Kinematics, Dynamics and Control of a Robotic Manipulator

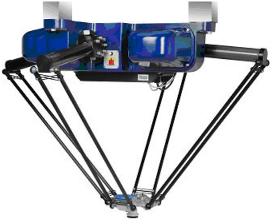
Corrado Santoro

ARSLAB - Autonomous and Robotic Systems Laboratory
Dipartimento di Matematica e Informatica - Università di Catania, Italy
santoro@dmi.unict.it

Robotic Systems

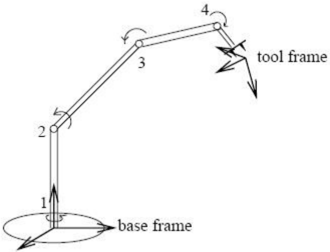
Robotic Manipulator

- A **Manipulator** is a robotic system made of a set of **arms** connected by proper **joints**
- A manipulator can be:
  - **serial** when the joints are connected in cascade and thus they depend on each other
  - **parallel** when the joints are independent
- An **end-effector** is in general attached to a manipulator; it is the tool which is required to perform the specific manipulation task (grip, mill, spray, sucker, etc.)



Spaces and Reference Frames

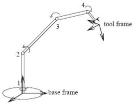
- A robotic manipulator is a rigid body that has **three** reference frames:
  - The frame which is **united with the base** (base frame), cartesian
  - The frame which is **united with the end-effector** (tool frame), cartesian
  - The frame of the **joint variables**, non-cartesian; they represent the **position of each joint** of the manipulator; in the figure, the joint variables are the **angles 1, 2, 3 e 4** that each arm forms with the next arm



Spaces and Reference Frames

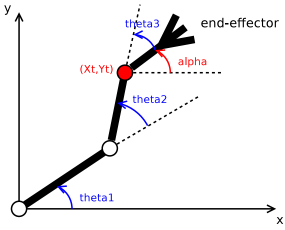
- The **joint space** is the one that we can control (we can drive the joint motors towards a certain anglular position)
- The **base space** (base frame) is the “working” frame of the manipulator
- To control a manipulator we must find the values of the **joint variables** that allow the end-effector to reach a certain position in the **base frame**
- In other words, say:
  - $\{q_1, q_2, \ldots, q_n\}$ the joint variables
  - and $\{x, y, z, \theta, \phi, \psi\}$ the **pose** of the manipulator, in terms of position $\{x, y, z\}$ of the end-effector (in the base frame) and rotation $\{\theta, \phi, \psi\}$ with respect to the base frame
- the problem is to find the transformation $T()$ such that

$$\{q_1, q_2, \ldots, q_n\} = T(x, y, z, \theta, \phi, \psi)$$



A Planar Manipulator

- The manipulator shown in figure is called **planar manipulator** because the base frame is bidimensional
- The examples show **three joints**, **two arms** and the **end-effector**
- The **pose** of the end-effector $\{X_t, Y_t, \alpha\}$ is represented by the position $\{X_t, Y_t\}$ of its joint and the angle α formed with the *x* axis
- The **joint variables** are the **angles** that each arm forms with the next $\{\theta_1, \theta_2, \theta_3\}$



Direct and Inverse Kinematics

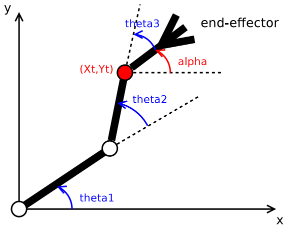
- **Direct Kinematics**: given the position in the joint space $\{\theta_1, \theta_2, \theta_3\}$, find the transformation T_D such that:

$$\{X_t, Y_t, \alpha\} = T_D(\theta_1, \theta_2, \theta_3)$$

- **Inverse Kinematics**: given the position in the base frame $\{X_t, Y_t, \alpha\}$, find the transformation T_I such that:

$$\{\theta_1, \theta_2, \theta_3\} = T_I(X_t, Y_t, \alpha)$$

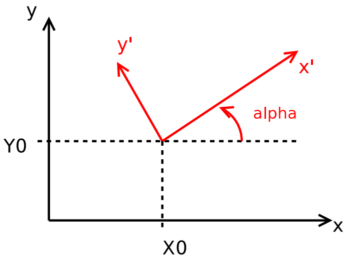
- Both problems are handled using the formulas of the **roto-translations**



