



Robótica de Manipulação 2024/2025

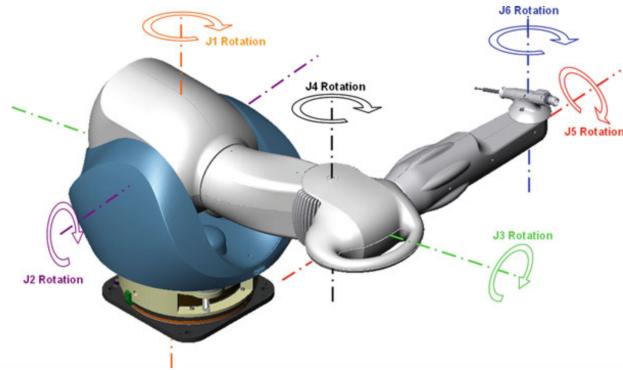
Prof. Jorge Martins (jorgemartins@tecnico.ulisboa.pt)
Prof. Rui Coelho (rui.coelho@tecnico.ulisboa.pt)

Trabalho Laboratorial: Mako Surgical Robot

Mestrado em Engenharia Mecânica

Mestrado em Engenharia Biomédica

The proposed work consists in the Kinematic and Dynamic analysis of the *Mako* surgical manipulator by *Stryker*. In your report, provide complete, concise answers to the points below, using your own code. For developing your computational models, a Symbolic Robotics Matlab Toolbox is provided in the course webpage, as the starting point.



Kinematics: (Due on May 25th)

1. Analyze the robot's drawing and build its kinematic model according to the Denavit-Hartenberg convention. To do so, draw your own joints diagram with the link reference frames and the corresponding table of parameters.
2. Using the provided Toolbox, create a Simulink model for the direct kinematics of the robot. Validate your model by placing the robot in configurations for which the end-effector's position and orientation can be easily determined.
3. Following a kinematic decoupling approach present one closed-form solution for the inverse kinematics of the robot. Create the Simulink model and validate your model against the direct kinematics model implemented in Point 2.
4. Create a Simulink model for the geometric Jacobian of the robot. Validate the result through numerical differentiation of the direct kinematics in Point 2. Determine the kinematic singularities of the robot and show their effect on the rank of the Jacobian matrix.
5. Using the direct kinematics and the geometric Jacobian models, determined in Points 2 and 4, implement a Simulink model for the Closed Loop Inverse Kinematics of the robot. Validate your model against the results of Point 3.

Dynamics and Control: (Due on June 15th)

1. Using simple solid shapes (bars, cylinders, etc), calculate approximate values for the mass, center of mass, and inertia tensor for each link.
2. Create a Simulink model for the dynamics of the robot (See Appendix B). You are free to choose either the Lagrange-Euler or the Newton-Euler method (Note: The Newton-Euler formulation is quicker to build the Simulink blocks for the dynamics).
3. Calculate the worst-case inertia configuration for each joint of the robot manipulator. Then, design and implement decentralized PID type joint controllers for the robotic arm.
4. Design and implement a centralized inverse dynamics controller for the robot arm.
5. Choose a task for the end-effector and design the corresponding task-space trajectory. Compare the performance of the controllers in points 3 and 4 for your task.

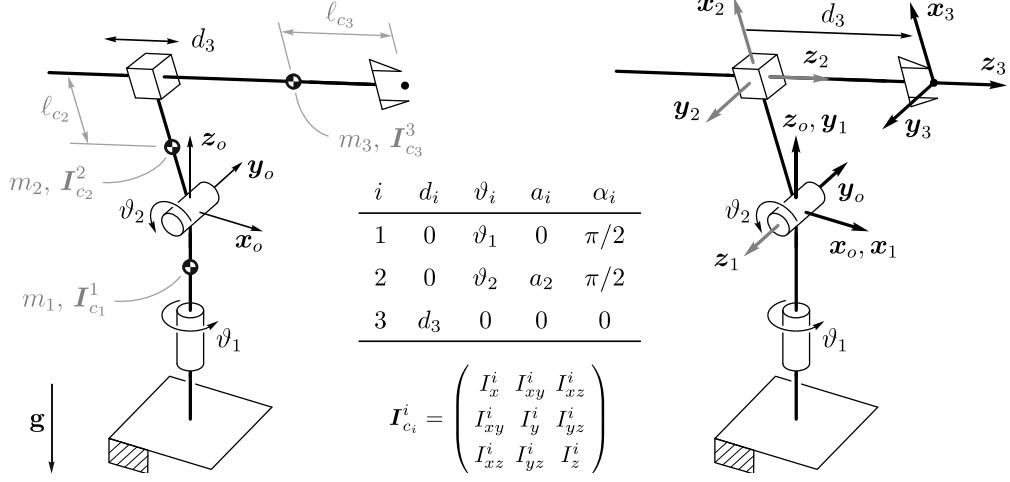
Note on Matlab toolboxes

To verify that you have the necessary Matlab and Simulink toolboxes, you may execute the following commands in Matlab's command window.

```
>> syms x  
>> y=x^2  
>> new_system('model_name')  
>> open_system('model_name')  
>> matlabFunctionBlock('model_name/parabola',y)  
>> vredit
```

