Workspace Analysis of 2D planar robot using MATLAB

*Selvakumar Hastham Sathiya Satchi Sadanandam**

ABSTRACT

This is the lab report on Workspace Analysis for 2D planar robot using MATLAB considering joint limits, singularities and the presence of obstacles. The report details the conditions for singularity, collision condition, joint space and workspace analysis for the robot.

INTRODUCTION

The main aim of this report is to study the detailed analysis of the workspace environment when a 2D planar robot with link length parameters $L_1 = L_2 = 1m$ and the joint limit parameters $\theta_1 = \pm 132°$ and $\theta_2 = \pm 141°$. The presence of the obstacle is limited to one but the size and the position of the obstacle is independent.

MATHEMATICAL REPRESENTATION

In this section, the robot is evaluated for the Direct Geometric Model (DGM). Inverse Geometric Model (IGM) and Singularities.

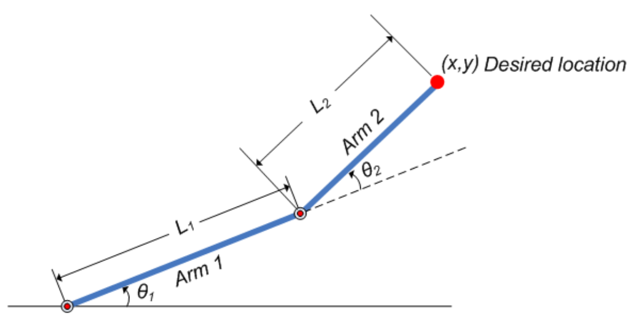

Figure 1: 2D Planar Robot - Model

Evaluation of DGM and IGM

The DGM and IGM of the robot (Fig. 1) is represented in algebraic form below where equation [1] gives the DGM of the model.

$$\begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \end{bmatrix} = \begin{bmatrix} L_1 * \cos(\theta_1) + L_2 * \cos(\theta_1 + \theta_2) \\ L_1 * \sin(\theta_1) + L_2 * \sin(\theta_1 + \theta_2) \end{bmatrix} \tag{1}$$

Assuming θ_1 and θ_2 won't consider any small angles, solving equation [1] for IGm gives,

$$\theta_2 = \frac{X^2 + Y^2 - L_1^2 - L_2^2}{2 * L_1 * L_2}$$

and θ_1 can be obtained from equation [1]. From DGM model [1], the kinematic equations can be given from differentiation as in equation [2].

$$\begin{bmatrix} \dot{\mathbf{X}} \\ \dot{\mathbf{Y}} \end{bmatrix} = \begin{bmatrix} L_1 * \sin(\theta_1) + L_2 * \sin(\theta_1 + \theta_2) & L_2 * \sin(\theta_1 + \theta_2) \\ L_1 * \cos(\theta_1) + L_2 * \cos(\theta_1 + \theta_2) & L_2 * \cos(\theta_1 + \theta_2) \end{bmatrix} * \begin{bmatrix} \dot{\theta_1} \\ \dot{\theta_2} \end{bmatrix} \tag{2}$$

where,

$$\mathbf{J} = \begin{bmatrix} L_1 * \sin(\theta_1) + L_2 * \sin(\theta_1 + \theta_2) & L_2 * \sin(\theta_1 + \theta_2) \\ L_1 * \cos(\theta_1) + L_2 * \cos(\theta_1 + \theta_2) & L_2 * \cos(\theta_1 + \theta_2) \end{bmatrix} \tag{3}$$

is the Jacobian matrix of the robot model.

Evaluation of Singularity

The singularity can be determined from the Jacobian matrix [3].

$$\begin{vmatrix} L_1 * \sin(\theta_1) + L_2 * \sin(\theta_1 + \theta_2) & L_2 * \sin(\theta_1 + \theta_2) \\ L_1 * \cos(\theta_1) + L_2 * \cos(\theta_1 + \theta_2) & L_2 * \cos(\theta_1 + \theta_2) \end{vmatrix} = 0 \tag{4}$$

$$\theta_2 = \arcsin(0) \tag{5}$$

i.e.$\theta_2 \neq 0°$ or $\theta_2 \neq 180°$. This means that link 2 of the robot attains singularity when link L_1 is parallel to L_2 and it should be avoided.

