


# Manufacturing Systems with Redundant DoFs

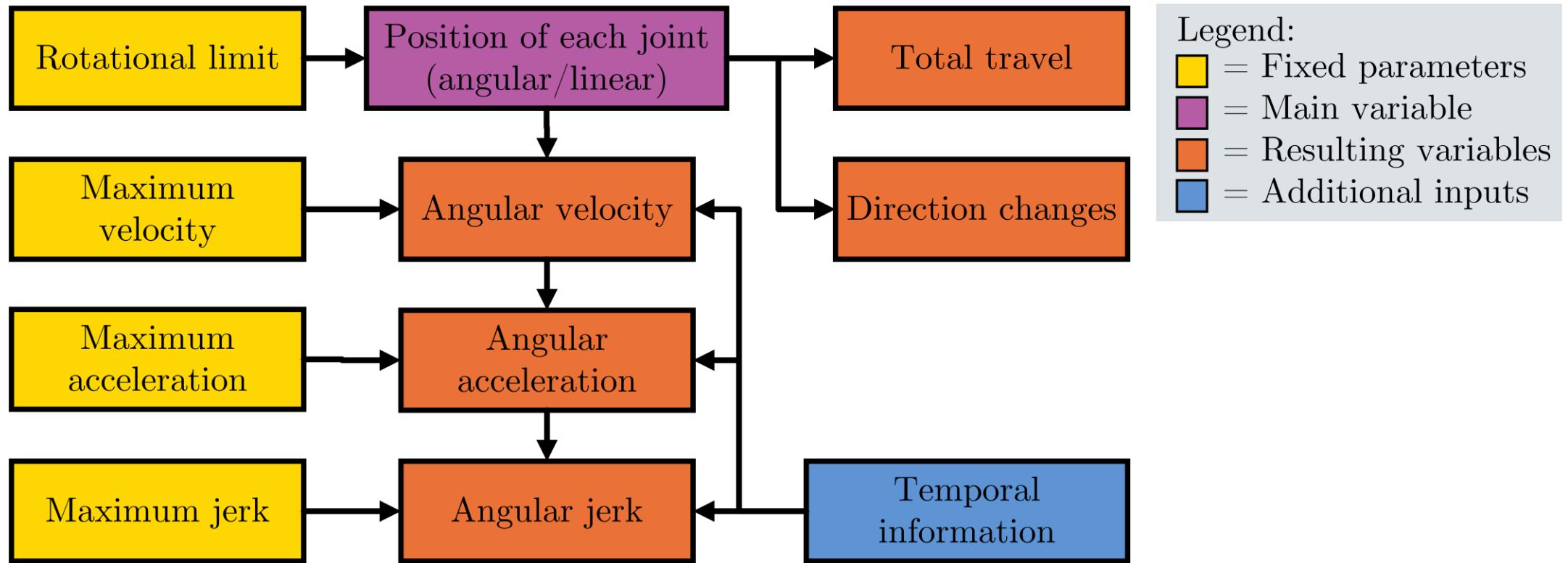




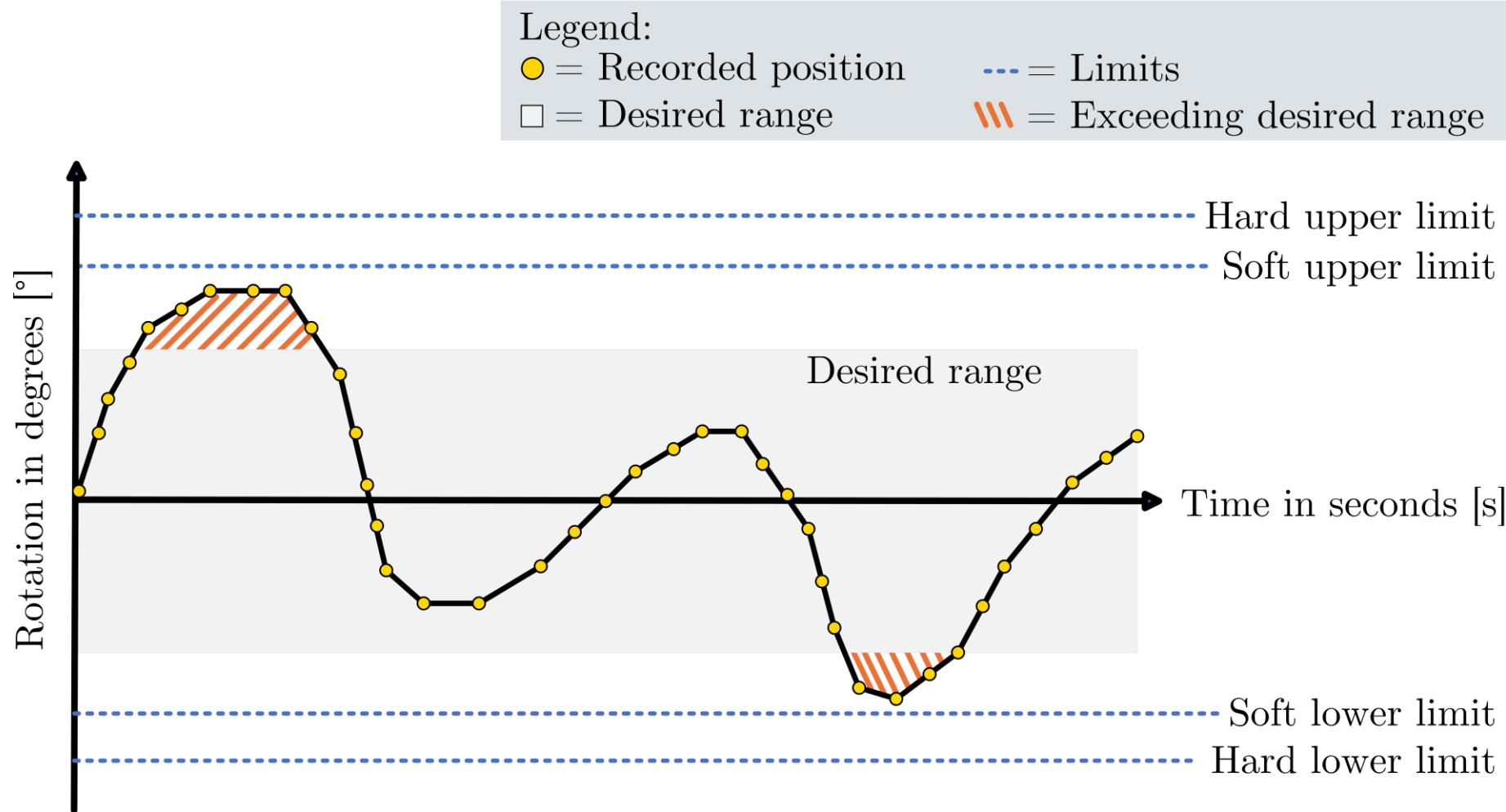
# Interdependence of various parameters and elements



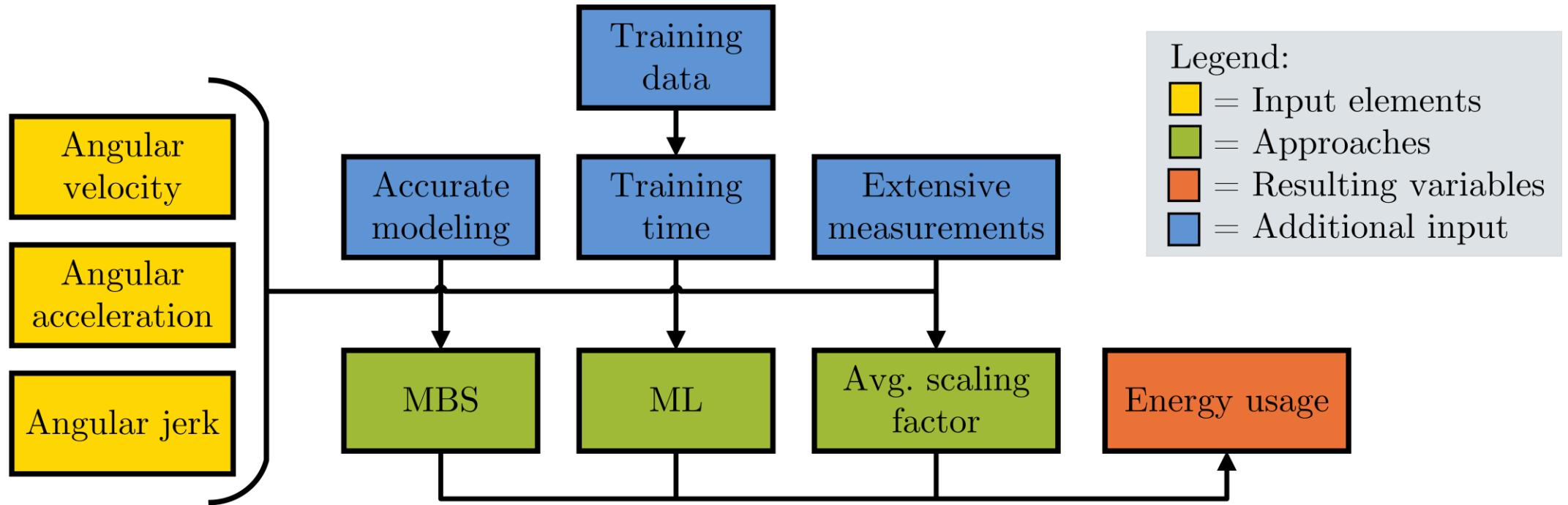
# Additional information for angular position of each joint



