ECOLE CENTRALE DE NANTES

DESIGN OF ROBOTS PROJECT REPORT

Presented by

Mellacheruvu Prithvi Bharadwaj

On January 21, 2022

Design of a three degrees of freedom planar parallel manipulator

Evaluators:	Stephane Caro	Research Director, LS2N, Nantes
Supervisors:	Durgesh Haribhau Salunkhe	PhD student, LS2N, Nantes

Acknowledgements

I sincerely thank Prof. Stéphane Caro for providing me with the opportunity to work on this project. It has been a very enriching experience.

I am very grateful to Mr. Durgesh Haribhau Salunkhe, doctorant LS2N, for helping me with the successful completion of this project.

My heartfelt thanks to Shourie G.S., my good friend and fellow classmate for his assistance in the report generation and MATLAB.

I also thank the Ecole Centrale de Nantes and LS2N laboratory for this opportunity.

THANK YOU

List of Figures

Contents

1 Introduction

This is a project to design a 3 degrees of freedom (DOF) parallel planar manipulator (PPM). This project describes the design, kinematic analysis and an optimization problem to minimize the mechanism footprint / link lengths for compactness while satisfying a given workspace criterion. A working video of the simulated mechanism is also made along with stress analysis for a given load of 10N on the end-effector normal to the plane of motion.

A parallel planar manipulator consists of an end effector or a moving platform, connected to the ground by several serial kinematic chains (legs or limbs of the parallel manipulator), with a few actuated joints. They find applications where there is a requirement of high dynamics. Usually, parallel manipulators have better payload capacity as compared to serial robots as they do not need to carry the weight of actuators, but PPMs have the disadvantage of not being able to carry a large payload with the weight acting in the direction normal to the plane of operation.

This project describes a 2-$\underline{R}$RR 1-$\underline{P}$RR manipulator, i.e., two limbs with three rotary joints and one limb with a prismatic joint to base and two rotary joints and where all the base joints are actuated. Fig.1 illustrates the aforementioned manipulator, in its plane of operation.

In the zip file, find the assemblies and parts in folder – “ModelforInventor”. The assembly “MAIN Assembly.iam” is the assembly that can be used to move around the end-effector with the cursor. The video file “DESRO Mechanism Video Presentation.mp4” contains the working along with explanation. The MATLAB code for the optimisation of the design problem for this manipulator can be found in the ”MATLAB Code” folder. Finally, the file “Stress Analysis Report.html” contains the complete stress analysis.

