Robotic Arm Forward & Inverse Kinematics for 2 DoF & 3 DoF

By

Mohammed Alsaggaf

Electrical Engineering Student

University of Prince Mugrin

Trainee at

Smart Methods Company

Robotics Track

July 13, 2020

Robotic Arm Control System in 2D with Servo Motors

For a robotic arm to be built we need to know how many degrees of freedom we have, the end effector's point or the joints angles, and do some robot kinematic analysis on cartesian coordinates. If we know the motors' rotating angles and we'd like to get the end effector point this is called forward kinematics. On the other hand, if we know the end effector's point and we'd like to get the motors' rotating angles we call it inverse kinematic. In this task, both of forward and inverse kinematics for 2 DoF and 3 DoF are accomplished for a robotic arm control system with servo motors.

Two Degrees of Freedom

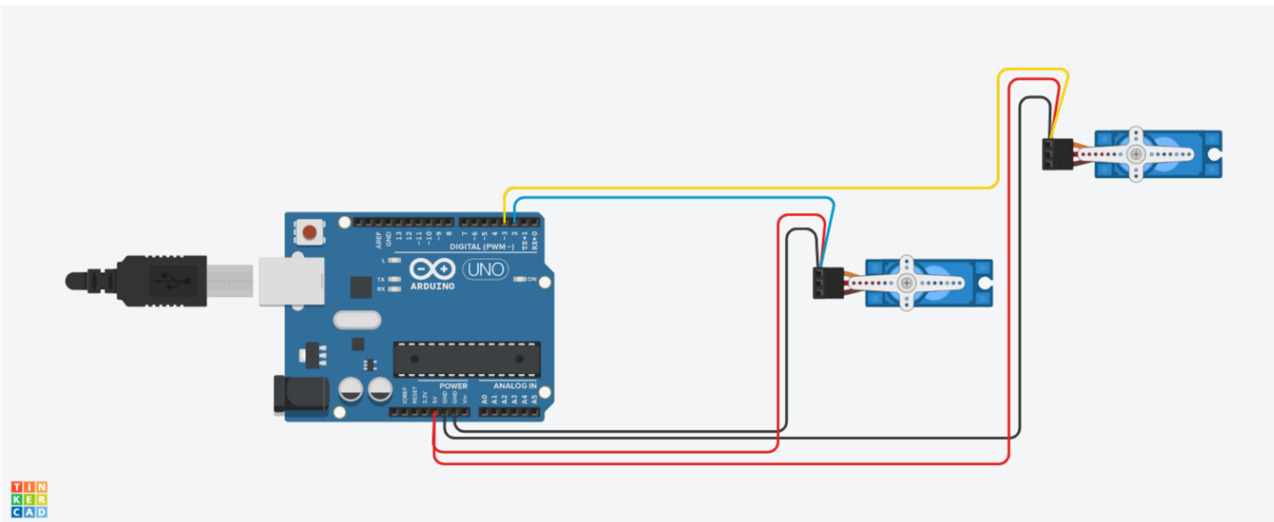

Figure 1: Circuit Diagram for 2 DoF Robotic Arm

The figure above shows the circuit diagram of the robotic arm control system which consists of an Arduino and two servo motors.

Forward Kinematics

In forward kinematics we're given the angles of rotation for each joint with the lengths to get the end effector's point in cartesian coordinates and its angle.