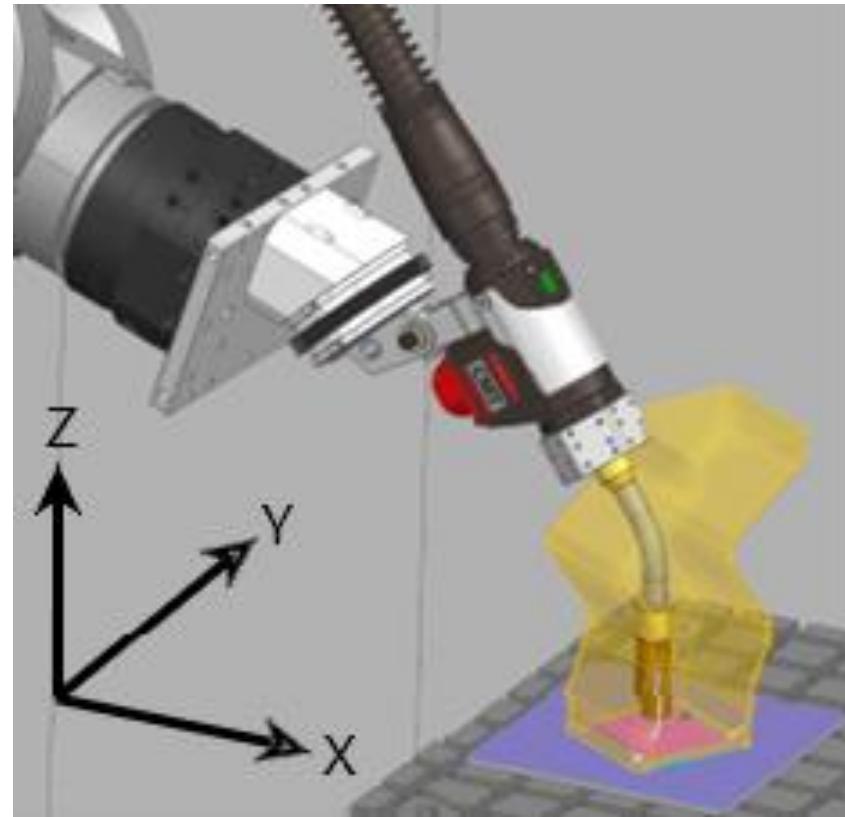
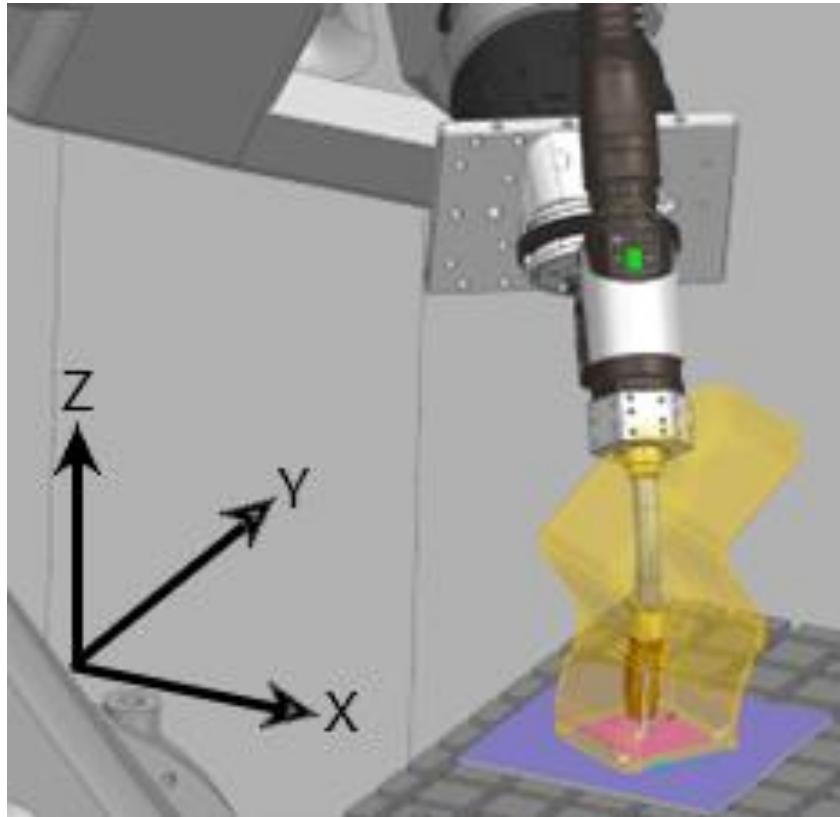
# Hard and soft limits with desired range



# Exemplary methods for energy usage calculations

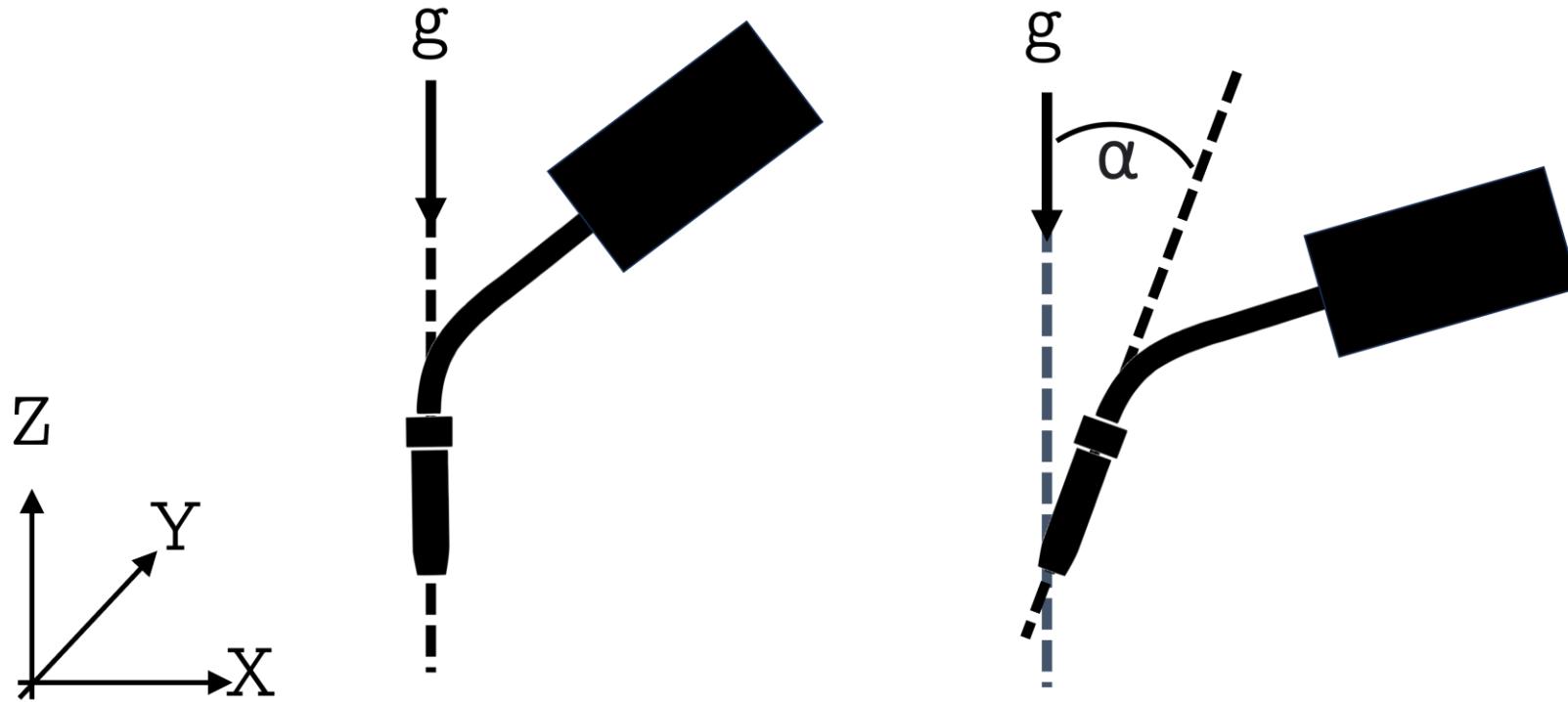


# Rotation around the C-Axis of a welding torch



# Torch Orientation

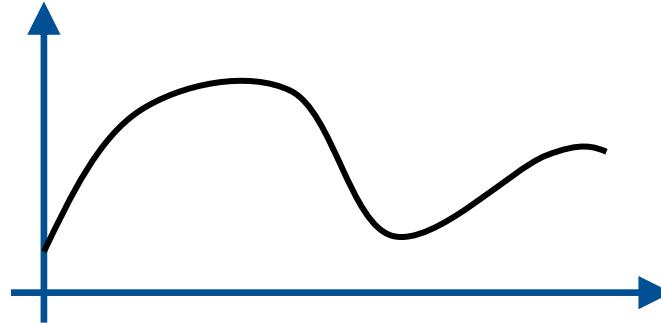
Example of optimal and non-optimal tilt in the welding torch



# Transforming Time-Series Data into Scalar Values

Time-series data:

- Velocity
- Acceleration
- Jerk
- ...