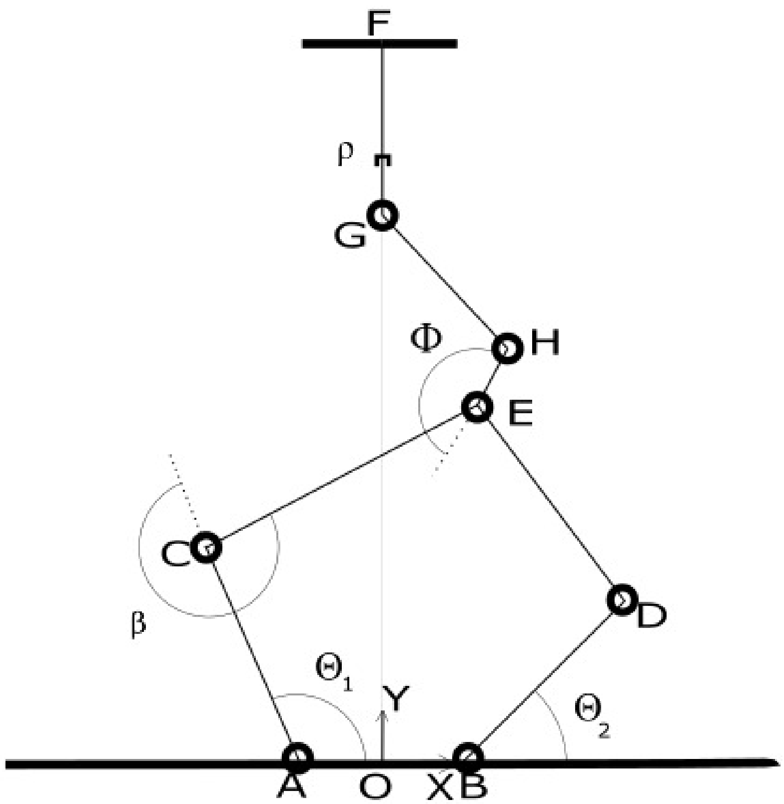

Figure 1: 2-RRR 1-PRR manipulator

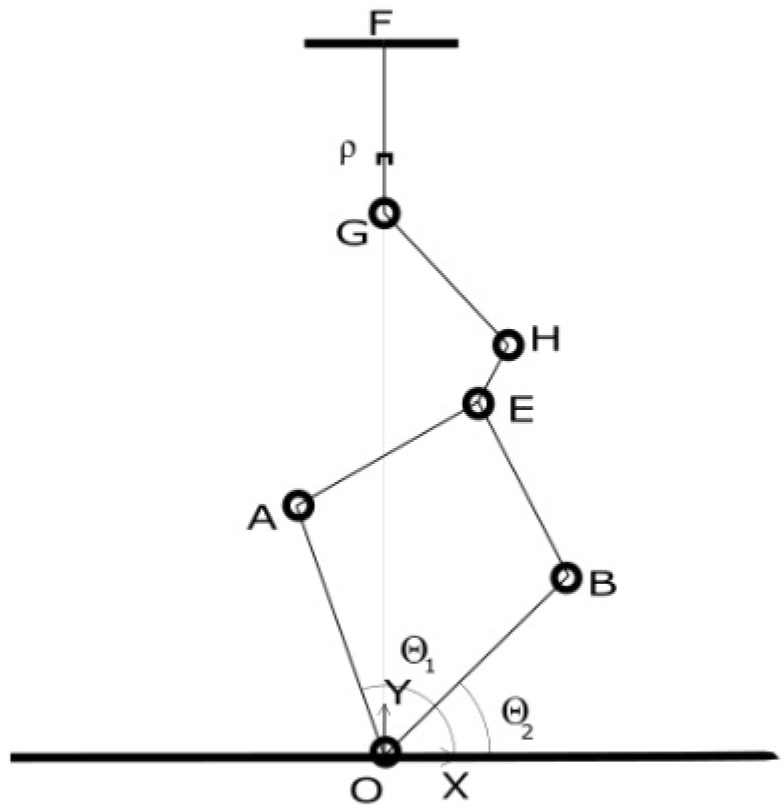

Figure 2: Same manipulator as in 1 with ground link length zero

2 Manipulator architecture

The chosen architecture is illustrated in fig.1. It has three limbs. Two limbs are of RRR type, i.e. with three rotary joints, of which the first one (connected to the ground) is actuated, and a limb of type PRR with the actuated prismatic joint. In figure 2:

- limbs OAE and OBE illustrate the RRR limbs.
- the other limb GHE is of type PRR, with a prismatic joint connected between ground at F and G i.e., F is the limit for the slider point G
- End-effector is the circular link with the center at point E which gives us the position coordinates X & Y and the orientation of end effector ϕ is based on angle of vector HE.
- O and F are ground points.
- X, Y and ϕ form the output variables in the Direct Geometric Model.
- the input variables are the joint angles of the actuated rotary joints $\theta 1$, $\theta 2$ and the length of the prismatic actuator ρ.

- the input variables are the joint angles of the actuated rotary joints θ_1, θ_2 and the length of the prismatic actuator ρ.
- a, b: correspond to the length OA, OB which are equal (a=b)
- c, d, Cr: length AE, BE GH which are equal (c=d=Cr)
- r: Crank length or distance between point H & E
- r: Crank length or distance between point H & E
- Cr stands for Connecting Rod and Crank length r for crank in the slider-crank mechanism notation.