Roto-translations in 2D

- Let the frame XOY and the frame $X'O'Y'$ **translated** (w.r.t. XOY) to the point (X_0, Y_0) and **rotated** (w.r.t. XOY) of an angle α
- Given a point (x', y') in $X'O'Y'$, its coordinates in XOY will be:

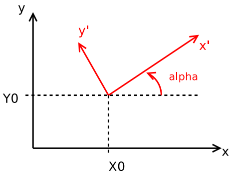
$$\begin{aligned} x &= X_0 + x' \cos\alpha - y' \sin\alpha \\ y &= Y_0 + x' \sin\alpha + y' \cos\alpha \end{aligned}$$



Roto-translations in 2D

- Let the frame *XOY* and the frame $X'O'Y'$ **translated** (w.r.t. *XOY*) to the point (X_0, Y_0) and **rotated** (w.r.t. *XOY*) of an angle α
- If we adopt **omogeneous coordinates**, we can represent the transformation using a matrix equation:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\alpha & -\sin\alpha & X_0 \\ \sin\alpha & \cos\alpha & Y_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix}$$



Roto-translations in 2D

- The matrix:

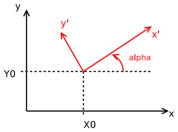
$$\begin{bmatrix} \cos\alpha & -\sin\alpha & X_0 \\ \sin\alpha & \cos\alpha & Y_0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} R & & T \\ 0 & 0 & 1 \end{bmatrix}$$

- is the **rototranslation matrix** composed of
- the **rotation matrix**:

$$R = \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix}$$

- and the **translation vector**:

$$T = \begin{bmatrix} X_0 \\ Y_0 \end{bmatrix}$$



Direct Kinematics

Direct Kinematics

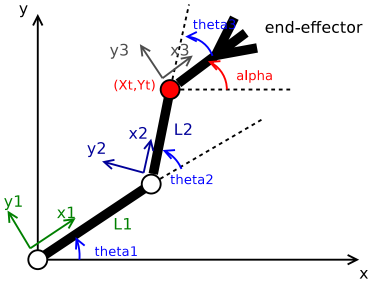
- Each **joint** implies a **rototranslation** where:
  - The **rotation** is related to the joint characteristics
  - The **translation** is given by the length of the arm
- Each joint i defines a rototranslation matrix ${}^{i-1}_{i}A$ that allows the transformation from the frame $i-1$ to the frame i
- The matrix ${}^{b}_{t}A$ that allows the transformation from the base frame to the tool frame is given by the **product** of all the joint matrices:

$$ {}^{b}_{t}A = {}^{b}_{1}A \; {}^{1}_{2}A \; {}^{2}_{3}A \; \ldots \; {}^{i-1}_{t}A $$

Direct Kinematics

- Let us consider the planar manipulator in figure, made of **three** rotation joints
- The passage from base frame to frame (x_1, y_1) is a simple rotation

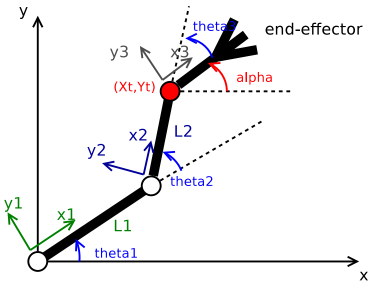
$$ {}^{b}_{1}A = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} $$



Direct Kinematics

- The transformation from (x_1, y_1) to (x_2, y_2) is based on a rotation of θ_2 and a transaltion L_1 along axis x_1 (here L_1 is the length of the first arm):