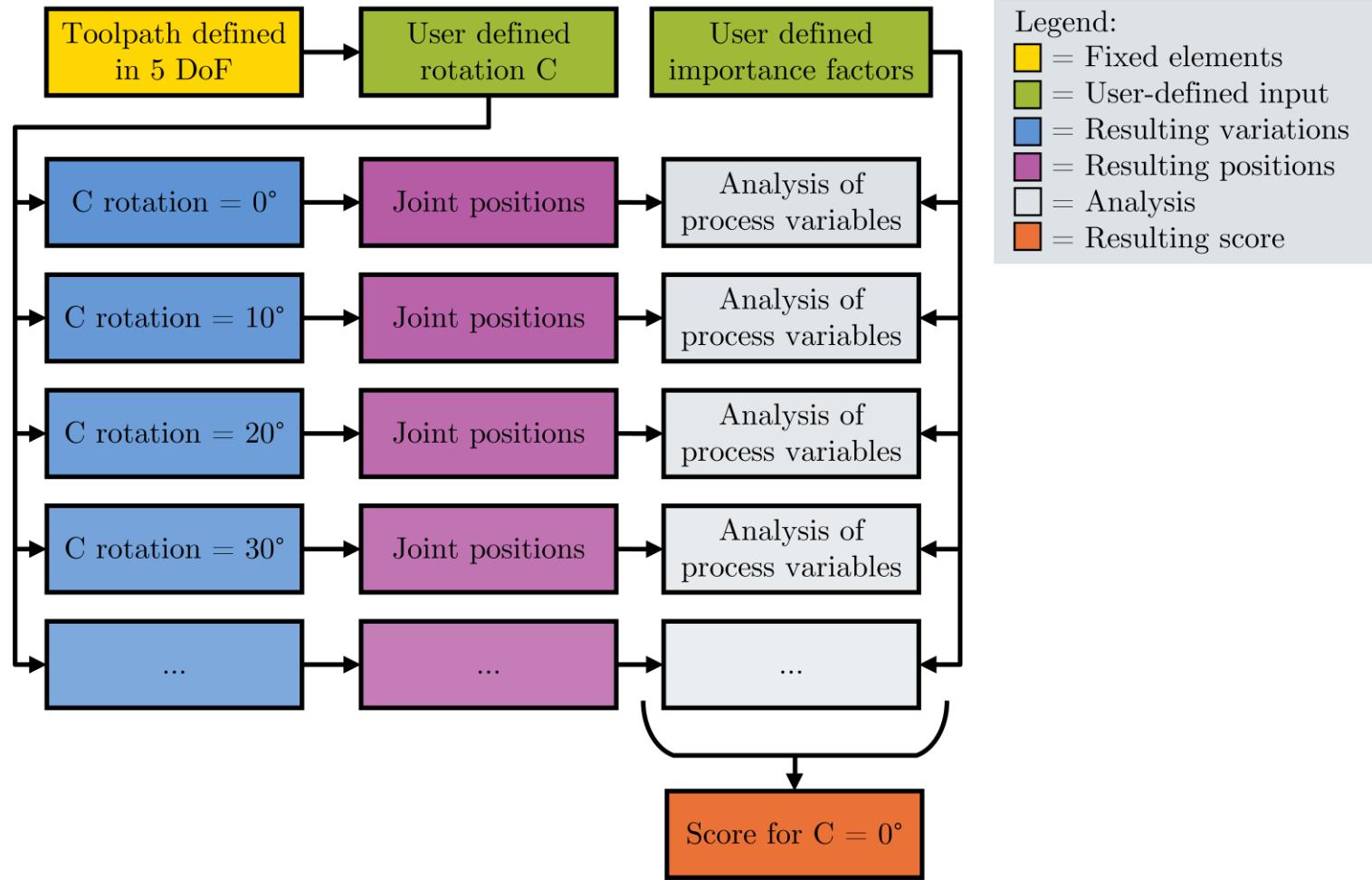


Optional procedures:

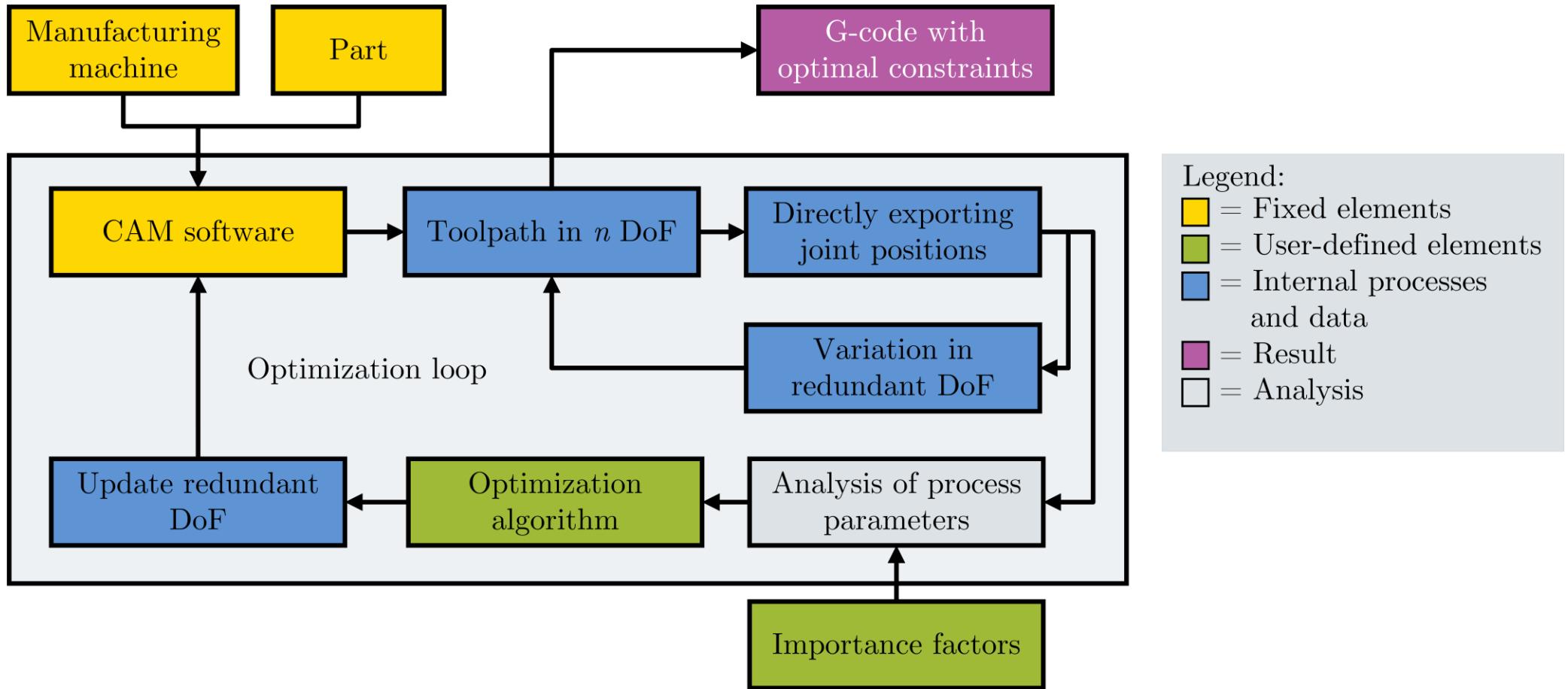
- Squaring and aggregation
- Cubing and aggregation
- Definition of thresholds
- Machine learning

Process variables	Local rating	Importance factors	Local score
Velocity in joints 1-6	45	0.1	4.5
Accelerations in joint 2	90	0.8	72
Accelerations in joint 1 and 3-6	15	0.1	1.5
Jerk in joints 1-6	4	0	0
Global Score			78