3 Geometric and kinematic modeling

Given the geometric parameters of the manipulator the inverse geometric, forward kinematic and the inverse kinematic model of the manipulator can be deduced. Forward geometric model can be described as:

$$x_p = f(\psi)$$

(1)

with

$$\psi = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \rho \end{bmatrix} \quad (2)$$

and

$$x_p = \begin{bmatrix} x \\ y \\ \phi \end{bmatrix} \quad (3)$$

To derive the inverse geometric model, method of intersection of circles can be used. The following equations from the inverse geometric model of the manipulator help us in calculating the points A, B, H as defined in the figure. With these points we can calculate the respective unit vectors which are required in the Kinematic Jacobian Matrices. The figure:2 is used as reference for the below derivation:

$$d = \sqrt{x^2 + y^2} \quad (4)$$

$$g = \frac{a^2 - c^2 + d^2}{2d} \quad (5)$$

$$h = \sqrt{a^2 - g^2} \quad (6)$$

$$B_x = \frac{xg + hy}{d}; B_y = \frac{gy + hx}{d} \quad (7)$$

$$A_x = \frac{xg - hy}{d}; A_y = \frac{gy + hx}{d} = B_y \quad (8)$$

$$H_x = r\cos\phi + x; H_y = r\sin\phi + y \quad (9)$$

$$\psi = g(x_p) \quad (10)$$

Hence, equations 3 and 10 represent the forward and inverse geometric models of the manipulator.
Now, we move ahead to find the forward and inverse kinematic models of the manipulators. We follow the same methodology as Stéphane Caro et al used in [1]. Let $\widehat{u}_1$, $\widehat{v}_1$, $\widehat{l}$ and $\widehat{m}$ represent unit vectors along AC, CE, GH and HE; let $\widehat{u}_2$, $\widehat{v}_2$ represent unit vectors along BD, DE; The figure:1 is used as reference for the below derivation. $\overrightarrow{P}$ represents the position vector of E.

$$\overline{OE} = \overline{OA} + \overline{AC} + \overline{CE} \tag{11}$$

$$OA\left(-\widehat{i}\right) + AC\widehat{u} + CE\widehat{v} \tag{12}$$

$$\overline{P} = \overline{OE}$$

$$\frac{dP}{dt} = 0 + AC\frac{d\theta_1}{dt}E\widehat{u} + cE\frac{d\left(\theta + \beta\right)}{dt}E\widehat{v} \tag{13}$$

where

$$i = 1, 2$$

$$E = Rot\left(\frac{-\pi}{2}\right) = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

Multiplying with $\widehat{v_i}^T$ on both sides of the equation and some rearrangement gives us the following:

$$\widehat{v}^T\frac{d\overline{P}}{dt} = AC\widehat{v}^T E\widehat{u}\frac{d\theta}{dt} \tag{14}$$

Now lets consider the prismatic link.

$$\overrightarrow{OE} = \overrightarrow{OF} + \overrightarrow{FG} + \overrightarrow{GH} + \overrightarrow{HE} \tag{15}$$

$$\overrightarrow{OE} = OF\widehat{j} + FG\left(-\widehat{j}\right) + GH\widehat{l} + HE\widehat{m} \tag{16}$$

$$\widehat{l}^T\frac{d\overline{P}}{dt} = -\frac{d\rho}{dt}\widehat{l}^T\widehat{j} + HE\frac{d\phi}{dt}\widehat{l}^T E\widehat{m} \tag{17}$$

$$HE\frac{d\phi}{dt}\widehat{l}^T E\widehat{m} - \widehat{l}^T\frac{d\overline{P}}{dt} = \frac{d\rho}{dt}\widehat{l}^T\widehat{j} \tag{18}$$

The following matrix form may be obtained by combining equation 14 with $i = 1, 2$ and equation 18:

$$\begin{bmatrix} \widehat{v_1}^T & 0 \\ \widehat{v_2}^T & 0 \\ -\widehat{l}^T & HE\widehat{l}^T E\widehat{m} \end{bmatrix} \begin{bmatrix} \frac{d\overrightarrow{e}}{dt} \\ \frac{d\phi}{dt} \end{bmatrix} = \begin{bmatrix} AC\widehat{v1}^T E\widehat{u1} & 0 & 0 \\ 0 & AC\widehat{v2}^T E\widehat{u2} & 0 \\ 0 & 0 & \widehat{l}^T\widehat{j} \end{bmatrix} \begin{bmatrix} \frac{d\theta}{dt} \\ \frac{d\rho}{dt} \end{bmatrix} \tag{19}$$