FORWARD KINEMATICS VARIABLES FOR 2 DOF

Given	Unknown
ϕ_1	X
ϕ_2	Y
L_1	θ
L_2	

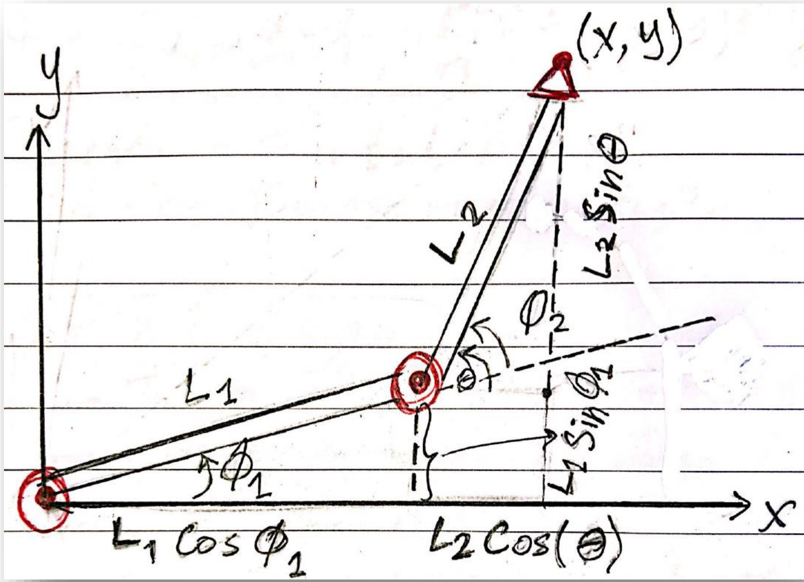

Figure 2: Forward Kinematics Sketch for 2 DoF

$$X = L_1 \cos(\phi_1) + L_2\, cos(\theta)$$
$$Y = L_1 \sin(\phi_1) + L_2\, sin(\theta)$$

$$Where\ \ \theta = \phi_1 + \phi_2$$

After getting the equations from the forward kinematics sketch, now we write the code.

```
#include <Servo.h>
Servo S1;
Servo S2;

//Given Parameters
int L1=20; //Length of the first part of the arm
int L2=30; //Length of the second part of the arm
float phy1=20; //First servo rotating angle in degrees
float phy2=50; //Second servo rotating angle in degrees
int theta=phy1+phy2; //End effector angle in degrees
float pi=3.141592654; //Pi, the mathmatical constant

float phy1Radian; //First servo rotating angle in radians
float phy2Radian; //Seond servo rotating angle in radians

//Unknown Parameters
float thetaRadian; //End effector angle in radians
float X; //End effector's X coordinate
float Y; //End effector's Y coordinate

```

Figure 3: Forward Kinematics Arduino Code Part 1

In the first part of the code as in figure 1, the two servo motors were included with the servo library and the given parameters were defined in the code with the first and second motors' rotating angles 20° and 30°, respectively, where the unknown parameters were declared as float only.

```
void setup()
{
    S1.attach(2);
    S2.attach(3);

//Converting the angles to radians
    phy1Radian=phy1*pi/180;
    phy2Radian=phy2*pi/180;

//Calculating the unkown parameters using equations
    thetaRadian=phy1Radian+phy2Radian;
    X = (L1*cos(phy1Radian))+(L2*cos(thetaRadian));
    Y = (L1*sin(phy1Radian))+(L2*sin(thetaRadian));

//Printing the end effector's X & Y coordinates
    Serial.begin(9600);
    Serial.print("X= "); Serial.println(X,DEC);
    Serial.print("Y= "); Serial.println(Y,DEC);
}

void loop()
{
  S1.write(phy1); //Rotating the first motor with the angle phy1
  S2.write(theta); //Rotating the second motor with the angle phy1+phy2
}
```

Figure 4: Forward Kinematics Arduino Code Part 2

For the second part of the code we have the void setup, where the forward kinematics equations were used to calculate the unknowns-the end effector's point and angle-, and the void loop where the code gives the order for the servo motors to rotate with the desired angles.

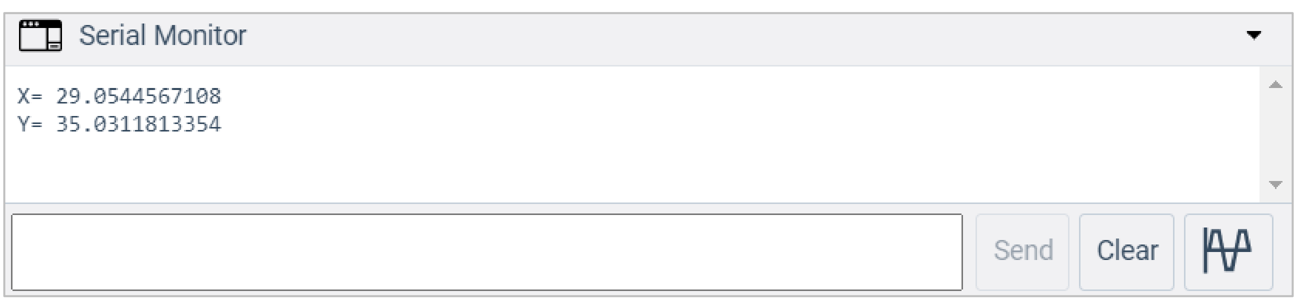

Figure 5: Serial Monitor for the Forward Kinematics

Using the Serial Monitor the end effector's coordinates were printed, as they're going to be used in inverse kinematics to justify the equations.