# General Process Analysis

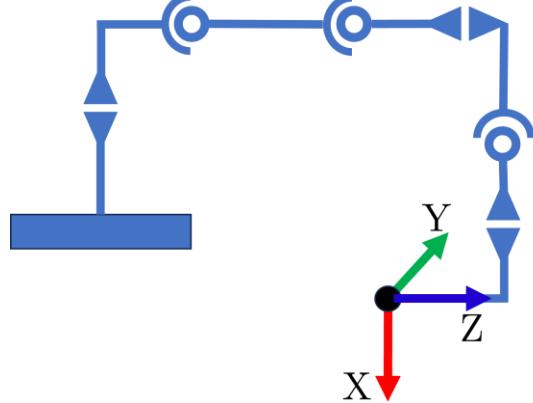


# Optimization Loop With CAM Software in the Loop

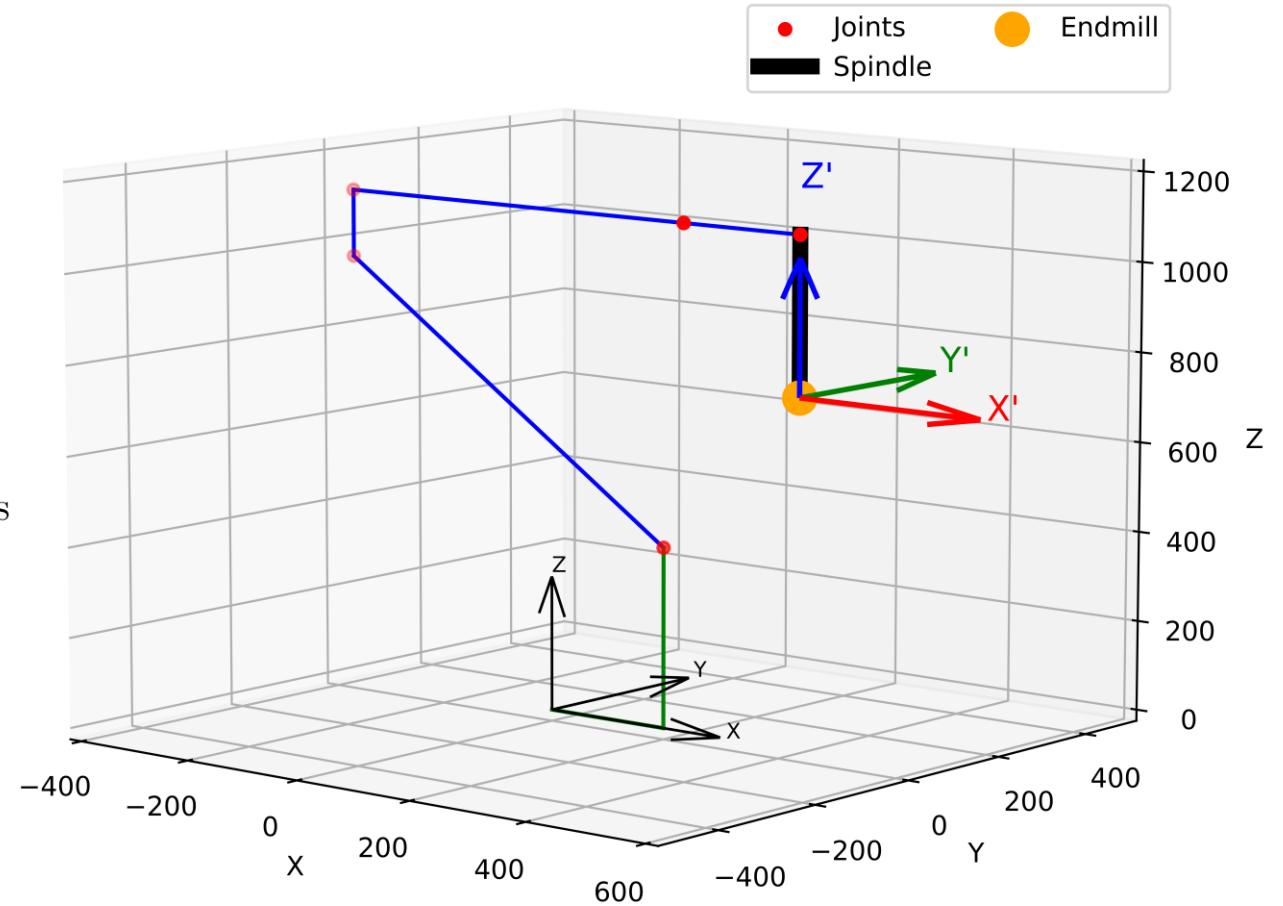


# Modeled Robot

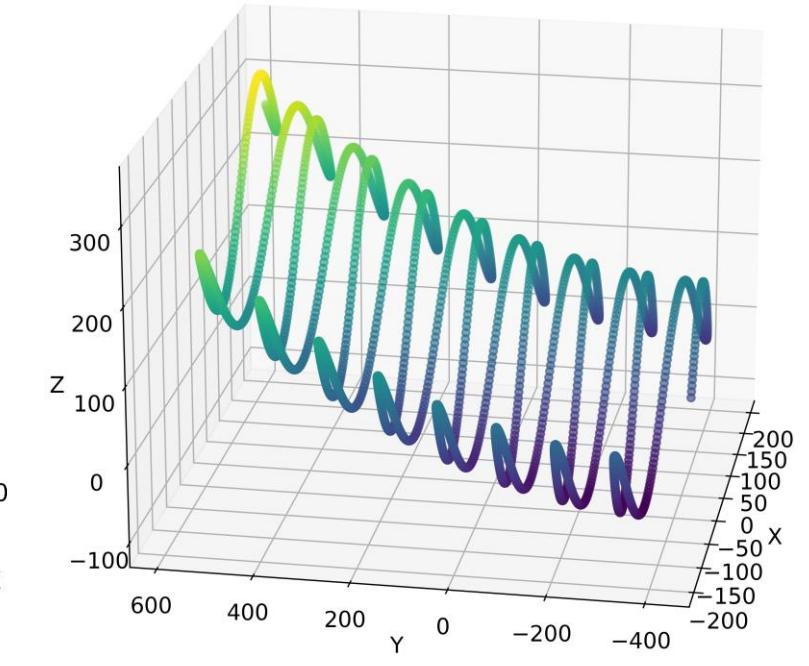
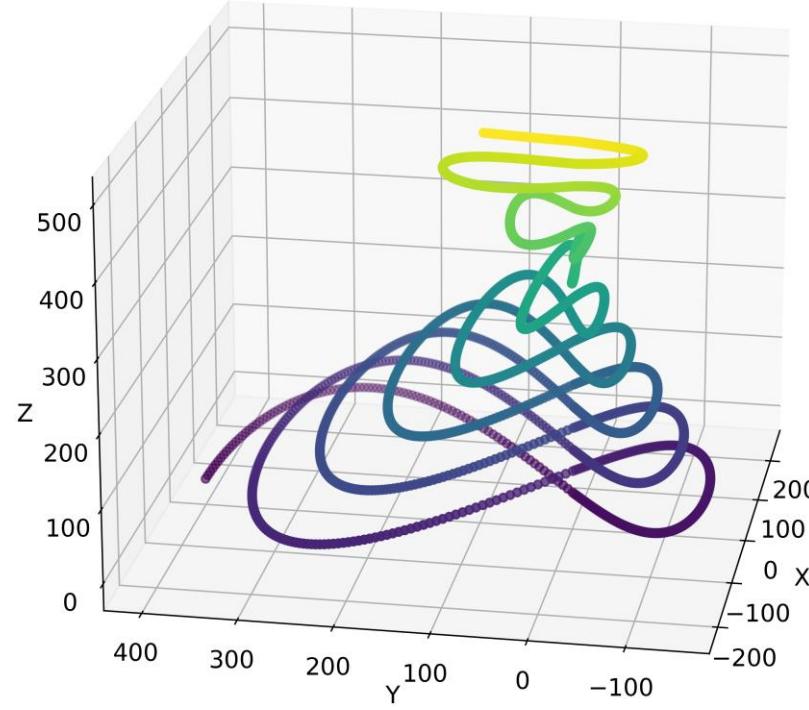
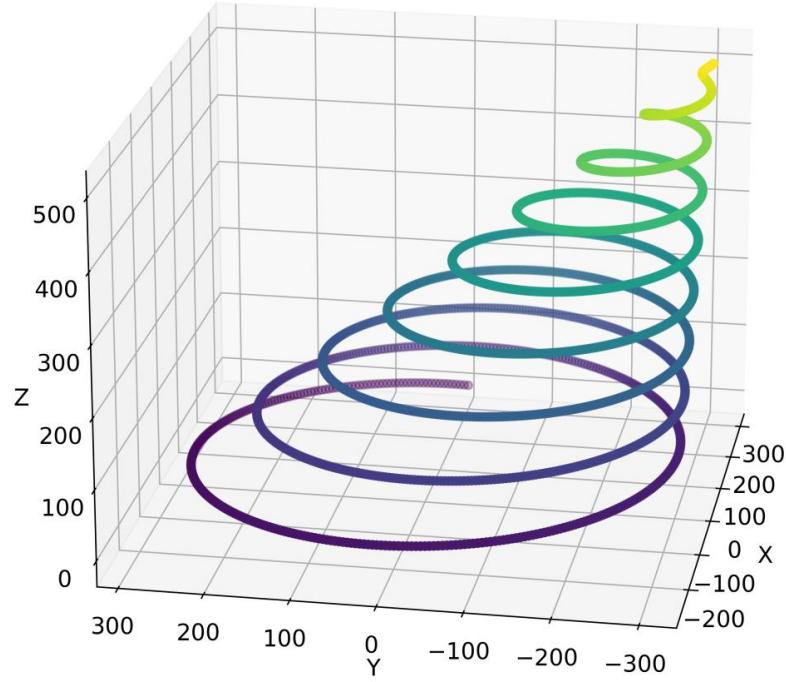
Parameters	Values
a in [mm]	[200, 800, 150, 0, 0, 0]
alpha in [°]	[90, 0, 90, -90, 90, 0]
d in [mm]	[400 0, 0, 600, 0, 200]



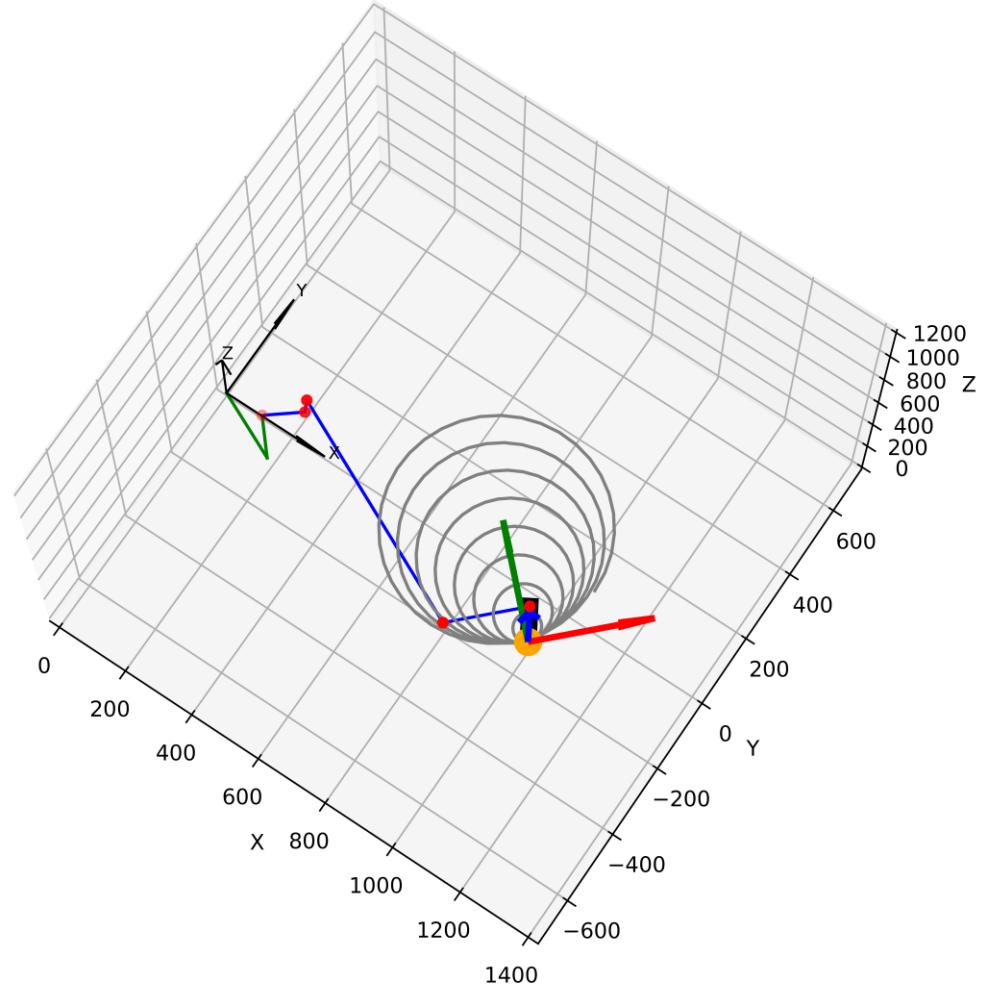
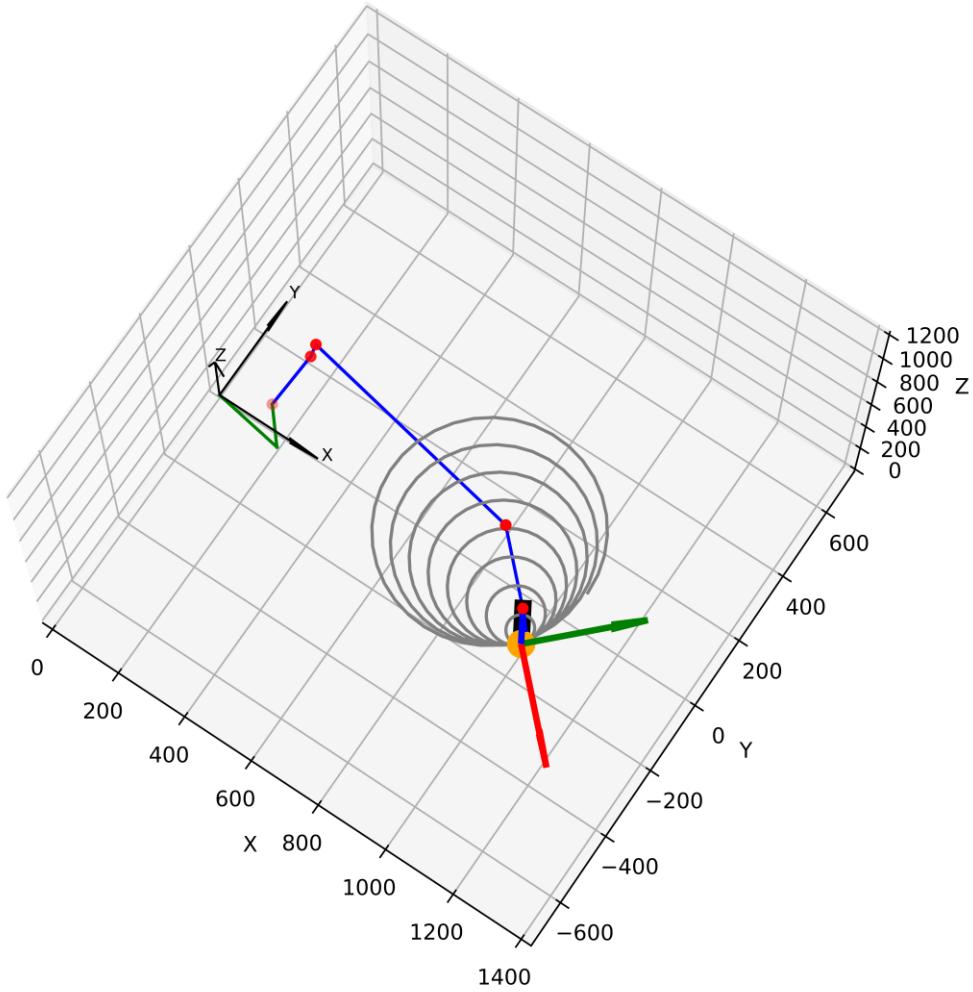
 = Rotating joints  
 = Tilting joints