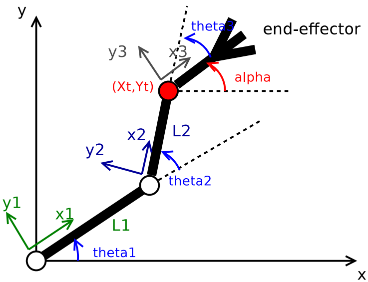
$$
{}^1_2A = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & L_1 \\ \sin\theta_2 & \cos\theta_2 & 0 \\ 0 & 0 & 1 \end{bmatrix}
$$



Direct Kinematics

- The transformation from (x_2, y_2) to (x_3, y_3) (tool frame) is based on a rotation θ_3 and a translation L_2 along axis x_2 (here L_2 is the length of the second arm):

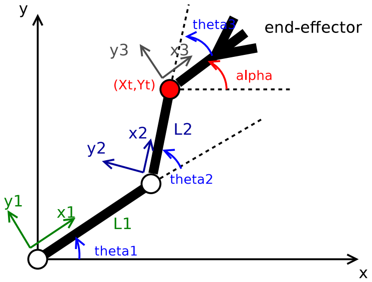
$$
{}^{2}_{t}A = \begin{bmatrix} \cos\theta_3 & -\sin\theta_3 & L_2 \\ \sin\theta_3 & \cos\theta_3 & 0 \\ 0 & 0 & 1 \end{bmatrix}
$$



Direct Kinematics

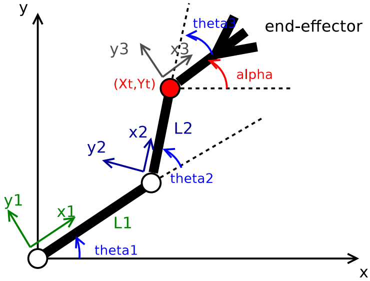
- The **complete transformation** is given by the product of the three matrices:

$$ {}^{b}_{1}A\,{}^{1}_{2}A\,{}^{2}_{t}A = \begin{bmatrix} \cos(\theta_1+\theta_2+\theta_3) & -\sin(\theta_1+\theta_2+\theta_3) & L_1\cos\theta_1 + L_2\cos(\theta_1+\theta_2) \\ \sin(\theta_1+\theta_2+\theta_3) & \cos(\theta_1+\theta_2+\theta_3) & L_1\sin\theta_1 + L_2\sin(\theta_1+\theta_2) \\ 0 & 0 & 1 \end{bmatrix} $$



- If we consider the **direct transformation** da from (x, y) to (x_3, y_3), we have:

$$
{}^{b}_{t}A = \begin{bmatrix} \cos\alpha & -\sin\alpha & X_t \\ \sin\alpha & \cos\alpha & Y_t \\ 0 & 0 & 1 \end{bmatrix}
$$



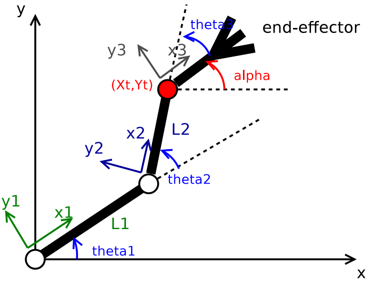
Direct Kinematics

- We have two representations:

$$ {}^b_1A\,{}^1_2A\,{}^2_tA = \begin{bmatrix} \cos(\theta_1+\theta_2+\theta_3) & -\sin(\theta_1+\theta_2+\theta_3) & L_1\cos\theta_1 + L_2\cos(\theta_1+\theta_2) \\ \sin(\theta_1+\theta_2+\theta_3) & \cos(\theta_1+\theta_2+\theta_3) & L_1\sin\theta_1 + L_2\sin(\theta_1+\theta_2) \\ 0 & 0 & 1 \end{bmatrix} $$

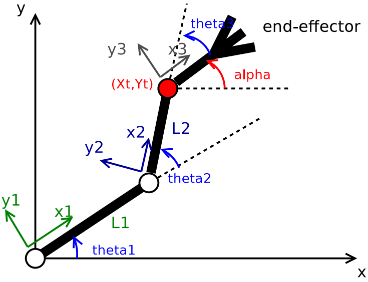
$$ {}^b_tA = \begin{bmatrix} \cos\alpha & -\sin\alpha & X_t \\ \sin\alpha & \cos\alpha & Y_t \\ 0 & 0 & 1 \end{bmatrix} $$

- We can derive the **final equations** of the direct kinematics ....