Inverse Kinematics

In inverse kinematics we’re given the end effector’s point with its angle and the lengths to get the angles of rotation for each joint.

INVERSE KINEMATICS VARIABLES FOR 2 DOF

Given	Unknown
X	ϕ_1
Y	ϕ_2
θ	
L_1	
L_2	

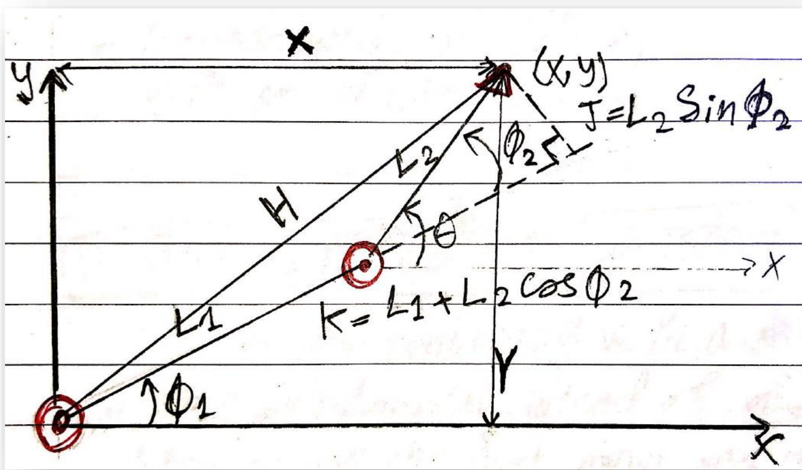

Figure 6: Inverse Kinematics Sketch for 2 DoF

Pythagorean theorem was used to solve this inverse kinematics problem.

$$H^2 = X^2 + Y^2 = J^2 + K^2$$

$$X^2 + Y^2 = (L_1\, sin(\phi_2))^2 + (L_1 + L_2\, cos(\phi_2))^2$$

$$X^2 + Y^2 = {L_2}^2\, sin^2(\phi_2) + {L_1}^2 + 2L_1L_2cos(\phi_2) + {L_2}^2cos^2(\phi_2)$$

$$X^2 + Y^2 = {L_2}^2\, [sin^2(\phi_2) + cos^2(\phi_2)] + {L_1}^2 + 2L_1L_2cos(\phi_2)$$

$$X^2 + Y^2 = {L_1}^2 + {L_2}^2 + 2L_1L_2cos(\phi_2)$$

$$\frac{X^2 + Y^2 - {L_1}^2 - {L_2}^2}{2L_1L_2} = cos(\phi_2)$$

$$\phi_2 = cos^{-1}\left(\frac{X^2 + Y^2 - {L_1}^2 - {L_2}^2}{2L_1L_2}\right)$$

$$\phi_1 = \theta - \phi_2$$

$$Where\ \ \theta = \phi_1 + \phi_2$$

After getting the equations from the inverse kinematics sketch, now we change the code.

```
#include <Servo.h>
Servo S1;
Servo S2;

//Given Parameters
int L1=20; //Length of the first part of the arm
int L2=30; //Length of the second part of the arm
int theta=70; //End effector's angle in degrees
float X=29.0544567108; //End effector's X coordinate
float Y=35.0311813354; //End effector's Y coordinate
float pi=3.141592654; //Pi, the mathmatical constant

//Unknown Parameters
float phy1; //First servo rotating angle in degrees
float phy2; //Second servo rotating angle in degrees
float phy2Radian; //Seond servo rotating angle in radians