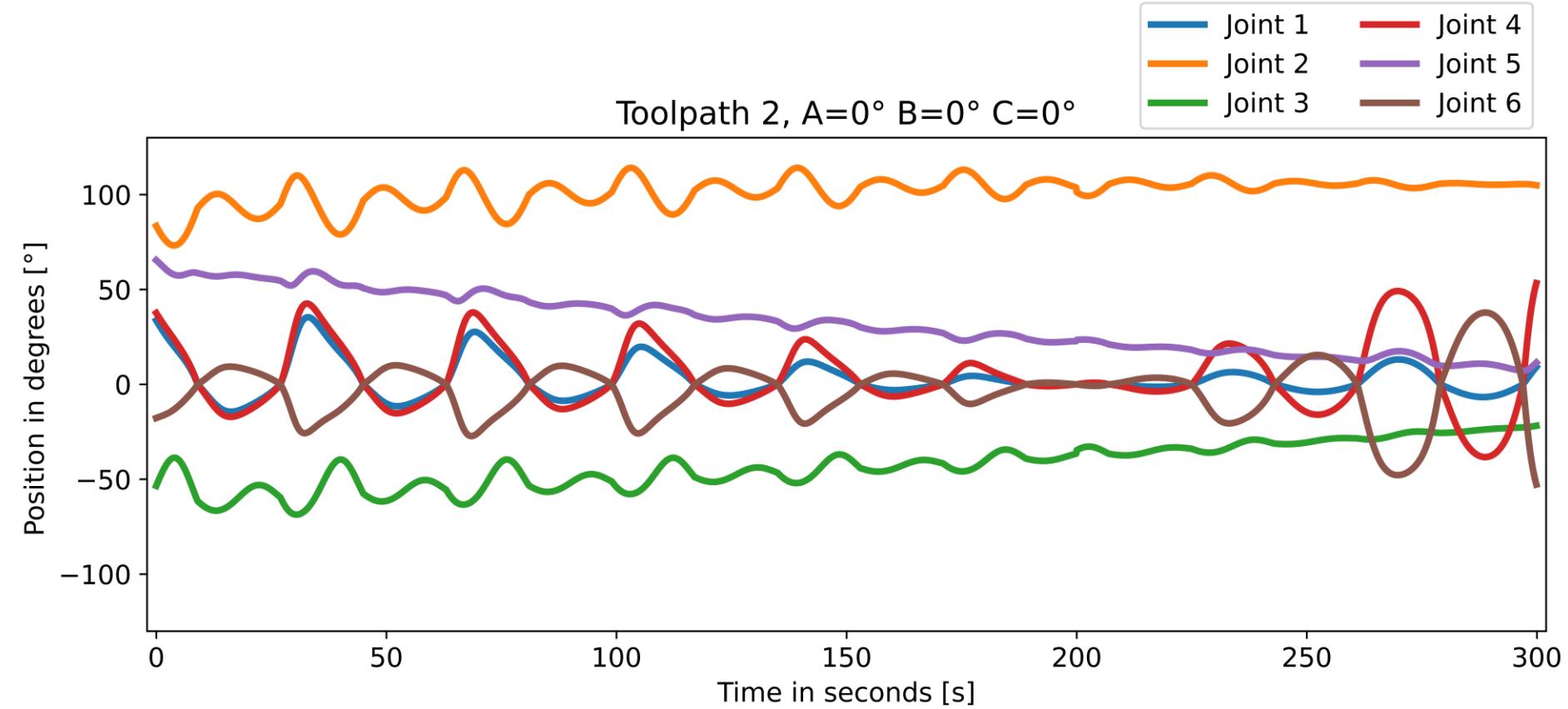
# Modeling a Basic Toolpath



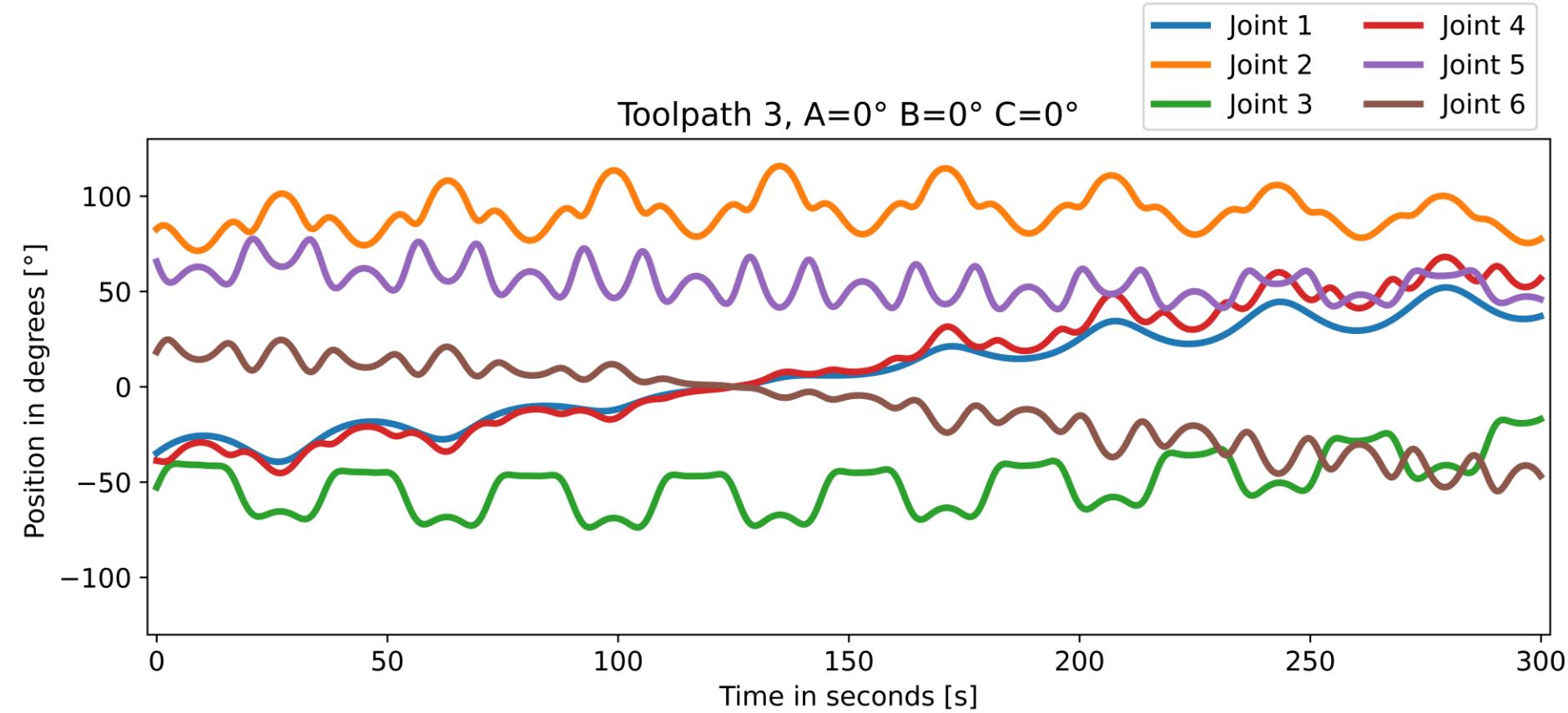
# Rotation around the Z-axis for toolpath 1



