1) For the forward kinematics rotational and translation matrices were used. Parameters d1 and a2 were known as 5 and 10.

2)For the inverse kinematics we used geometry for all angles.

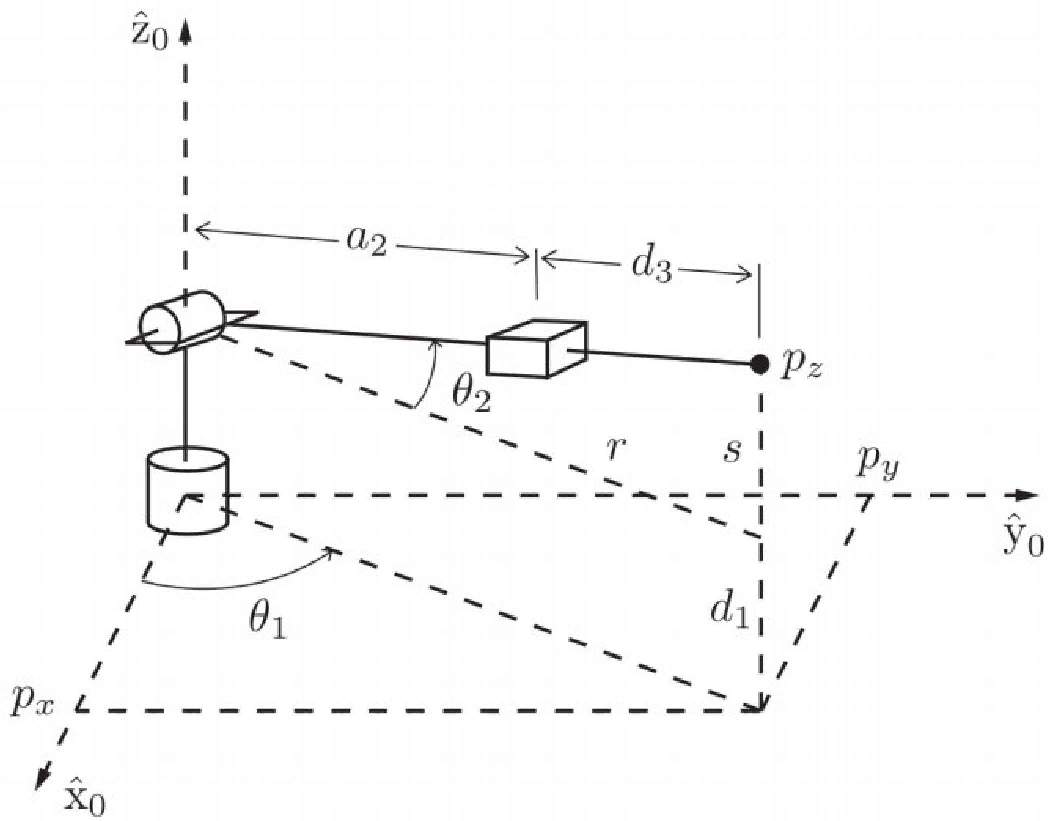

Figure 1 – RRP robot

From figure 1 it is obvious that theta1 angle is atan(px,py), but there may be another solution when theta1>pi. So theta11 solution is when first angle is less than 180 degree, while theta12 solution is when this angle bigger than 180 degree. The same fromulas for the second angle. So we have 4 different solutions (theta11,theta21) ; (theta12,theta21) ; (theta12,theta22) ; (theta11,theta22). Distance d3 was founded from geometry approach too. Also we have to add singularity if x and y coordinates of the end effector are equal to 0, there will be a lot of different solutions for theta1 angle.

3) Finding Jacobian by **geometric approach**. It is very simple, due to the fact that we have to use the formula $J_i = \begin{matrix} z_{i-1}x(o_n - o_{i-1}) \\ z_{i-1} \end{matrix}$. Where o_i is a coordinate of the joint I , z is a rotational part of the R matrix for every joint and On is a coordinate of an end-effector. So in matlab result, J_geometric is a main formula and J_geometric_initial is a numerical result of a Jacobian for given parameters. All operations provided in “main” matlab” file.
4) So, there is one singularity, when end-effector intersects z axis.

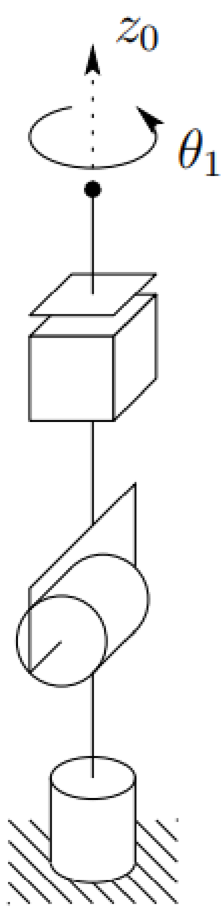

Figure 2 – Singularity

https://github.com/adilermek1/HomWork3-Robotics.git