

Mini-project 3

Singularity Analysis of a Parallel Robot

In this mini-project you will combine your knowledge on polynomial system solving and kinetostatics to delimit the workspace of a fully parallel robot, and to find the forward kinematic singularities present in its interior. You will use the resulting workspace maps to devise singularity-free motion paths for the robot. Overall, you are brought close to a real scenario encountered by a robot designer. All files needed to solve the mini-project are available in

<http://www.iri.upc.edu/people/ros/Master/GFR/MP3.zip>.

Analysis to be performed

Fig. 1 shows a schematic view of a 3-RRR parallel manipulator. The coordinates of the base joints in the ground coordinate system OXY are

$$J_{1,1} = (0, 0) \quad J_{1,2} = (2.350, 0) \quad J_{1,3} = (1.175, 2.035)$$

and those of the platform joints in the platform coordinate system $O'X'Y'$ are

$$J_{3,1}^r = (0, 0) \quad J_{3,2}^r = (1.2, 0) \quad J_{3,3}^r = (0.6, 1.03)$$

The length of links 2, 3, and 4 is $l_1 = 1$. The length of links 5, 6, and 7 is $l_2 = 1.350$. We attach a local coordinate system to every link i (indicated in red in Fig. 1) and define θ_i as the angle between such system and OXY .

1. **(4 points)** The file `cspace.cuik` allows you to compute the C-space of the robot for a constant value $\theta_8 = 0$. Make a copy of this file, saving it with the name `forward.cuik`. Also make a copy of the file `cspace.param`, saving it with the name `forward.param`. Add additional variables and equations to `forward.cuik` in order to determine the forward kinematic singularities of the manipulator, assuming that it moves with $\theta_8 = 0$. Use the hints at the bottom to this end.

Explain the new variables and equations introduced, how you obtained the bounds for all variables, and include the resulting `forward.cuik` and `forward.param` files in your miniproject report.

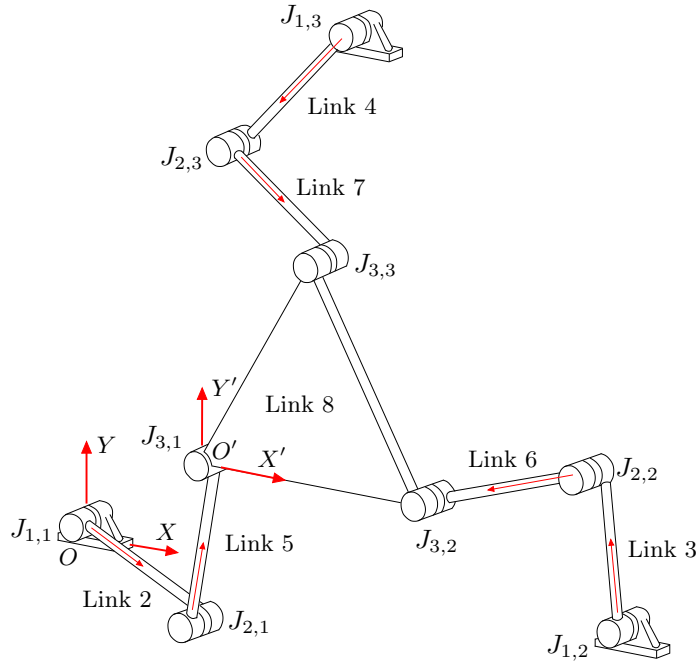


Figure 1: A 3-RRR manipulator. All coordinate systems are indicated in red. For the systems of the leg links, only the X axis is indicated, the Y axis being at 90° counterclockwise.

2. (1 point) Execute

```
~/CuikSuite/bin/cuik forward
~/CuikSuite/bin/cuiksols2samples forward
```

This produces a file `forward.links` describing the coordinates of the forward singularity locus. Plot the results using the MATLAB command

```
plotLoci('forward.links')
```

available in file `plotLoci.m` and verify that you obtain the picture in Fig. 2.

Warning: The preparation of the `forward.cuik` file is prone to errors/typos. To ease the debugging of the file, initially use larger values of the σ parameter in `forward.param`, and limit the computations to a smaller region of the workspace.