# Toolpath 2



# Toolpath 3



# G-code Variation

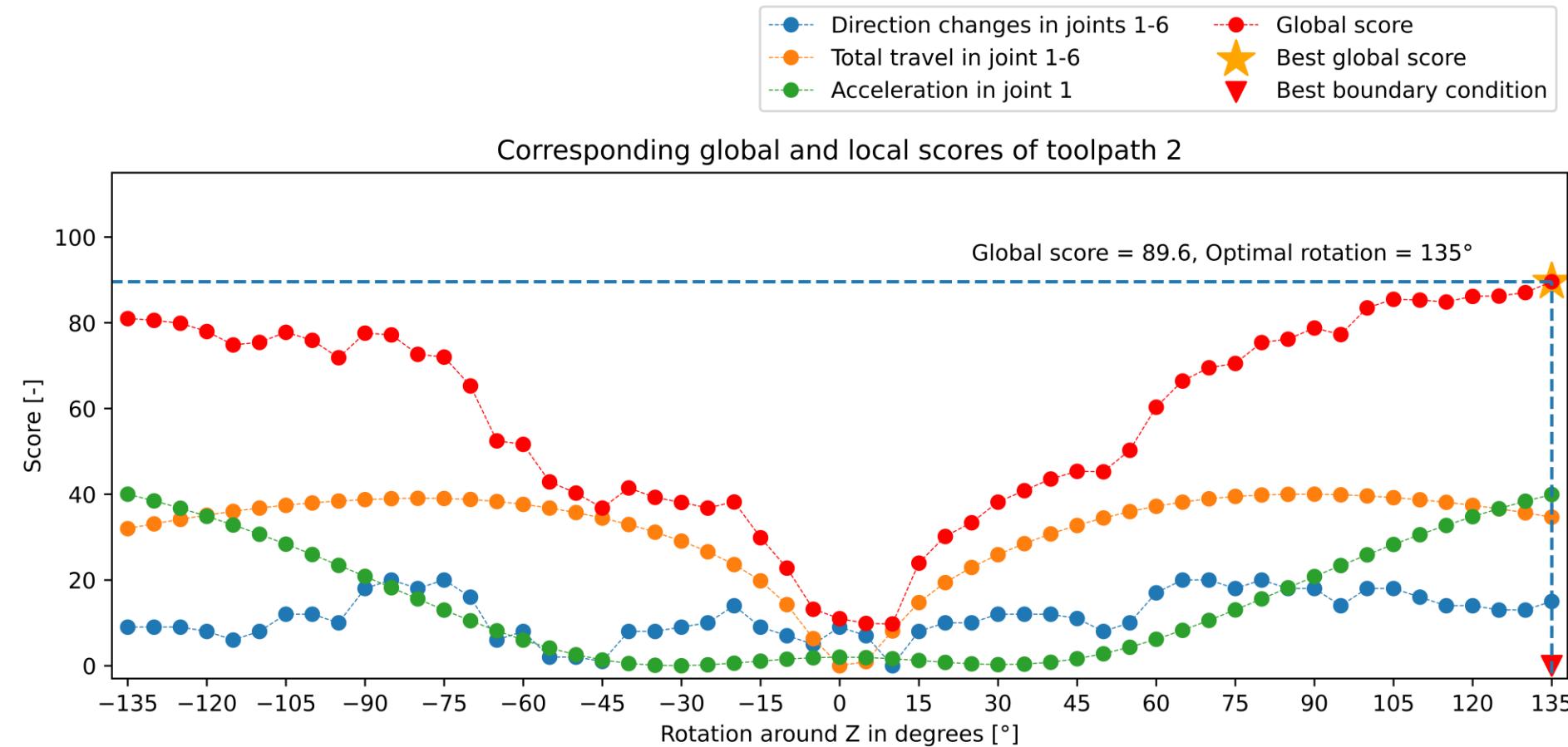
```
N10 G1 X=-19.988 Y=51.04 Z=56. A=0.0 B=0.0 C=0.0  
N20 G1 X=-19.988 Y=49.221 Z=56. A=0.0 B=0.0 C=0.0  
N30 G1 X=-19.988 Y=46.19 Z=56. A=0.0 B=0.0 C=0.0  
N40 G1 X=-19.988 Y=44.371 Z=56. A=0.0 B=0.0 C=0.0  
N50 G1 X=-19.988 Y=41.34 Z=56. A=0.0 B=0.0 C=0.0  
N60 G1 X=-19.988 Y=39.521 Z=56. A=0.0 B=0.0 C=0.0
```

```
N10 G1 X=-19.988 Y=51.04 Z=56. A=0.0 B=0.0 C=5.0  
N20 G1 X=-19.988 Y=49.221 Z=56. A=0.0 B=0.0 C=5.0  
N30 G1 X=-19.988 Y=46.19 Z=56. A=0.0 B=0.0 C=5.0  
N40 G1 X=-19.988 Y=44.371 Z=56. A=0.0 B=0.0 C=5.0  
N50 G1 X=-19.988 Y=41.34 Z=56. A=0.0 B=0.0 C=5.0  
N60 G1 X=-19.988 Y=39.521 Z=56. A=0.0 B=0.0 C=5.0
```

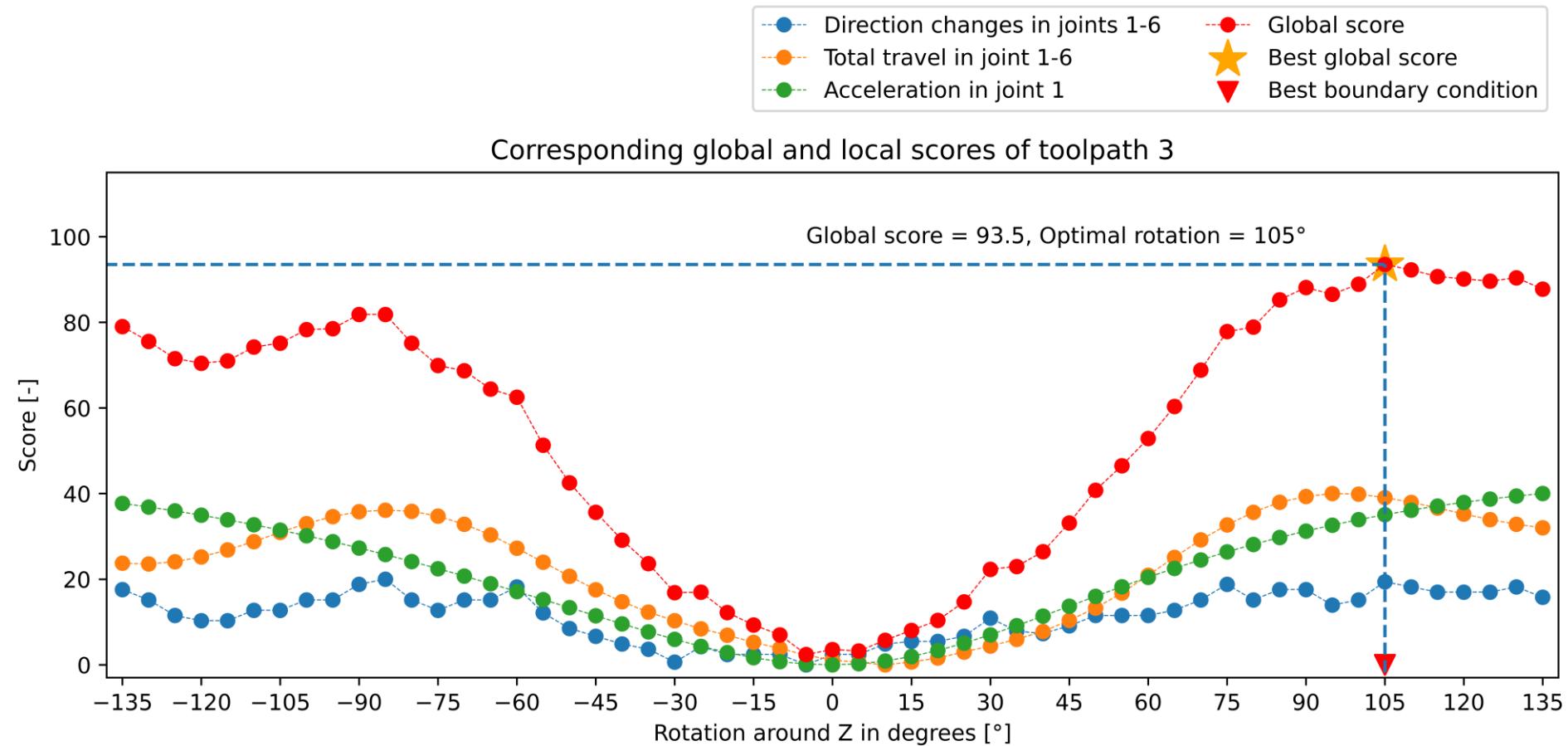
```
N10 G1 X=-19.988 Y=51.04 Z=56. A=0.0 B=0.0 C=10.0  
N20 G1 X=-19.988 Y=49.221 Z=56. A=0.0 B=0.0 C=10.0  
N30 G1 X=-19.988 Y=46.19 Z=56. A=0.0 B=0.0 C=10.0  
N40 G1 X=-19.988 Y=44.371 Z=56. A=0.0 B=0.0 C=10.0  
N50 G1 X=-19.988 Y=41.34 Z=56. A=0.0 B=0.0 C=10.0  
N60 G1 X=-19.988 Y=39.521 Z=56. A=0.0 B=0.0 C=10.0
```