IMPLEMENTATION

The implementation of the analysis include mainly the collision detection for both links and the joint limit calculation for the first link. DGM model [1] is sufficient for this implementation. The mentioned tasks are detailed below.

Algorithm for collision detection

The implementation of the analysis is performed using trigonometry and mathematical operations which are described below. Two concepts are take into consideration for collision detection.

- Shortest distance between circle and point.
- Shortest perpendicular distance between circle and line.

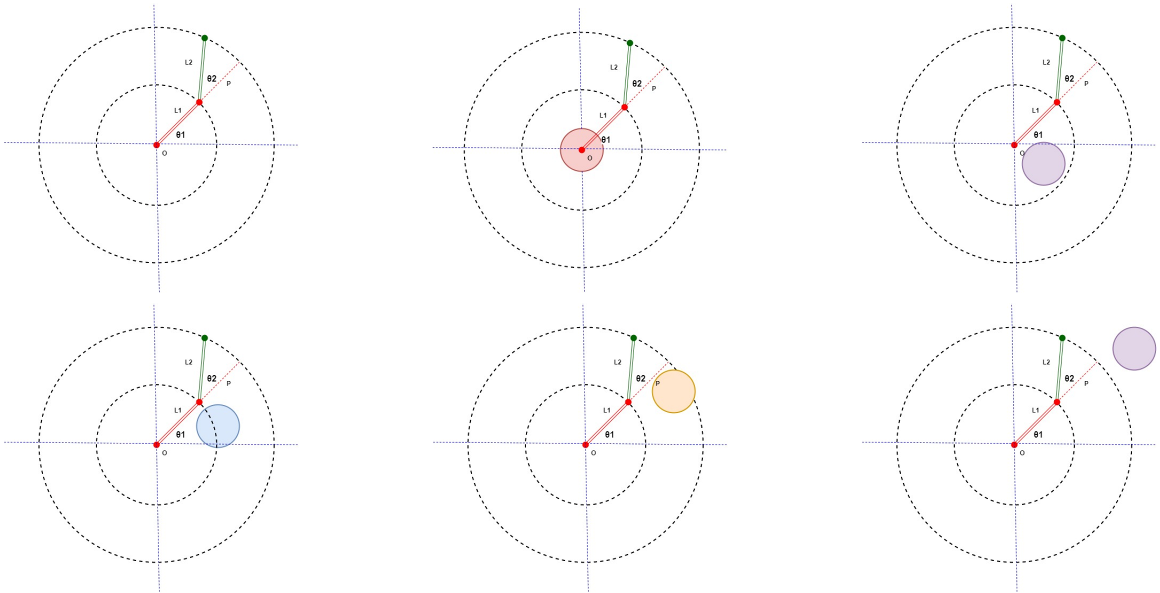

Figure 2: Collision Analysis for 2D planar robot *(order towards right)* (a) No collision state. (b) When the collision is at base frame(hypothetical). (c) Link 1 only collision or Link 1 and Link 2 both may collide. (d) Link 1 and Link 2 collision. (e) Link 2 only collision. (f)Obstacle too far away.

Algorithm

For given $L_1, L_2, \theta_1, \theta_2, x_1, y_1, x_2, y_2, x_0, y_0, , radius$
  if $\sqrt{x_0^2 + y_0^2} - radius < 0$ **then**
    Always in collision with link 1
  else if $\sqrt{x_0^2 + y_0^2} - radius > L_1 + L_2$ **then**
    Obstacle too far away
  else if $\sqrt{(x_0 - x_1)^2 + (y_0 - y_1)^2} - radius <= 0$ **then**
    Both link 1 and link2 will be collided
  else
    % For link 1
    $N_{12} = [x_1, y_1] - [0, 0]$
    $N = N_{12}/norm(N_{12})$
    $P_C = [x_0, y_0]$
    $dist = \|N_x * P_x - N_y * P_y\|$
    if $dist > radius$ **then**
      Link 1 will not be collided
    else
      Link 1 will be collided
      $count = 1$
    end if
    % For link 2
    $N_{21} = [x_2, y_2] - [x_1, y_1]$

$N = N_{21}/norm(N_{21})$
$P_C = [x_0, y_0] - [x_1, y_1]$
$dist = \|N_x * P_x - N_y * P_y\|$
if $dist > radius$ **then**
Link 2 will not be collided
else
Link 2 will be collided
$count+ = 1$
if $count > 1$ **then**
Both link 1 and link2 will be collided
end if
end if
end if

Joint Limit Calculation

When the situation of robot colliding the obstacles arises, the joint limits are calculated and eliminated from the range of the robot. For now, link L_1 will only be checked for joint limits in this case. Fig. 3 represents the link 1 being a tangent with the obstacle.