$$M\dot{x_p} = N\dot{\psi} \tag{20}$$

where ; HE is the length of link HE and not confused with E

$$M = \begin{bmatrix} \widehat{v_1}^T & 0 \\ \widehat{v_2}^T & 0 \\ -\widehat{l}^T & HE\widehat{l}^T E\widehat{m} \end{bmatrix}$$

$$N = \begin{bmatrix} AC\widehat{v1}^T E\widehat{u1} & 0 & 0 \\ 0 & AC\widehat{v2}^T E\widehat{u2} & 0 \\ 0 & 0 & \widehat{l}^T\widehat{j} \end{bmatrix}$$

$$J_i\dot{x_p} = \dot{\psi} \tag{21}$$

$$J_f\dot{\psi} = \dot{x_p} \tag{22}$$

where J_i and J_f are inverse and forward kinematic Jacobian matrices corresponding to:

$$J_i = N^{-1}M$$
$$J_f = M^{-1}N$$

4 Optimization problem

The optimization problem aims to generate the optimum design parameters in order to have a wide functional workspace with a mechanism that is as compact as possible. The criteria that quantifies the objective of having a wide functional workspace is that the workspace must be cylindrical and at least have a diameter of 100mm, 60° manipulability and that the inverse condition numbers of the forward and inverse kinematic Jacobian matrices are greater than 0.1. To quantify the objective of having a compact mechanism we could consider the radius of the circle that encapsulates the mechanism in its home position. Instead, we could also quantify compactness using one parameter of the manipulator. For that we could consider the aspect of symmetry (which is important in every mechanism) and parametrize the dimensions of entire manipulator to depend on just one parameter - the length of the links. The following could be postulated: if lengths of links OA, OB are l, let length AE, BE, GH be equal to $2 * l/3$ and length HE be r = 12. Hence, to obtain a compact manipulator, the length l must be minimized.

To fulfil the criteria of the inverse condition numbers to be greater than 0.1 throughout the workspace is a very strict one, and we relax the criteria to 0.0001

Objective function:
minimize l
Constraints:

$$0.001 - abs\,(ICN_M) \leq 0$$
$$0.001 - abs\,(ICN_N) \leq 0$$

where ICN_M and ICN_M are the inverse condition numbers of matrices M and N obtained for a circular region of workspace with diameter 100mm with an arbitrarily chosen center $(0, l)$

Bounds of link lengths may be chosen as 15mm and 250mm

The MATLAB code (find in appendix) uses the $fmincon$ MATLAB function to optimize the function parameter l. We vary the length l, which leads to changes in other dependent parameters as mentioned before. The *cCeqReturnFunction* function varies positions that are considered for the end-effector. For every position within the circle of diameter 100mm centered at (0, l), at end-effector orientations ϕ between -60° and -30° the inverse condition numbers are evaluted. Based on these values vectors c and ceq are calculated for the $fmincon$ function.

Using the $fmincon$ function from MATLAB we could optimize the link length to satisfy the criteria of having a compact manipulator and also to have functional workspace of diameter 100mm where the inverse condition numbers of the forward and inverse kinematic Jacobian matrices are greater than 0.0001. With an initial link length 130mm, the link length may be optimized to 127mm as shown by the following MATLAB output. This is only the local minimum. By varying the initial length and other parameters, a different optimized length may be obtained. MATLAB code can be in appendix.

```
 Iter  Func-count            Fval   Feasibility   Step Length       Norm of   First-order
                                                                       step    optimality
    0           2    1.300000e+02     0.000e+00     1.000e+00     0.000e+00     1.000e+00
    1           4    1.290000e+02     0.000e+00     1.000e+00     1.000e+00     1.000e+00
    2           6    1.277190e+02     9.515e-07     1.000e+00     1.281e+00     2.508e-01
    3           8    1.277237e+02     1.229e-11     1.000e+00     4.682e-03     1.614e-04
    4          10    1.277237e+02     1.323e-17     1.000e+00     6.046e-08     7.786e-08

Local minimum found that satisfies the constraints.

Optimization completed because the objective function is non-decreasing in
feasible directions, to within the value of the optimality tolerance,
and constraints are satisfied to within the value of the constraint tolerance.

<stopping criteria details>

le =

  127.7237
```