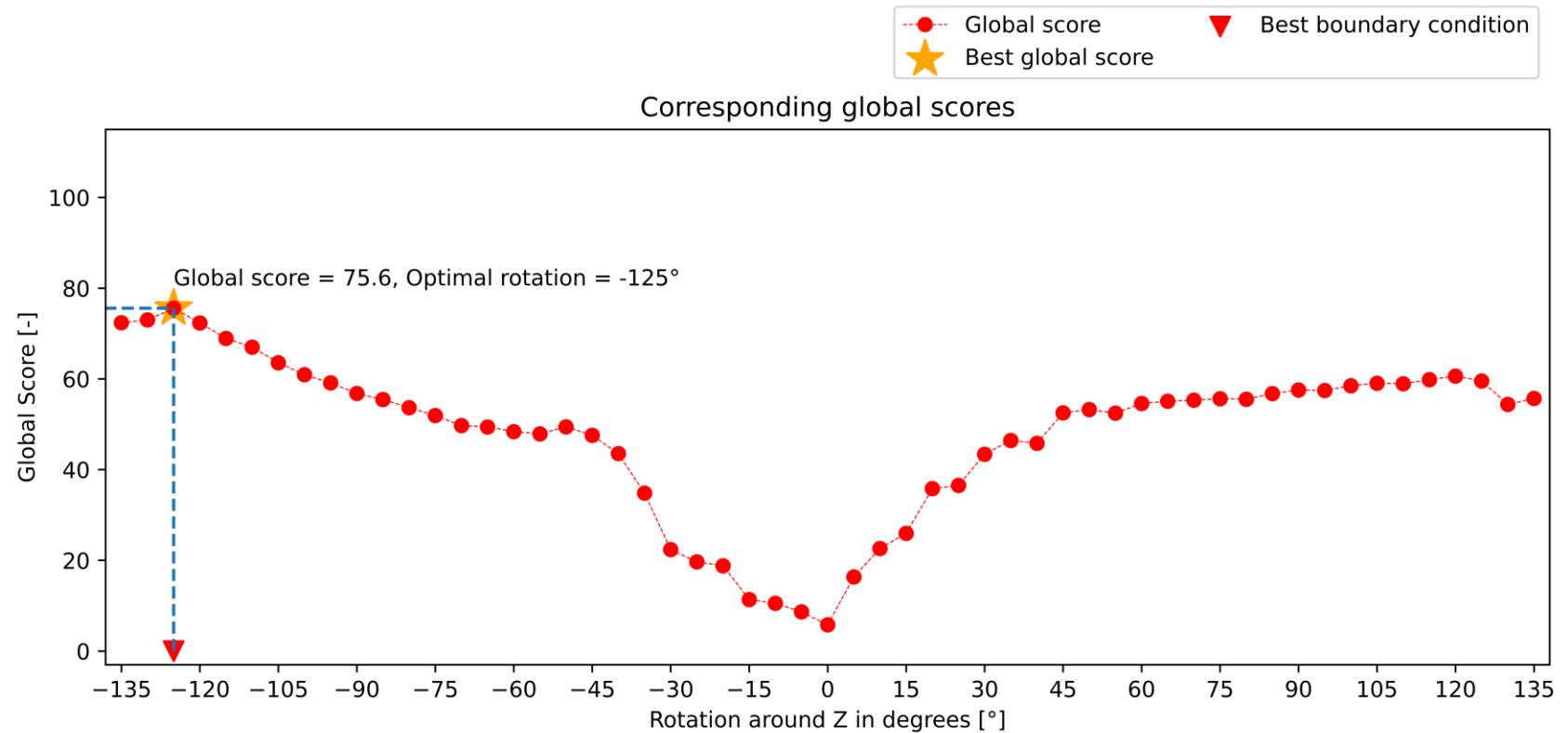
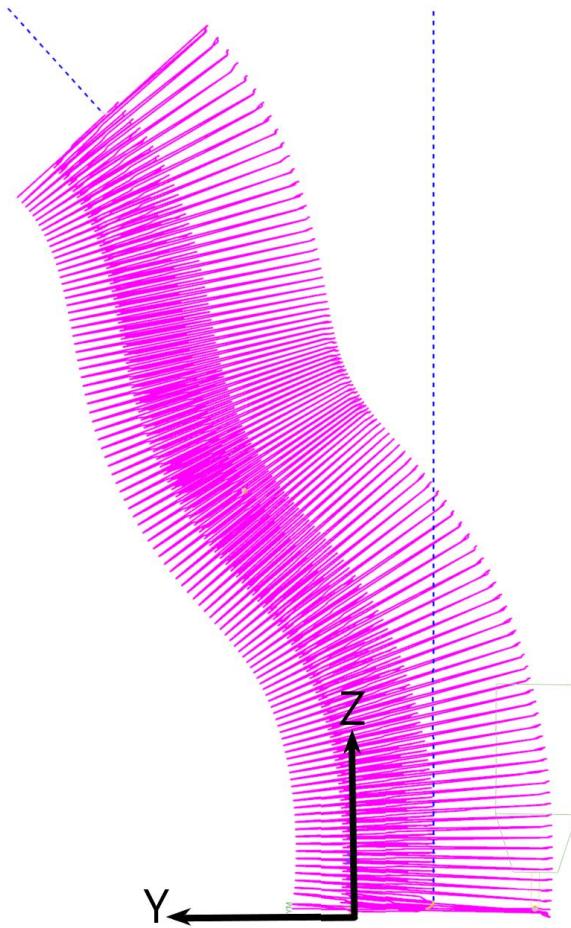
```
N10 G1 X=-19.988 Y=51.04 Z=56. A=0.0 B=0.0 C=15.0  
N20 G1 X=-19.988 Y=49.221 Z=56. A=0.0 B=0.0 C=15.0  
N30 G1 X=-19.988 Y=46.19 Z=56. A=0.0 B=0.0 C=15.0  
N40 G1 X=-19.988 Y=44.371 Z=56. A=0.0 B=0.0 C=15.0  
N50 G1 X=-19.988 Y=41.34 Z=56. A=0.0 B=0.0 C=15.0  
N60 G1 X=-19.988 Y=39.521 Z=56. A=0.0 B=0.0 C=15.0
```

# Toolpath 2 and Score

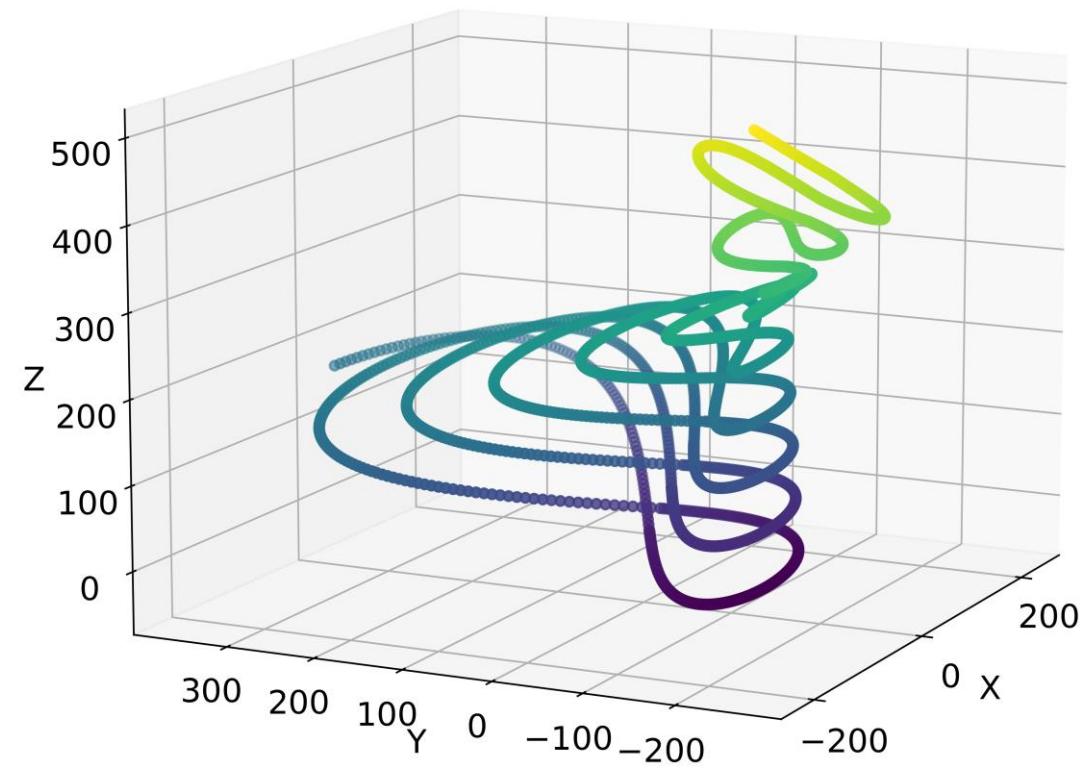
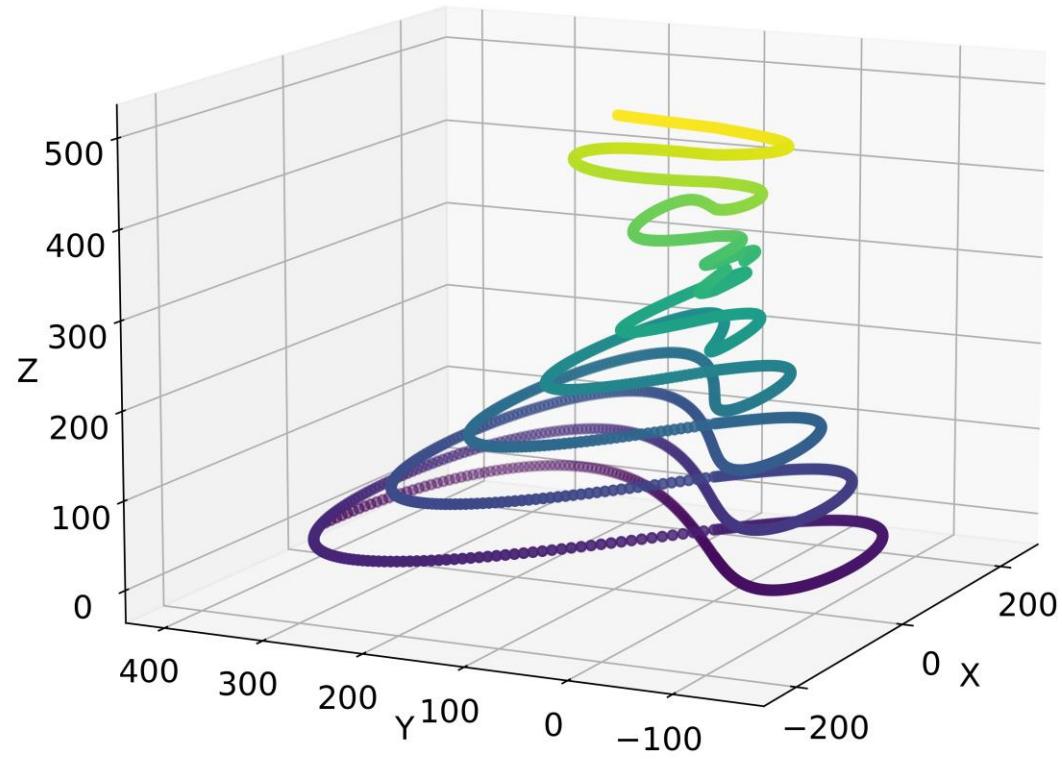