- Final equations:

$$
\begin{aligned}
X_t &= L_1 \cos\theta_1 + L_2 \cos(\theta_1 + \theta_2) \\
Y_t &= L_1 \sin\theta_1 + L_2 \sin(\theta_1 + \theta_2) \\
\alpha &= \theta_1 + \theta_2 + \theta_3
\end{aligned}
$$



Inverse Kinematics

Inverse Kinematics

- The **inverse kinematics** implies to invert the equations:

$$\begin{aligned} X_t &= L_1 \cos\theta_1 + L_2 \cos(\theta_1 + \theta_2) \\ Y_t &= L_1 \sin\theta_1 + L_2 \sin(\theta_1 + \theta_2) \\ \alpha &= \theta_1 + \theta_2 + \theta_3 \end{aligned}$$

- in order to determine θ_1, θ_2, θ_3 from X_t, Y_t, α
- .. let's assume:
  $c_1 = \cos\theta_1$, $s_1 = \sin\theta_1$, $c_{12} = \cos(\theta_1 + \theta_2)$, $s_{12} = \sin(\theta_1 + \theta_2)$

Inverse Kinematics

- We have:

$$\begin{aligned} X_t &= L_1 c_1 + L_2 c_{12} \\ Y_t &= L_1 s_1 + L_2 s_{12} \\ \alpha &= \theta_1 + \theta_2 + \theta_3 \end{aligned}$$

- Let's square the first two equations:

$$\begin{aligned} X_t^2 &= L_1^2 c_1^2 + L_2^2 c_{12}^2 + 2 L_1 L_2 c_1 c_{12} \\ Y_t^2 &= L_1^2 s_1^2 + L_2^2 s_{12}^2 + 2 L_1 L_2 s_1 s_{12} \end{aligned}$$

- Let'us sum each member:

$$X_t^2 + Y_t^2 = L_1^2 c_1^2 + L_2^2 c_{12}^2 + 2 L_1 L_2 c_1 c_{12} + L_1^2 s_1^2 + L_2^2 s_{12}^2 + 2 L_1 L_2 s_1 s_{12}$$

- given that:

$$\begin{aligned} c_1^2 + s_1^2 &= 1 \\ c_{12}^2 + s_{12}^2 &= 1 \end{aligned}$$

Inverse Kinematics

- we have:

$$X_t^2 + Y_t^2 = L_1^2 + L_2^2 + 2L_1 L_2 c_1 c_{12} + 2L_1 L_2 s_1 s_{12}$$

- Let's apply the trigonometric sum forumulas:

$$\begin{aligned} c_{12} &= \cos\theta_1 \cos\theta_2 - \sin\theta_1 \sin\theta_2 = c_1 c_2 - s_1 s_2 \\ s_{12} &= \sin\theta_1 \cos\theta_2 + \cos\theta_1 \sin\theta_2 = s_1 c_2 + c_1 s_2 \end{aligned}$$

- we have:

$$\begin{aligned} X_t^2 + Y_t^2 &= L_1^2 + L_2^2 + 2L_1 L_2 c_1 (c_1 c_2 - s_1 s_2) + 2L_1 L_2 s_1 (s_1 c_2 + c_1 s_2) \\ X_t^2 + Y_t^2 &= L_1^2 + L_2^2 + 2L_1 L_2 c_1^2 c_2 - 2L_1 L_2 c_1 s_1 s_2 + 2L_1 L_2 s_1^2 c_2 + \\ &\quad + 2L_1 L_2 s_1 c_1 s_2 \\ X_t^2 + Y_t^2 &= L_1^2 + L_2^2 + 2L_1 L_2 c_1^2 c_2 + 2L_1 L_2 s_1^2 c_2 \\ X_t^2 + Y_t^2 &= L_1^2 + L_2^2 + 2L_1 L_2 (c_1^2 + s_1^2) c_2 \\ X_t^2 + Y_t^2 &= L_1^2 + L_2^2 + 2L_1 L_2 c_2 \end{aligned}$$