Figure 3: Optimization output

5 Design of the manipulator

The basis for this mechanism has stemmed from the idea of decoupling position and orientation control which is achieved by effectively overlaying a slider-crank mechanism (orientation control) over the 5-bar planar parallel mechanism (position control). The genesis of this idea was during my interaction with Mr. VS Rajashekhar (Independent Researcher) in the conference named “iNaCoMM” 2019 in IIT Mandi, India were the paper [3] was also published about a 5-bar 2-DOF mechanism.
This particular orientation of the 5-Bar mechanism is used to have an increased area of the workspace as explained in this paper [2]. Hence the length of the ground link was made zero, thereby the axes of rotary joints of the 2 links coincide at O.
Fig.1 illustrates the basic schematic of the manipulator. The following illustrate the 3D CAD model of the manipulator as modeled on Autodesk Inventor:

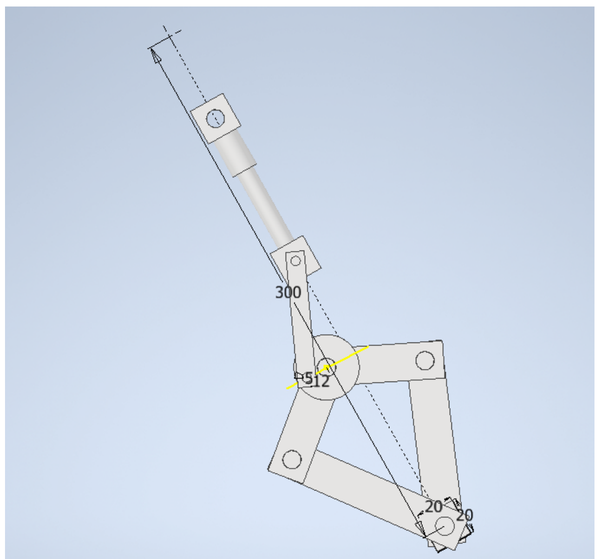

Figure 4: Top view of the manipulator

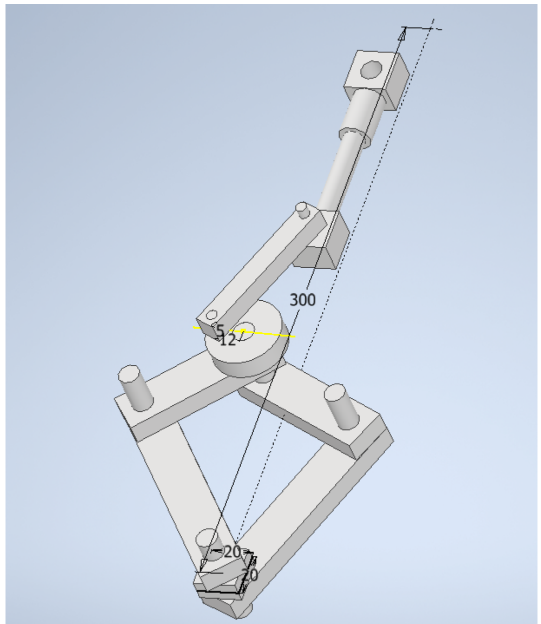

Figure 5: Isometric view of the manipulator

As shown in the optimization section, and also by moving around the end-effector on the CAD software, the manipulator has a wide workspace. This workspace also includes a circular region of diameter 100mm, where the end-effector has freedom to rotate at least 60° and the inverse condition numbers of both forward and inverse kinematic Jacobian matrices are greater than 0.001.

6 Stress Analysis

For any mechanical structure, structural rigidity and strength are very important characteristics. The payload that the structure can carry depends on these parameters. The manipulator is designed to pick, place and handle objects which apply load on the manipulator, and the manipulator is supposed to be able to withstand the loads with minimal deformation. A force of 10N has been applied on the end-effector, normal to plane of operation. Fixed part is the base (in fig.5) and all the parts in the manipulator are made of ABS plastic, as this is the widely used material for 3-D printing. The following illustrate the material properties and the displacement analyis for the manipulator. The maximum displacement was found to be 0.1504mm as shown below:

Material(s)