```

Figure 7: Inverse Kinematics Arduino Code Part 1

In this part the given parameters were defined in the code with the X and Y coordinates that resulted from the forward kinematics earlier, and the unknown parameters were declared as float only.

```
void setup()
{
//Calculating the unkown parameters using equations
    phy2Radian=acos((sq(X)+sq(Y)-sq(L1)-sq(L2))/(2*L1*L2));
    phy2=(phy2Radian/pi)*180;

    phy1 = theta-phy2;

//Printing the motors' angles
    Serial.begin(9600);
    Serial.print("The First Motor's Angle = "); Serial.println(phy1);
    Serial.print("The Second Motor's Angle = "); Serial.println(phy2);
}

void loop()
{
  S1.attach(2);
  S2.attach(3);

  S1.write(phy1); //Rotating the first motor with the angle phy1
  S2.write(theta); //Rotating the second motor with the angle phy1+phy2
}
```

Figure 8: Inverse Kinematics Arduino Code Part 2

For the second part of the code we have the void setup, where the inverse kinematics equations were used to calculate the unknowns-the motors' rotating angles-, and the void loop where the code gives the order for the servo motors to rotate with the desired angles.

Figure 9: Serial Monitor for the Inverse Kinematics

As we see in the serial monitor in figure 9, the two motor's rotating angles we calculated to be 20° and 30°, respectively, as our first angles input with the forward kinematics and this justifies our calculation, code and solution.

Three Degrees of Freedom

Forward Kinematics

In forward kinematics we're given the angles of rotation for each joint with the lengths to get the end effector's point in cartesian coordinates and its angle.

FORWARD KINEMATICS VARIABLES FOR 3 DOF

Given	Unknown
ϕ_1	X
ϕ_2	Y
ϕ_3	θ_2
L_1	
L_2	
L_3	

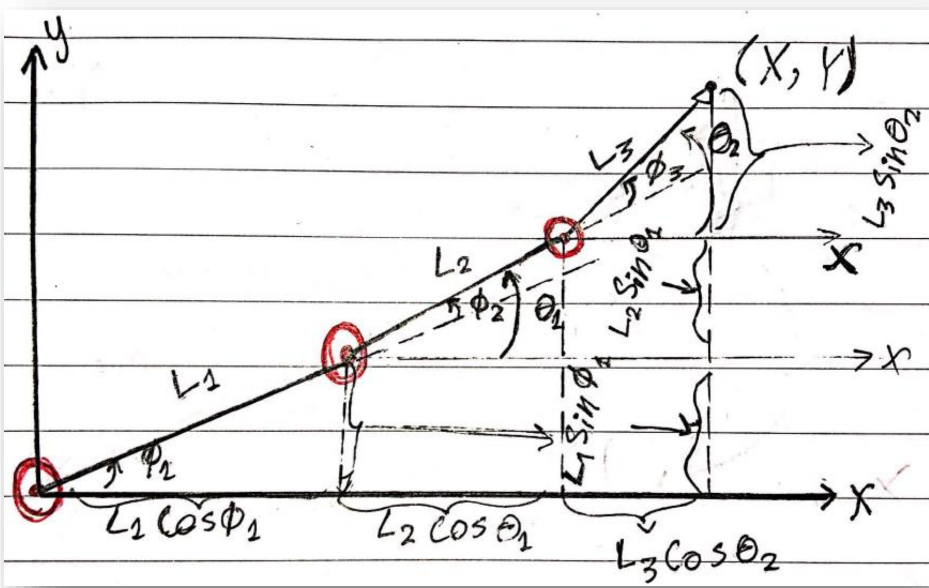

Figure 10: Forward Kinematics Sketch for 3 DoF

$$X = L_1 \cos(\phi_1) + L_2\, cos(\theta_1) + L_3\, cos(\theta_2)$$
$$Y = L_1 \sin(\phi_1) + L_2\, sin(\theta_1) + L_3\, sin(\theta_2)$$

$$Where\ \theta_1 = \phi_1 + \phi_2\ and\ \theta_2 = \phi_1 + \phi_2 + \phi_3$$

Inverse Kinematics

In inverse kinematics we're given the end effector's point with its angle and the lengths to get the angles of rotation for each joint.