Appendix A. Validating the Dynamics Implementation

Before building the model for your robot, test your code on the simpler system given below:



$$\begin{aligned}
B_{11} &= \left(I_z^2 - I_x^2 + I_z^3 - I_x^3 + m_2 (a_2 - \ell_{c2})^2 + m_3 a_2^2 - m_3 (d_3 - \ell_{c3})^2 \right) \cos(\vartheta_2)^2 \\
&\quad + (m_3 a_2 (d_3 - \ell_{c3}) - I_{xz}^2 - I_{xz}^3) \sin(2\vartheta_2) + I_y^1 + I_x^2 + I_x^3 + m_3 (d_3 - \ell_{c3})^2 \\
B_{12} = B_{21} &= (I_{xy}^2 + I_{xy}^3) \sin(\vartheta_2) - (I_{yz}^2 + I_{yz}^3) \cos(\vartheta_2) \\
B_{13} = B_{31} &= 0 \\
B_{22} &= I_y^2 + I_y^3 + m_3 (d_3 - \ell_{c3})^2 + (a_2 - \ell_{c2}) (m_3 a_2 + m_2 (a_2 - \ell_{c2})) + m_3 a_2 \ell_{c2} \\
B_{23} = B_{32} &= -m_3 a_2 \\
B_{33} &= m_3
\end{aligned}$$

$$\begin{aligned}
\dot{\phi}_1 &= - \left(I_z^2 - I_x^2 + I_z^3 - I_x^3 + m_2 (a_2 - \ell_{c2})^2 + m_3 a_2^2 - m_3 (d_3 - \ell_{c3})^2 \right) \sin(2\vartheta_2) \dot{\vartheta}_1 \dot{\vartheta}_2 \\
&\quad + 2 (m_3 a_2 (d_3 - \ell_{c3}) - I_{xz}^2 - I_{xz}^3) \cos(2\vartheta_2) \dot{\vartheta}_1 \dot{\vartheta}_2 \\
&\quad + [(I_{xy}^2 + I_{xy}^3) \cos(\vartheta_2) + (I_{yz}^2 + I_{yz}^3) \sin(\vartheta_2)] \dot{\vartheta}_2^2 \\
&\quad + m_3 [(d_3 - \ell_{c3}) (1 - \cos(2\vartheta_2)) + a_2 \sin(2\vartheta_2)] \dot{\vartheta}_1 \dot{d}_3 \\
\dot{\phi}_2 &= (I_{xz}^2 + I_{xz}^3 - m_3 a_2 d_3 + m_3 a_2 \ell_{c3}) \dot{\vartheta}_1^2 \cos(2\vartheta_2) + 2 m_3 \dot{\vartheta}_2 \dot{d}_3 (d_3 - \ell_{c3}) \\
&\quad + \frac{1}{2} \left(I_z^2 - I_x^2 + I_z^3 - I_x^3 + m_2 (a_2 - \ell_{c2})^2 + m_3 a_2^2 - m_3 (d_3 - \ell_{c3})^2 \right) \dot{\vartheta}_1^2 \sin(2\vartheta_2) \\
\dot{\phi}_3 &= -\frac{1}{2} m_3 \left(a_2 \sin(2\vartheta_2) + 2 (d_3 - \ell_{c3}) \sin(\vartheta_2)^2 \right) \dot{\vartheta}_1^2 - m_3 (d_3 - \ell_{c3}) \dot{\vartheta}_2^2
\end{aligned}$$

$$g_1 = 0$$

$$g_2 = m_3 |\mathbf{g}| (a_2 \cos(\vartheta_2) + (d_3 - \ell_{c3}) \sin(\vartheta_2)) + m_2 |\mathbf{g}| (a_2 - \ell_{c2}) \cos(\vartheta_2)$$

$$g_3 = -m_3 |\mathbf{g}| \cos(\vartheta_2)$$

$$\begin{pmatrix} B_{11} & B_{12} & B_{13} \\ B_{21} & B_{22} & B_{23} \\ B_{31} & B_{32} & B_{33} \end{pmatrix} \begin{pmatrix} \ddot{\vartheta}_1 \\ \ddot{\vartheta}_2 \\ \ddot{d}_3 \end{pmatrix} + \begin{pmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{pmatrix} + \begin{pmatrix} g_1 \\ g_2 \\ g_3 \end{pmatrix} = \boldsymbol{\tau}$$