Name	ABS Plastic	
General	Mass Density	1.06 g/cm^3
	Yield Strength	20 MPa
	Ultimate Tensile Strength	29.6 MPa
Stress	Young's Modulus	2.24 GPa
	Poisson's Ratio	0.38 ul
	Shear Modulus	0.811594 GPa

Figure 6: Material properties - ABS plastic

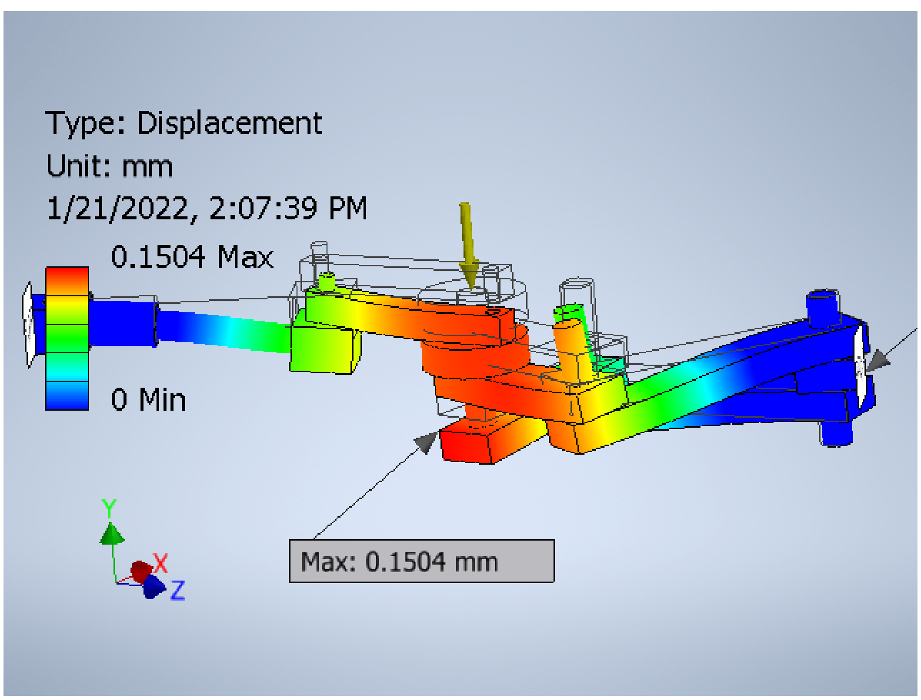

Figure 7: Displacement analysis of the manipulator showing a maximum displacement of 0.1504mm

7 Conclusion

The design for a 2-$\underline{R}$RR 1-$\underline{P}$RR manipulator has been successfully completed. Inverse geometric, forward and inverse kinematic models were determined.

The successful de-coupling of the position and orientation control of the end-effector among the different actuated joints can be observed and can be taken up further to assess this property of this mechanism for practical applications in the industry.

An optimization problem was formulated to strike a good balance between manipulator compactness and its wide operational workspace. The inverse condition numbers of forward and inverse kinematic Jacobian matrices were greater than 0.1 for a few locations and orientations, but in general, they were greater than 0.0001 throughout the operational workspace. The manipulator was modeled and its behaviour is explained in the video file attached. Stress analysis was also performed on the manipulator in order to make sure that the manipulator is capable of handling the payload, and the manipulator design has showed good strength and rigidity characteristics.

A Appendix

The following are the MATLAB main function and other functions that were used to optimize the link lengths to obtain a balance between manipulator compactness and its wide operational workspace.

1. Main optimization function:

```
clc;
clear all;

len0=130;
A=[];
b=[];
Aeq = [];
beq = [];
lb = 15;
ub = 450;
nonlcon=@mpbcCeqReturnFunction;
options = optimoptions('fmincon','Display','iter','Algorithm','sqp');
[le,fval,exitflag,output]=fmincon(@mpboptimizationObjective,len0,A,b,Aeq,
beq,lb,ub,nonlcon,options)
```

