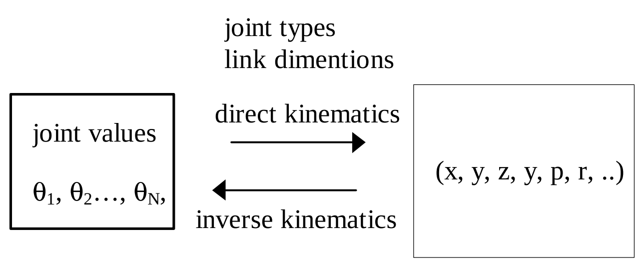
Robot Motion Analysis - Kinematics

Kinematics Analysis

We are made of bones, muscles and senses. We control using muscles and measure with senses: touch, vision, etc. Robots are built with links and joints in various configurations. Robot without intelligence can only control and measure the joints directly, such as rotate joint 1 for 300 pulses. We call this joint coordinates (you can also consider angles).

To accomplish a task in an application, we need to control the position and orientation in various coordinate systems such as world, work piece to tool. The primitive robot does not know the relationships between joint coordinates and other coordinate systems. It is very difficult to be used in applications. That separates a toy robot from an industrial robot.

In order for a robot to go to certain place at certain orientation conveniently, it is necessary to know the relationship between the joint coordinate system and some other systems, such as base or tool systems.



Coordinate systems (mostly based on ANSI R15-1)

Joint coordinate system: defined in each joint (θ_1, θ_2, … θ_N). For rotational joint, it can be angles. The direct joint parameters can also be pulses or encoder counts.
World coordinate system: A Cartesian coordinate system (X, Y, Z, A, B, Z) with arbitrary location.
Base coordinate system: A Cartesian coordinate system (X_0, Y_0, Z_0, A_0, B_0, Z_0), with its origin at the base of the robot mounting plate. For robot with rotational first joint, the z-axis is vertical coincide with first joint (waist rotation). For robot with linear first joint, the origin is at the intersection of midpoint of travel of the fist axis and mounting surface. In addition,

X_0 : from origin to projection of center point of the work envelop.
Z_0 : from origin away from base plate.

Jointed arm robot top view

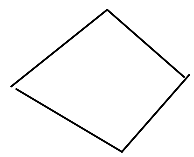
Tool interface coordinate system (X_t, Y_t, Z_t, A_t, B_t, Z_t): It is a Cartesian coordinate system defined at the tool mounting plate.

Z_t: Normal of the mounting plate or in the direction of the tool (gripper, nozzle)

X_t: normal to the surface of the gripper.

In applications, there is another important coordinate system called Tool Center Point (TCP). It translates and rotates from the tool interface coordinate system. For a welding robot, it is convenient to define the origin at the tip of the welding gun. The Z_t is in the direction of the tip. For a gripper, it is convenient to define the center of the grip. The Z_t is

The solution to inverse kinematics may not be unique. For example, a SCARA robot can reach most positions within the working envelope in two possible ways:



When you solve the inverse, there will be two solutions. When there are possible multiple solutions, industrial robots are often designed with default and can be modified.

For a robot that has more than 6 dof, the solution to the inverse kinematics can be a continuous function. Theoretically, there are infinite solutions.

Inverse kinematics is performed by an industrial robot often also. For example, you instruct the robot to move to location P_1 defined in base coordinate system using joint interpolation. In order to perform joint interpolation, a robot must know the increment for each joint in joint coordinate. Therefore, the robot must first find the joint values correspond to P_1.

$$(\theta_{11}, \theta_{21}, \ldots \theta_{N1}) = f^{-1}(x_1, y_1, z_1, y_1, p_1, r_1)$$

It then find the difference between the current point P_0 and P_1 to perform joint interpolation.

f and f^{-1} is not too complicated for a Cartesian robot because the translation can be decomposed from orientation. This is the major reason why the control for Cartesian robot is simpler, But f is rather complex for other configurations. The inverse can be even more complex. An arbitrary design may not even have a closed form solution.

Background

Right hand coordinate system

xyz: model

Matrix

Square matrix: number of rows = no. of col.

Column matrix: one col.

Row matrix: one row.

Identity (unit vector)

Orientation and unit vector

Orientation of a vector can be expressed with a unit vector. A unit vector is a vector with magnitude of “1”. It is used to indicate directions. In 3-D,

$$\sqrt{x^2 + y^2 + z^2} = 1$$

e.g.

(1, 0, 0), along x

(-.5, 0, .5) is not a unit vector

(.707, 0, -.707) is between x and -z in xz plane.

$\left(\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}\right)$ is between x, y and z.

Transpose

A’: columns and rows are exchanged.

Equality

Each element is the same.

Addition and Substraction

Association: (A + B) + C = A + (B + C)

Multiplication

No of col in the 1st = no of rows in the 2nd.
A(B + C) = AB + AC
ABC = A(BC)
AB <> BA

Inverse

Dot product

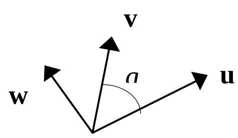
The dot product of two vectors is a scalar $\mathbf{u} * \mathbf{v}$, $\mathbf{u} = (u_x, u_y, u_z)$, $\mathbf{v} = (v_x, v_y, v_z)$

$$w = \mathbf{u} * \mathbf{v} = u_x v_x + u_y v_y + u_z v_z$$

Cross product

The cross product of two vectors is a vector

Two non-parallel vectors **u** and **v** can be moved such that their initial points coincide, $\mathbf{w} = \mathbf{u} \times \mathbf{v}$ is also a vector. **w** is in the normal defined by the plane by **u** and **v**. The magnitude is |**u**| |**v**| *sin q*.