Plot interpretation: Note that the function `plotLoci.m` plots the points specified in `forward.links`, in red, on top of a pre-computed diagram colored in grey and blue. The blue curve gives the locus of positions for $J_{3,1}$ for which the robot is in an inverse kinematic singularity. Hence the inner grey region between such curves is the position workspace for $J_{3,1}$. The red curve gives the locus of positions for $J_{3,1}$ for which there is a forward kinematic singularity. It corresponds to configurations in which the moving platform loses its stiffness.

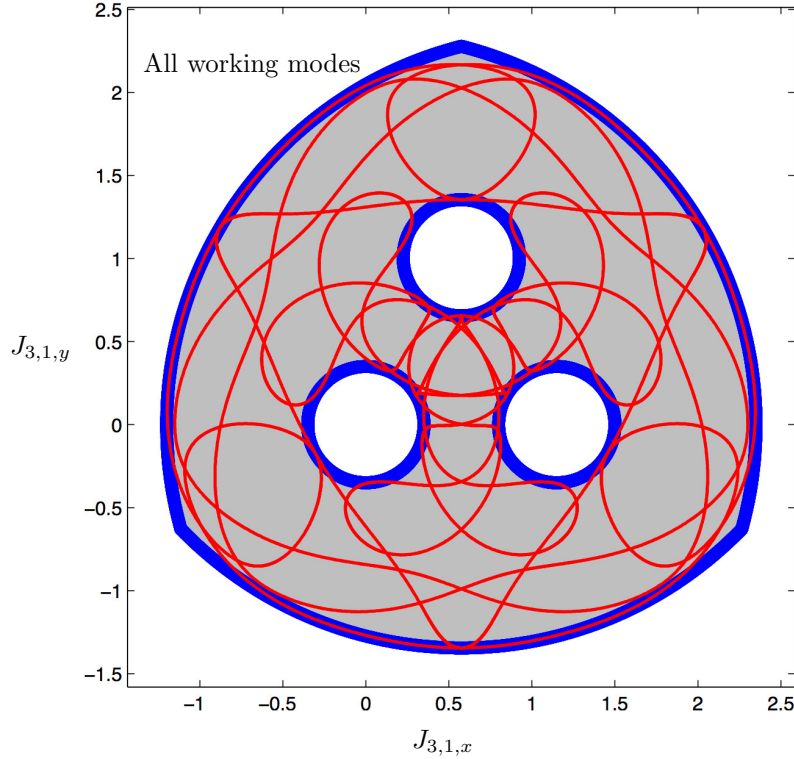


Figure 2: Singularity loci of the 3-RRR manipulator for $\theta_8 = 0$, projected onto the position workspace of $J_{3,1}$. The red and blue curves respectively correspond to the forward and inverse singularity loci. The red and blue points are the loci of positions for joint $J_{3,1}$ corresponding to such singularities.

3. (2 points) We can define the working mode of the i -th leg as the sign of the product

$$\hat{s}_i^T \cdot \hat{S}_{1,i},$$

where \hat{s}_i and $\hat{S}_{1,i}$ are the Plücker vectors defined in the course. Note that the working mode is *positive* if the triangle $J_{1,i}J_{2,i}J_{3,i}$ has positive orientation, and *negative* otherwise. Using this terminology, one can define the overall working mode of the manipulator using a sign triple. The triple $(+, +, -)$, for example, indicates that legs 1, 2, and 3 are working under modes $+$, $+$, and $-$, respectively. Add the three inequalities

$$\hat{s}_1^T \cdot \hat{S}_{1,1} \geq 0$$

$$\hat{s}_2^T \cdot \hat{S}_{1,2} \geq 0$$

$$\hat{s}_3^T \cdot \hat{S}_{1,3} \leq 0$$

to the previous `forward.cuik` file in order to compute the forward singularity sub-locus for the $(+, +, -)$ mode. Verify that you obtain the plot in Fig. 3 and provide the resulting `*.cuik` and `*.param` files.

4. (1 point) Compute and provide plots similar to that in Fig. 3 for all possible working modes.