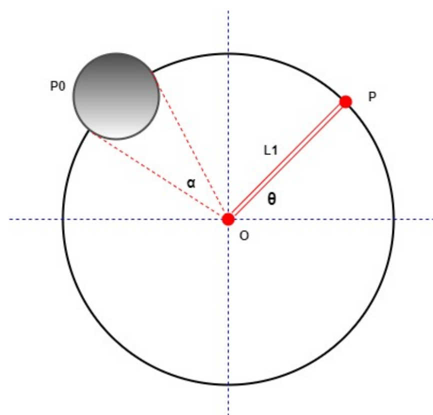

Figure 3: Joint Limit Analysis for Link L_1

Given the coordinates of the centre of the obstacle (x_0, y_0), the angle to the base frame of the robot **O** can be represented as

$$\beta = \arctan\left(\frac{y_0}{x_0}\right)$$

At collision, as shown in Fig [3], link L_1 becomes tangent to the obstacle. Then, the obstacle constraining the joint space for link L_1 is given by

$$\alpha = 2 * \|\beta - \theta_1\|$$

where β is the angle between x-axis of the robot base frame to the obstacle centre $\mathbf{P_0}$. Then, we can calculate the joint limits γ for L_1:

if $\beta == \min(\beta, \theta_1)$ **then**
  $\gamma = (\beta - \alpha/2, \theta_1)$
else
  $\gamma = (\theta_1, \beta + \alpha/2)$
end if

ANALYSIS

This section mainly aims at choosing the link configurations for the controlled workspace and joint limits. A study based on the following required configurations is performed and the parameters are chosen.

- Maximal reach of the robot is set to $2m$.
- The joint limits of the robot should not exceed $\pm 132°$ and $\pm 141°$ for joints 1 and 2 respectively.
- An obstacle is present at $(1.2m, 0.1m)$ with radius of $0.3m$.
- The robot should be able to handle an *L-shaped object* with dimensions $2m*1m*0.5m$.

For this configurations, we need to plan two tasks: Pick and Place operation and Laser cutting operation. The pick and place operation can use robot configuration with multiple solutions but Laser cutting operations need to continuous and cannot handle multiple solutions. The robot configuration will be validated for this one primary solution which will be discussed in the following sections.

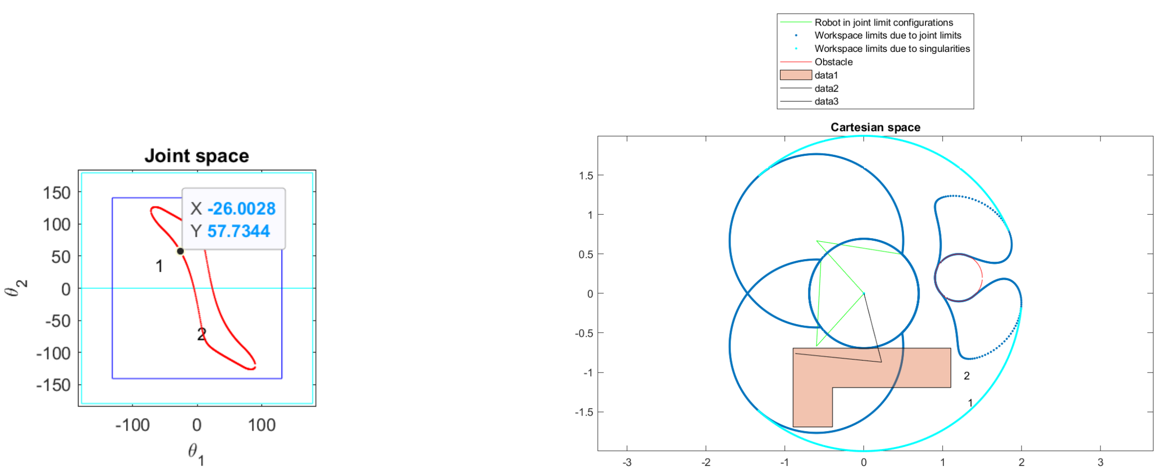

Figure 4: (a) Joint space analysis and (b) Workspace analysis for the proposed tasks.