Inverse Kinematics

- we have:

$$X_t^2 + Y_t^2 = L_1^2 + L_2^2 + 2L_1L_2c_2$$

- Let's solve w.r.t. c_2:

$$\cos\theta_2 = \frac{X_t^2 + Y_t^2 - L_1^2 - L_2^2}{2L_1L_2}$$

- and since:

$$\sin\theta_2 = \pm\sqrt{1 - \cos^2\theta_2}$$

- we have:

$$\theta_2 = atan2\left(\pm\sqrt{1 - \left(\frac{X_t^2 + Y_t^2 - L_1^2 - L_2^2}{2L_1L_2}\right)^2}, \frac{X_t^2 + Y_t^2 - L_1^2 - L_2^2}{2L_1L_2}\right)$$

here $atan2(a, b) = \arctan\frac{a}{b}$

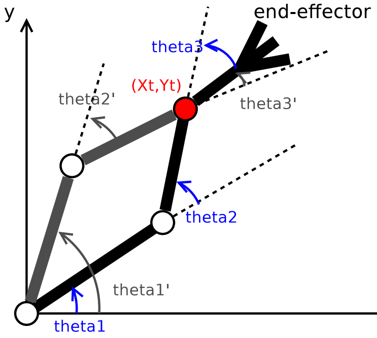
Inverse Kinematics

- The solution:

$$\theta_2 = atan2\left(\pm\sqrt{1-\left(\frac{X_t^2+Y_t^2-L_1^2-L_2^2}{2L_1L_2}\right)^2}, \frac{X_t^2+Y_t^2-L_1^2-L_2^2}{2L_1L_2}\right)$$

implies two possibile values for the angle θ_2

- Indeed, if we take a look at the figure, the **same pose** can be obtained with **two possible** configurations of $\theta_1, \theta_2, \theta_3$



Inverse Kinematics

- To obtain θ_1, let us consider:

$$\begin{aligned} X_t &= L_1 c_1 + L_2 c_{12} \\ Y_t &= L_1 s_1 + L_2 s_{12} \end{aligned}$$

- and let's apply the trigonometric sum formulas:

$$\begin{aligned} X_t &= L_1 c_1 + L_2 (c_1 c_2 - s_1 s_2) \\ Y_t &= L_1 s_1 + L_2 (s_1 c_2 + s_2 c_1) \end{aligned}$$

$$\begin{aligned} X_t &= L_1 c_1 + L_2 c_1 c_2 - L_2 s_1 s_2 \\ Y_t &= L_1 s_1 + L_2 s_1 c_2 + L_2 s_2 c_1 \end{aligned}$$

$$\begin{aligned} X_t &= (L_1 + L_2 c_2) c_1 - (L_2 s_2) s_1 \\ Y_t &= (L_1 + L_2 c_2) s_1 + (L_2 s_2) c_1 \end{aligned}$$

Inverse Kinematics

- We notice that the obtained relations:

$$
\begin{aligned}
X_t &= (L_1 + L_2 c_2) c_1 - (L_2 s_2) s_1 \\
Y_t &= (L_1 + L_2 c_2) s_1 + (L_2 s_2) c_1
\end{aligned}
$$