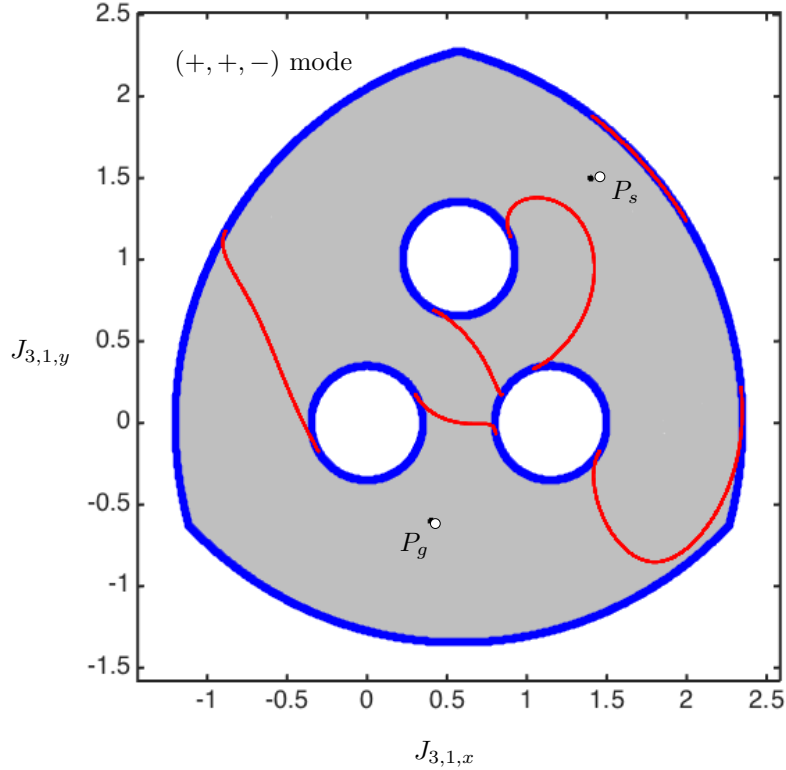


Figure 3: The forward singularity sub-locus for the $(+, +, -)$ working mode (in red), with the start and goal positions, P_s and P_g of the path planning problem.

5. **(2 points)** The manipulator has to be used in a pick-and-place operation while keeping $\theta_8 = 0$ all the time. Using the obtained plots, qualitatively describe a C-space path allowing to bring $J_{3,1}$ from the position $P_s = (1.4, 1.5)$ to the position $P_g = (0.4, -0.6)$ indicated in Fig. 3, assuming that the working modes in such positions are $(+, +, -)$ and $(+, +, +)$, respectively. To keep the control of the platform at all times the path must avoid crossing the forward singularity locus. Is the platform dexterity kept along the path?

Hints

Recall that the forward singularities of this manipulator are the configurations for which the matrix $\mathbf{A} = [\hat{s}_1, \hat{s}_2, \hat{s}_3]^T$ is rank deficient, where $s_i = [L_i, M_i, R_i]^T$ is any Plücker vector of the line through $J_{2,i}$ and $J_{3,i}$. To compute the forward singularity locus, first add the following equations to define the \hat{s}_i vectors

$$\hat{s}_i = \hat{S}_{2,i} \times \hat{S}_{3,i} \quad i = 1, 2, 3.$$

Second, note that \mathbf{A} is rank deficient if, and only if, it has a non-null vector $\hat{T} = [v_{ox}, v_{oy}, \omega]^T$ in its kernel. This condition can be expressed by adding the

equations

$$\begin{aligned}\mathbf{A} \cdot \hat{T} &= \mathbf{0} \\ \hat{T}^\top \cdot \hat{T} &= 1\end{aligned}$$

to the `*.cuik` file. The vector \hat{T} can be interpreted as a platform twist that is compatible with a zero velocity at the actuators. Finally, note that when \mathbf{A} is singular, its kernel is one-dimensional in general, and hence the two previous equations will commonly present two solutions for \hat{T} . To avoid unnecessary computations, we can add the following inequality to the file

$$\mathbf{a}^\top \cdot \hat{T} \geq 0 \quad (\mathbf{a} \text{ is a random unit vector of } \mathbb{R}^3)$$

which selects the value of \hat{T} on the half-plane $\mathbf{a}^\top \cdot \hat{T} \geq 0$. The “ \geq ” sign is written as “`>=`” in the CUIK syntax.

You will have to provide conservative bounds for the feasibility intervals of all variables introduced. Since all variables can be interpreted geometrically, simple reasonable bounds can be defined in all cases.