INVERSE KINEMATICS VARIABLES FOR 3 DOF

Given	Unknown
X	ϕ_1
Y	ϕ_2
θ_2	ϕ_3
L_1	
L_2	
L_3	

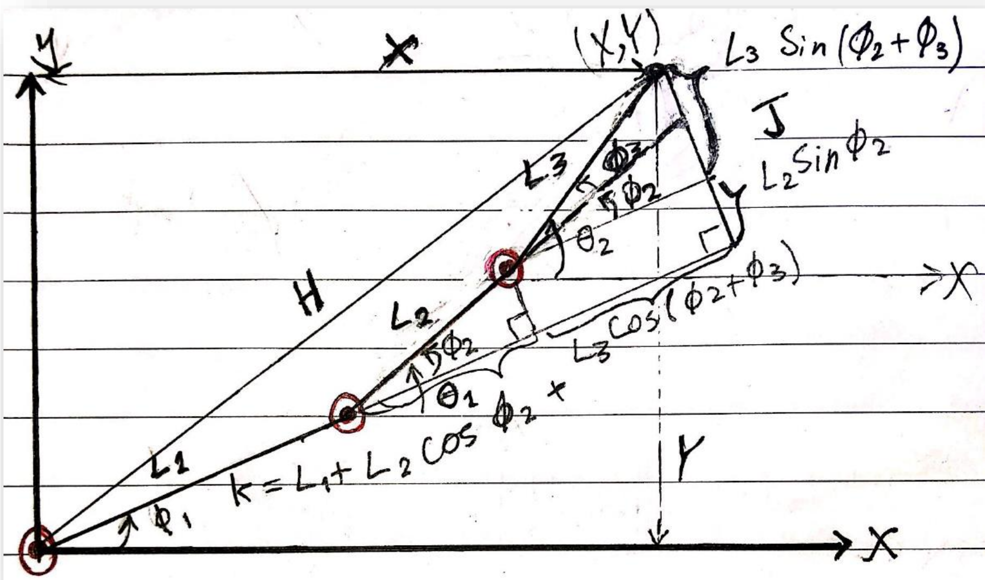

Figure 11: Inverse Kinematics Sketch 1 for 3 DoF

$$H^2 = X^2 + Y^2 = K^2 + J^2$$

$$X^2 + Y^2 = [L_1 + L_2\, cos(\phi_2) + L_3\, cos(\phi_2 + \phi_3)]^2 + [L_2\, sin(\phi_2) + L_3\, sin(\phi_2 + \phi_3)]^2$$

I tried to use this method using Pythagorean theorem to solve for the angles but turned out that it's complicated to solve. So, I found another less complicated method that I'm doing next.