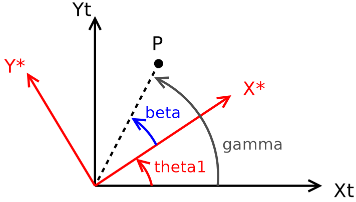
  are a rotation of the point $(L_1 + L_2 c_2, L_2 s_2)$ of an angle θ_1;
- if we consider $X^* = L_1 + L_2 c_2$, $Y^* = L_2 s_2$, we have:

$$
\begin{aligned}
X_t &= X^* c_1 - Y^* s_1 \\
Y_t &= X^* s_1 + Y^* c_1
\end{aligned}
$$

Inverse Kinematics

- to determine θ_1, let's represent graphically the model of the equations:

$$
\begin{aligned}
X_t &= X^* c_1 - Y^* s_1 \\
Y_t &= X^* s_1 + Y^* c_1
\end{aligned}
$$



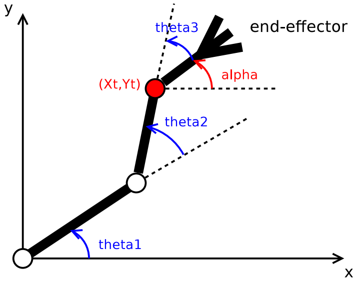
- we have:

$$\beta = atan2(Y^*, X^*) \quad \gamma = atan2(Y_t, X_t) \quad \theta_1 = \gamma - \beta$$

Inverse Kinematics

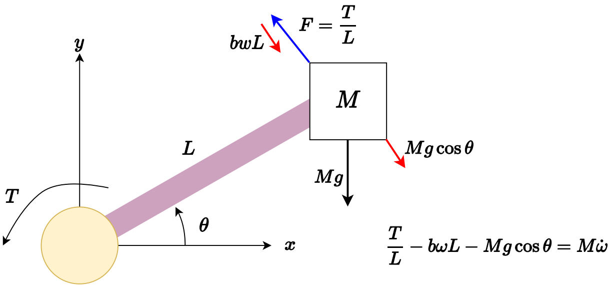
- We have the final equations:

$$
\begin{aligned}
\theta_2 &= atan2\left(\pm\sqrt{1-\left(\frac{X_t^2+Y_t^2-L_1^2-L_2^2}{2L_1L_2}\right)^2}, \frac{X_t^2+Y_t^2-L_1^2-L_2^2}{2L_1L_2}\right)\\
\theta_1 &= atan2(Y_t, X_t) - atan2(L_2\sin\theta_2, L_1+L_2\cos\theta_2)\\
\theta_3 &= \alpha-\theta_1-\theta_2
\end{aligned}
$$



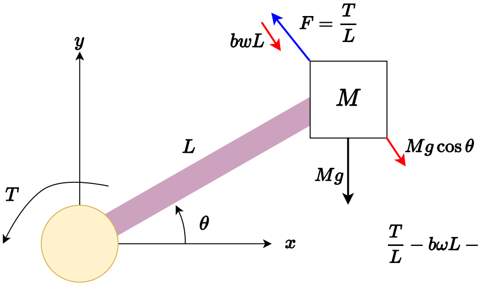
Dynamic Model

Dynamic Model



- We consider the cartesian frame ($\theta = 0$ with the arm in the horizontal position)
- The input is the **torque**
- We consider the linear speed ωL to compute the friction force

Dynamic Model

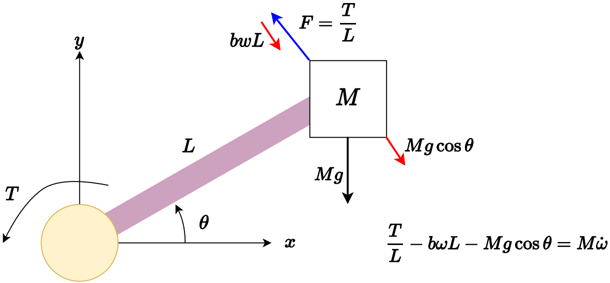


$$\frac{T}{L} - b\omega L - Mg\cos\theta = M\dot{\omega}$$

$$\dot{\omega} = -\frac{bL}{M}\omega - g\cos\theta - \frac{1}{ML}T$$

$$\dot{\theta} = \omega$$

Dynamic Model Discretized



$$
\begin{aligned}
\omega(k+1) &= (1 - \frac{bL}{M}\Delta T)\omega(k) - g\Delta T\cos\theta(k) - \frac{\Delta T}{ML}T(k) \\
\theta(k+1) &= \theta(k) + \Delta T\omega(k)
\end{aligned}
$$