# Toolpath 3 and Score

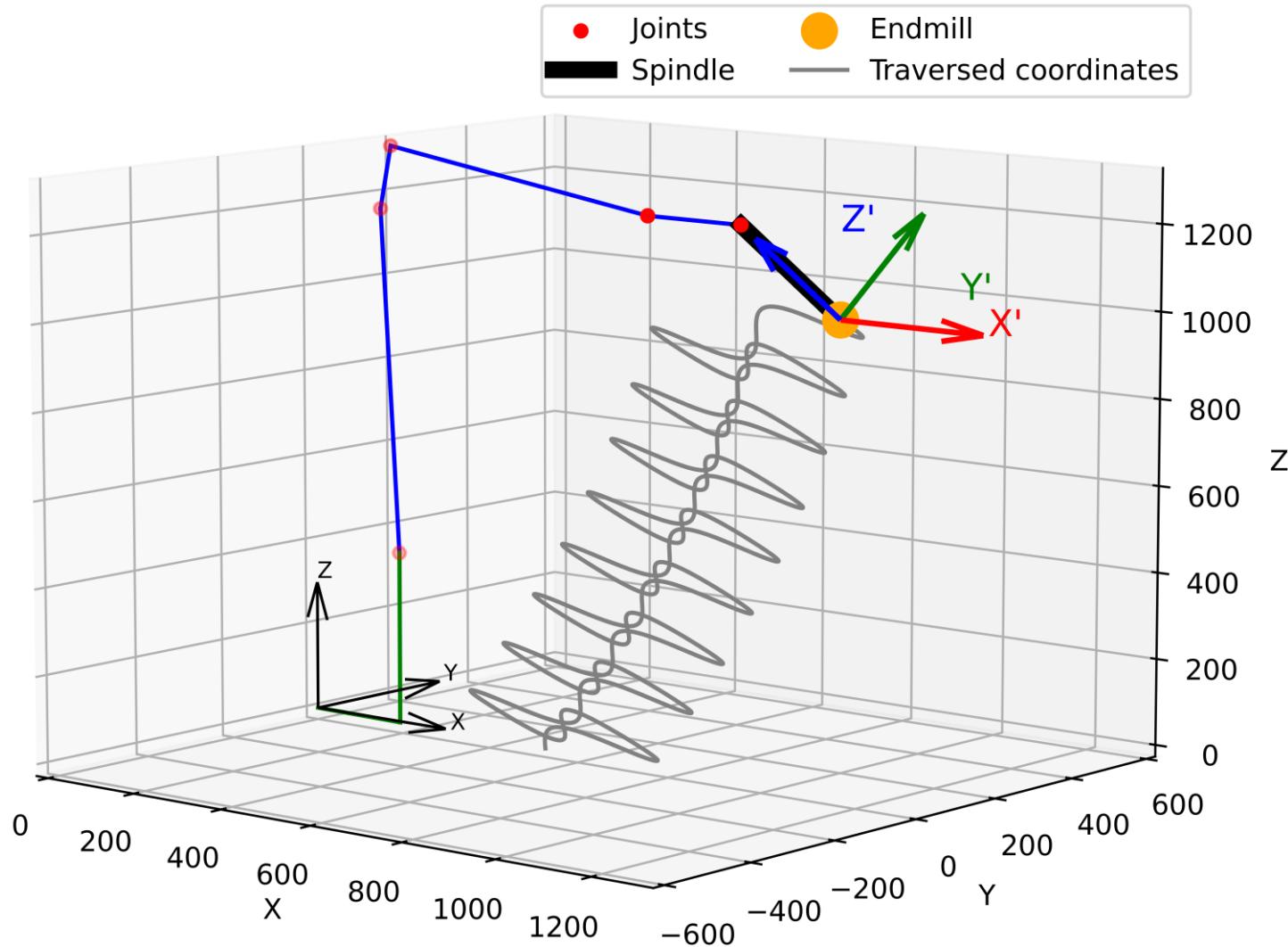




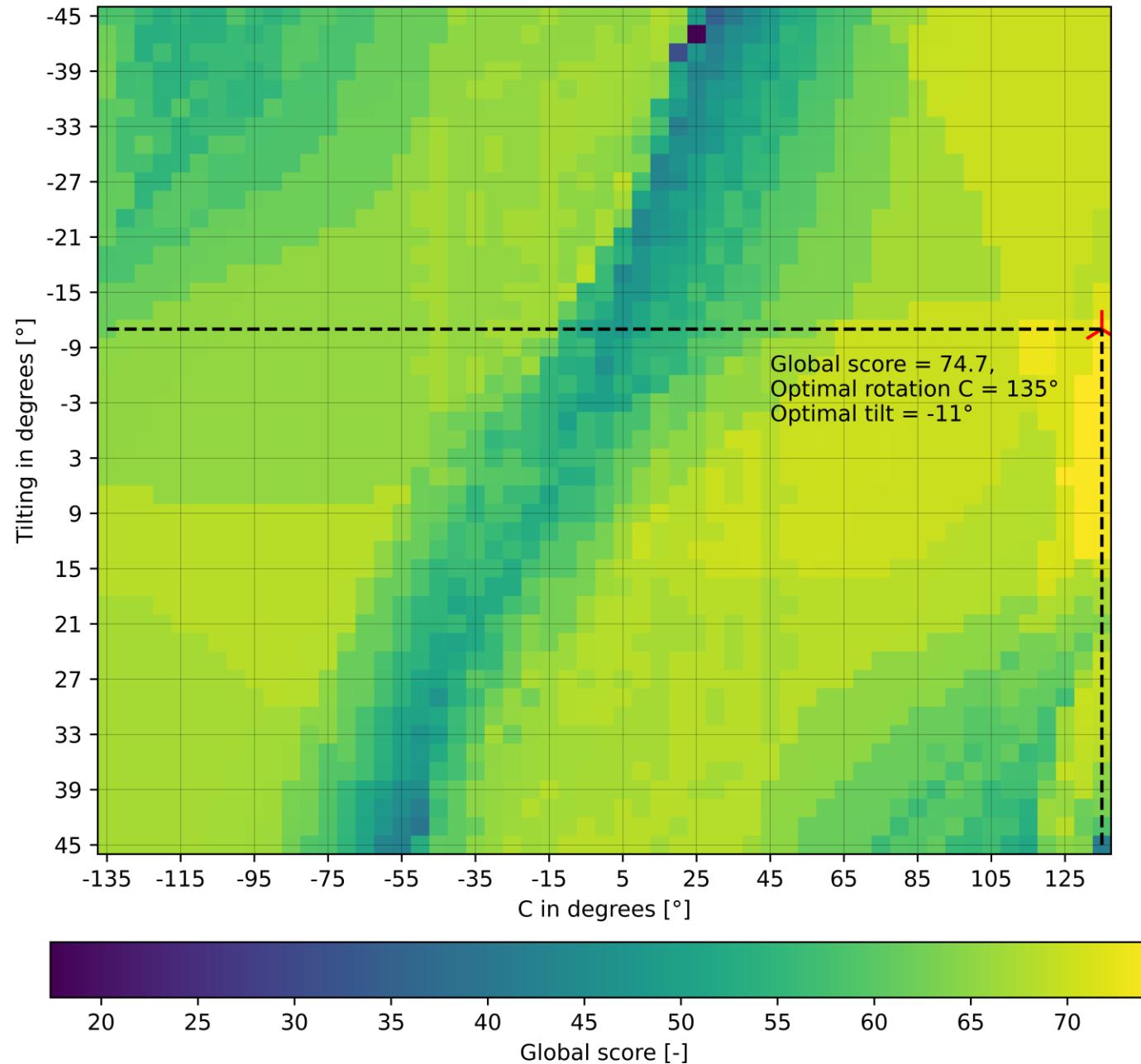
# Rotation toolpath 2



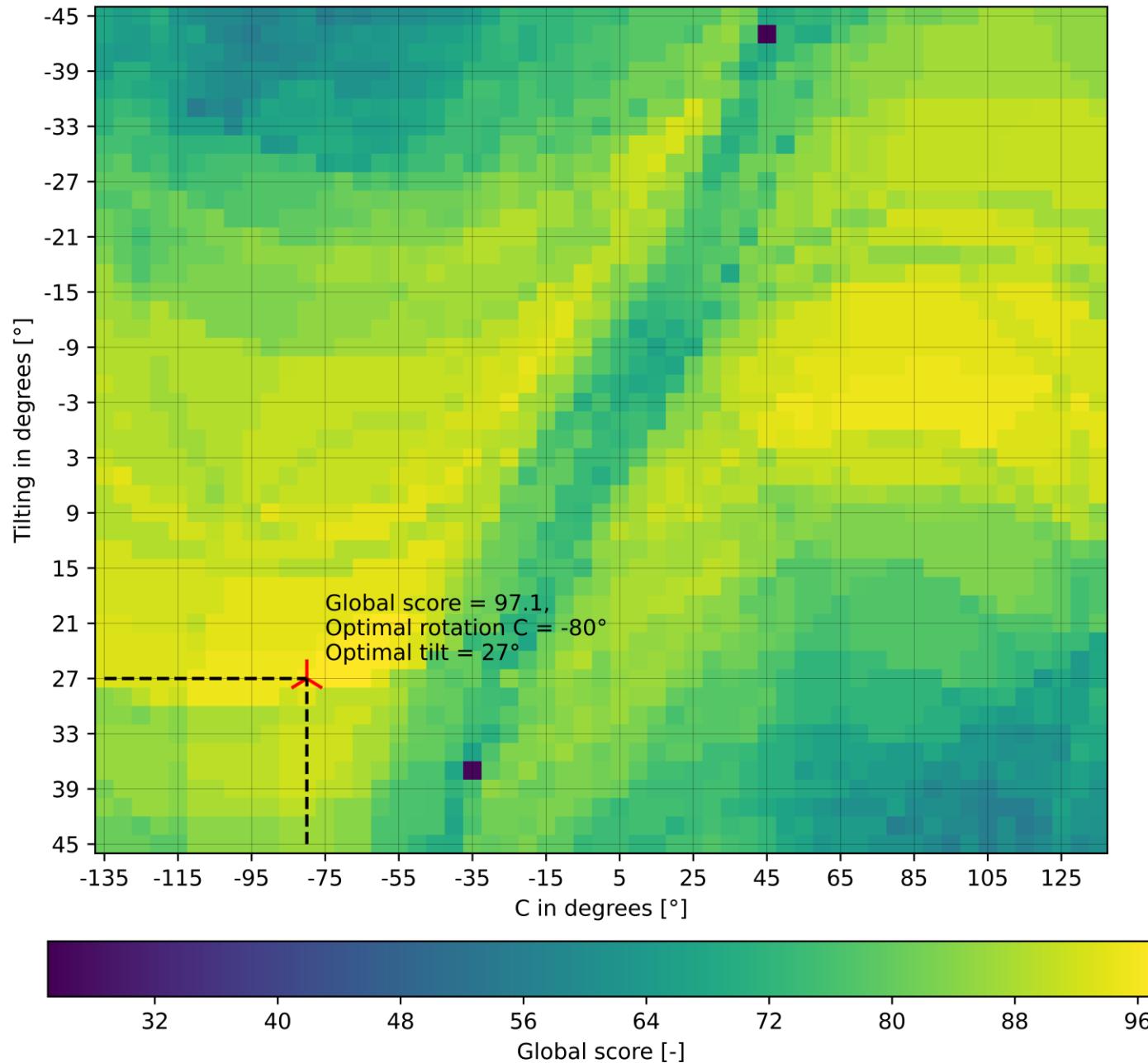
# Toolpath 3



Score of the individual boundary conditions as a hyperplane



## Score of the individual boundary conditions as a hyperplane



## Score of the individual boundary conditions as a hyperplane