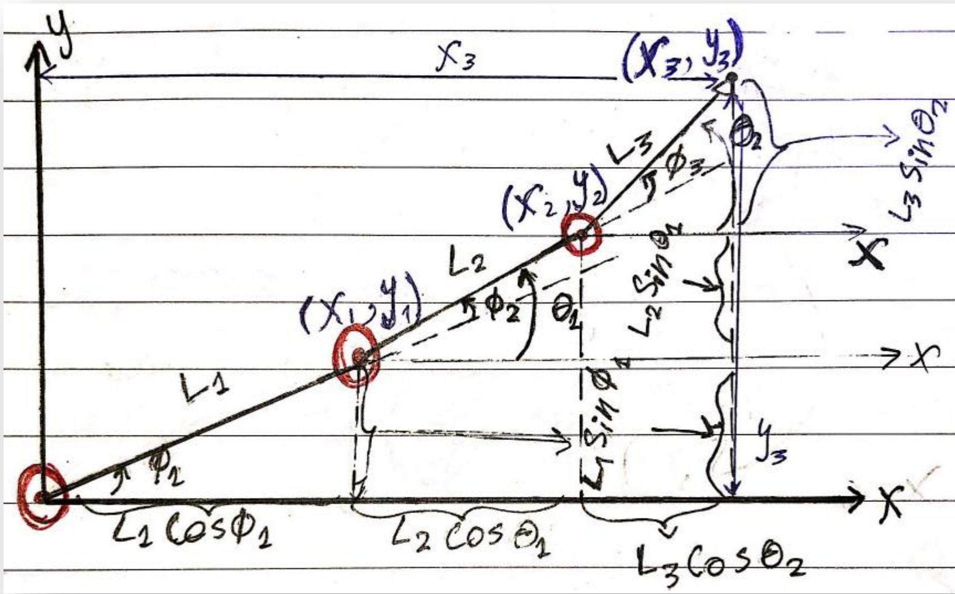

Figure 12: Inverse Kinematics Sketch 2 for 3 DoF

Firstly, the equation for the second angle ϕ_2 is calculated,

$$x_2 = x_3 - L_3\, cos(\theta_2)$$

$$y_2 = y_3 - L_3\, sin(\theta_2)$$

And by using the equation from the 2 degree of freedom inverse kinematics we get:

$$\phi_2 = cos^{-1}\left(\frac{{x_2}^2 + {y_2}^2 - {L_1}^2 - {L_2}^2}{2L_1L_2}\right)$$

$$\phi_2 = cos^{-1}\left(\frac{(x_3 - L_3\, cos(\theta_2))^2 + (y_3 - L_3\, sin(\theta_2))^2 - {L_1}^2 - {L_2}^2}{2L_1L_2}\right)$$

$$\phi_2 = cos^{-1}\left(\frac{{x_3}^2 - 2x_3L_3\, cos(\theta_2) + {L_3}^2\, cos^2(\theta_2) + {y_3}^2 - 2y_3L_3\, sin(\theta_2) + {L_3}^2\, sin^2(\theta_2) - {L_1}^2 - {L_2}^2}{2L_1L_2}\right)$$

$$\phi_2 = cos^{-1}\left(\frac{{x_3}^2 + {y_3}^2 - 2L_3(x_3\, cos(\theta_2) + y_3 sin(\theta_2)) + {L_3}^2\,(cos^2(\theta_2) + sin^2(\theta_2)) - {L_1}^2 - {L_2}^2}{2L_1L_2}\right)$$

$$\phi_2 = cos^{-1}\left(\frac{{x_3}^2 + {y_3}^2 - 2L_3(x_3\, cos(\theta_2) + y_3 sin(\theta_2)) + {L_3}^2 - {L_1}^2 - {L_2}^2}{2L_1L_2}\right)$$

$$Where\ \theta_2 = \phi_1 + \phi_2 + \phi_3$$

Then the first angle ϕ_1 is calculated by:

$$x_2 = L_1\, cos(\phi_1) + L_2\, cos(\theta_1) = L_1\, cos(\phi_1) + L_2\, cos(\phi_1 + \phi_2)$$

$$y_2 = L_1\, sin(\phi_1) - L_2\, sin(\theta_1) = L_1\, sin(\phi_1) - L_2\, sin(\phi_1 + \phi_2)$$

With using the addition trigonometric identities for sine and cosine, and using the substitution method to solve the system of equation we get:

$$cos(\phi_1) = \left(\frac{y_2}{L_2\, sin(\phi_2)} - \frac{cos(\phi_2)\,(L_1 + L_2\, cos(\phi_2))^2 - x_2}{\left(L_2\, sin(\phi_2)\right)^2}\right)$$

$$sin(\phi_1) = \frac{(cos(\phi_1))(L_1 + L_2\, cos(\phi_2)) - x_2}{L_2\, sin(\phi_2)}$$

$$sin(\phi_1) = \left[\frac{\left(\frac{y_2}{L_2\, sin(\phi_2)} - \frac{cos(\phi_2)\,(L_1 + L_2\, cos(\phi_2))^2 - x_2}{\left(L_2\, sin(\phi_2)\right)^2}\right)(L_1 + L_2\, cos(\phi_2)) - x_2}{L_2\, sin(\phi_2)}\right]$$

Now that we have $cos(\phi_1)$ & $sin(\phi_1)$ we can get ϕ_1

$$\tan\,(\phi_1) = \left(\frac{sin(\phi_1)}{cos(\phi_1)}\right)$$

$$\phi_1 = tan^{-1}\left(\frac{sin(\phi_1)}{cos(\phi_1)}\right)$$

Finally, we can calculate ϕ_3 as we have both ϕ_1 & ϕ_2 by knowing that $\phi_3 = \theta_2 - \phi_1 - \phi_2$.