Implementation

The Arm

lib/system/manipulator.py

```
class ArmElement:

    def __init__(self, _L, _M, _b):
        self.w = 0
        self.theta = 0
        self.L = _L
        self.M = _M
        self.b = _b

    def evaluate(self, delta_t, _input_torque):
        w = self.w - GRAVITY * delta_t * math.cos(self.theta) - \
            (self.b * delta_t * self.w * self. L) / self.M + \
            delta_t * _input_torque / (self.M * self.L)
        self.theta = self.theta + delta_t * self.w
        self.w = w

    def get_pose(self):
        return (self.L * math.cos(self.theta),
                self.L * math.sin(self.theta) )
```

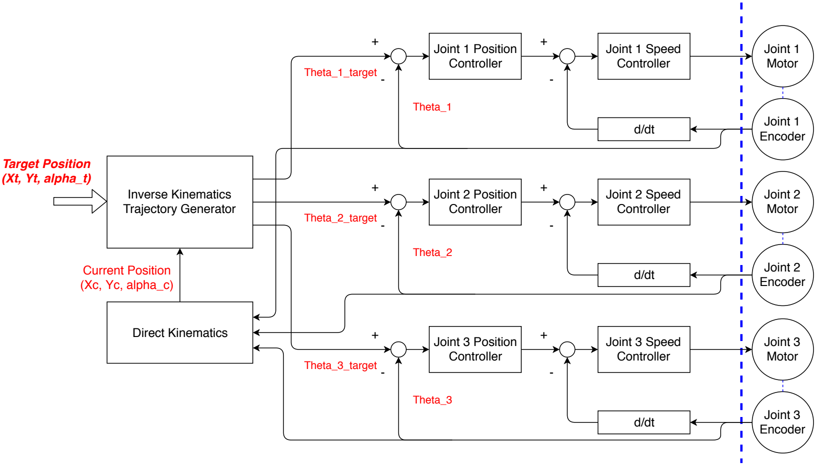
The Manipulator

lib/system/manipulator.py

```
class ThreeJointsPlanarArm:

    def __init__(self, _L1, _L2, _L3, _M2, _M3, _Mend, _b):
        self.element_1 = ArmElement(_L1, _M2 + _M3 + _Mend, _b)
        self.element_2 = ArmElement(_L2, _M3 + _Mend, _b)
        self.element_3 = ArmElement(_L3, _Mend, _b)

    def evaluate(self, delta_t, _T1, _T2, _T3):
        self.element_1.evaluate(delta_t, _T1)
        self.element_2.evaluate(delta_t, _T2)
        self.element_3.evaluate(delta_t, _T3)

    def get_pose_degrees(self):
        return ( math.degrees(self.element_1.theta),
                 math.degrees(self.element_2.theta),
                 math.degrees(self.element_3.theta) )

    def get_joint_positions(self):
        (x1, y1) = self.element_1.get_pose()
        (_x2, _y2) = self.element_2.get_pose()
        (x2, y2) = local_to_global(x1, y1,
                                   self.element_1.theta, _x2, _y2)
        alpha = self.element_1.theta + self.element_2.theta + \
                                   self.element_3.theta
        (x3, y3) = local_to_global(x2, y2, alpha, self.element_3.L, 0)
        return [ (x1, y1), (x2, y2), (x3, y3) ]
```

Manipulator Control

Manipulator Control



Kinematics, Dynamics and Control of a Robotic Manipulator

Corrado Santoro

ARSLAB - Autonomous and Robotic Systems Laboratory
Dipartimento di Matematica e Informatica - Università di Catania, Italy
santoro@dmi.unict.it

Robotic Systems