Workspace Analysis and Joint Space Analysis

The figure [4] represent the analysis of joint space (a) and workspace (b) respectively for the specified tasks. In Fig [4](a), a closed surface (in red) denote the presence of the obstacle in θ-space.And the blue line denote the partition due to singularities. For pick and place operation, the robot should be able hold the object at its centre (definitive) while for laser cutting, the robot should track the path of the closed surface.

In Fig [4](b), the L-shaped closed surface denote the work object and red circle is the obstacle. Based on the specified tasks, the robot parameters are determined such that laser cutting has continuous path and pick and place has larger workspace. Note that pick and place operation does not require the work object entirely within the workspace. The robot parameters for both configurations considering the obstacle in $(1.2m, 0.1m)$ from the base frame with radius $0.3m$ are detailed in table [1].

Table 1: Robot Parameters for the specified tasks

Parameters	Pick and Place	Laser Cutting
Link length L_1 (in m)	0.9	0.9
Link length L_2 (in m)	1.1	1.1
Joint Limits θ_1 (in $^\circ$)	± 132	± 132
Joint Limits θ_2 (in $^\circ$)	± 141	± 141

CONCLUSION

The results for the collision detection and workspace limits are plotted in Fig. [5]. Note: The results are plotted for specific instance (for a single iteration).Thus the workspace analysis and collision detection for 2D planar robot is studied and implemented in MATLAB.

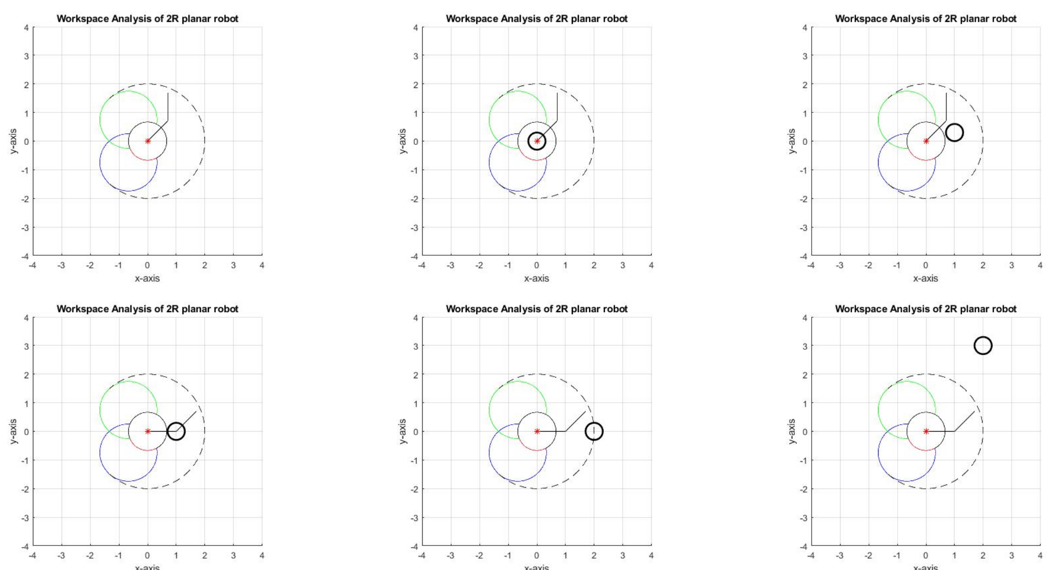

Figure 5: Results: Collision Analysis for 2D planar robot for an instance *(order towards right)* (a) No collision state. (b) When the collision is at base frame(hypothetical). (c) No collision. (d) Link 1 and Link 2 collision. (e) No Collision. (f)Obstacle too far away.