2. Function *optimizationObjective*:

```
function le = optimizationObjective(l)
le=l;
    end
```

3. Function *cCeqReturnFunction*:

```
function [c,ceq]=mpbcCeqReturnFunction(le)

a=le/3; %length OA
b=le; %length OB
h=le*2/3; %length OH
%le=1*150; %length of links
n=100; %length of end-effector

% x=-70:20:70;
% y=(-1.2*le - 70):10:(-1.2*le+70);
% phi=-30:5:30;
```

```
wcx=0;  wcy=le;     wcr=50;

x=-70:10:70;
y=(wcy - 70):10:(wcy + 70);
phi=-(pi/3):0.1:-(pi/6);

lenx=length(x);
leny=length(y);
lenphi=length(phi);

c=zeros(1,(lenx*leny*lenphi)*2+1);

score=0;
counter=1;
for i=1:lenx
    for j=1:leny
        for k=1:lenphi
            yesInside=mpbinOrOut(x(i),y(j),wcx,wcy,wcr);
            if yesInside==1
                %fprintf("here");
                ICN=mpbfindICN(x(i),y(j),phi(k),le);
%                 if abs(ICN(1))<0.1 | abs(ICN(2))<0.1
%                     score=score+1
%                 else
%
%                 end
                c(counter)=0.001-ICN(1);
                counter=counter+1;
                c(counter)=0.001-ICN(2);
                counter=counter+1;
            %else
                %fprintf("there");
            end
        end
    end
end
ceq=[];
```

4. Function *findICN*:

```
function ICN = mpbfindICN(x, y, phi, le)

a=le;
b=le;
```

```
c=2*le/3;
d=2*le/3;
e=0;
r=12;
cr=2*le/3;

d=sqrt((x*x)+(y*y));
g=((a*a)-(c*c)+(d*d))/(2*d);
h=sqrt((a*a)-(g*g));

X2=g*x/d;
Y2=g*y/d;

Cy=Y2 - h*x/d;
Dy=Cy;
Cx=X2 + h*y/d;
Dx=-Cx;


O = [0 ; 0];
A = [0 ; 0];
B = [0 ; 0];
C = [Cx ; Cy];
D = [Dx ; Dy];
E = [x ; y];
F = [0 ; 2*le];

H = [x + r*cos(phi) ; y + r*sin(phi)];
rho=2*le - (H(2)-sqrt((cr*cr) - (H(1)*H(1))));

G = [0 ; 2*le - rho];

u1Cap = (C - A)/a;
v1Cap = (E - C)/c;

u2Cap = (D - B)/b;
v2Cap = (E - D)/d;

lvCap = (H - G)/cr;
mCap = (E - H)/r;
jCap = [0 ; 1];

Em = [0 -1; 1 0];

row1=[transpose(v1Cap) 0];
```

```
row2=[transpose(v2Cap) 0];
row3=[-transpose(lvCap) r*transpose(lvCap)*Em*mCap];

forJacob = [row1 ; row2 ; row3];
invJacob = [a*transpose(v1Cap)*Em*u1Cap 0 0 ; 0 b*transpose(v2Cap)*Em*u2Cap 0 ;
        0 0 transpose(lvCap)*jCap];

if det(forJacob)==0 | det(invJacob)==0
    %fprintf("here");
    forICN=0;
    invICN=0;
else
    forICN=1/cond(forJacob);
    invICN=1/cond(invJacob);
end

ICN = [forICN ; invICN];
end
```

5. Function *inOrOut*:

```
function yesInside = mpbinOrOut(x,y,wcx,wcy,wcr)
%wcx=workspace center x
%wcy=workspace center y
%wcr=workspace circle radius

val=sqrt((x-wcx)*(x-wcx)+(y-wcy)*(y-wcy));

if val<wcr
    yesInside=1;
else
    yesInside=0;
end
end
```

Bibliography

[1] Stéphane Caro et al. "Multiobjective design optimization of 3–prr planar parallel manipulators". In: *Global Product Development*. Springer, 2011, pp. 373–383.

[2] MH Koul et al. "Synthesis and design of a 2-DOF haptic device for simulating epidural injection". In: *Proc. 13th World Congress in Mechanism and Machine Science, Guanajuato, Mexico*. 2011, pp. 1–7.

[3] Prithvi Bharadwaj Mellacheruvu et al. "Effective Education Using a 2-DOF Five-Bar Mechanism". In: *Machines, Mechanism and Robotics*. Springer, 2022, pp. 1393–1399.