Motion Transformation

Transformation matrix: To transform a position and orientation from one coordinate to another, including rotation and translation:

$$T = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

4 regions.

P translation

Here, **n, o, a** orientation change

6 unit vectors

n, o, a

(n_x, o_x, a_x)

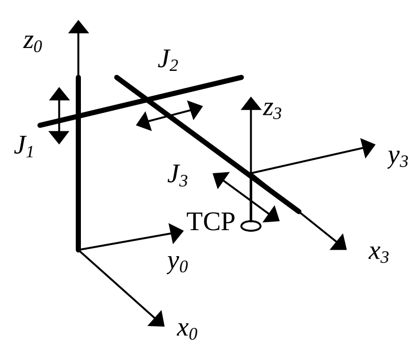
Translation

$$T_{translation} = \begin{bmatrix} 1 & 0 & 0 & a \\ 0 & 1 & 0 & b \\ 0 & 0 & 1 & c \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Rotation is identify

P = (a, b, c)

E.g. A point in the tool coordinate system is (x_N, y_N, z_N). The origin of the tool coordinate is at (a, b, c) in the base coordinate system. What is (x_0, y_0, z_0), this point in the base coordinate system?



Using simple analytical geometry

$$\mathbf{V_3} = (x_3 + a, y_3 + b, z_3 + c)'$$

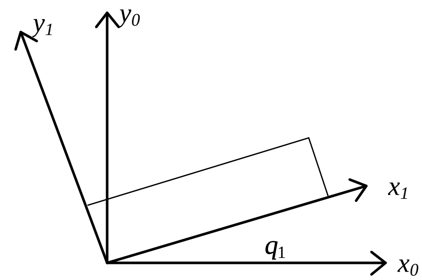
We can also use the translation transformation.

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & a \\ 0 & 1 & 0 & b \\ 0 & 0 & 1 & c \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_3 \\ y_3 \\ z_3 \\ 1 \end{bmatrix} = \begin{bmatrix} x_3 + a \\ y_3 + b \\ z_3 + c \\ 1 \end{bmatrix}$$

The advantage is not obvious, other than a fancy representation

Rotation transformation

Simple rotation about z axis. Consider the waist rotation about z axis in Rhino. It is easy to know a point in the first joint (x_1, y_1). What is its corresponding (x_0, y_0)?



$$\begin{aligned} x_0 &= x_1 \cos q_1 - y_1 \sin q_1 \\ y_0 &= x_1 \sin q_1 + y_1 \cos q_1 \\ z_0 &= z_1 \end{aligned}$$

In matrix form.

$$Rot(z,q) = \left[\begin{array}{ccc|c} \cos q & -\sin q & 0 & 0 \\ \sin q & \cos q & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

Characteristics

- first 3 * 3 is for rotation
- z does not change, the row and column for z is 1.
- 6 unit vectors
- P = 0
- If 1 → -1, z change sign, 2nd col change sign. Show with model.

Rotation for other axes

$$Rot(x,q) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos q & -\sin q & 0 \\ 0 & \sin q & \cos q & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$Rot(y,q) = \begin{bmatrix} \cos q & 0 & \sin q & 0 \\ 0 & 1 & 0 & 0 \\ -\sin q & 0 & \cos q & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Characteristics

- Translation vector.
- Rotational <u>unit</u> vectors. This is because rotation should not change the scale.
- "1" can appear in any position. That mean a variable rotation plus a change of axis. The signs can also change due to the orientation of the rotating axis.
- When two consecutive joints do not intersect, the transformation may also include translation. The rotation may also involve constant rotation of an axis. Then, "1" may appear in other parts of the matrix.

Example

CRS F3 is holding a plate. The plate is at location (0, 0, 200) in the tool coordinate system. At the ready position, the tool coordinate system is at (500, 0, 600).

- What is the location of the plate in the base coordinate system?
- If the base is rotated 30^0 about z_0, what is the location of the plate?

$$v_0 = Tran(500,0,600)Rot(y,90)v_N = \begin{bmatrix} 0 & 0 & 1 & 500 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 600 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 200 \\ 1 \end{bmatrix} = \begin{bmatrix} 700 \\ 0 \\ 600 \\ 1 \end{bmatrix}$$

$$v_0 = Rot(z,30)Tran(500,0,600)Rot(y,90)v_N$$

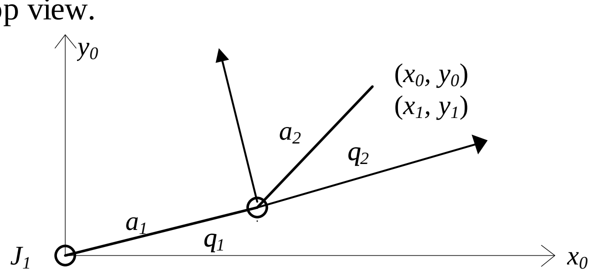
$$= \begin{bmatrix} .866 & -.5 & 0 & 0 \\ .5 & .866 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 500 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 600 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 200 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} 433+173 \\ 250+100 \\ 600 \\ 1 \end{bmatrix}$$

If you swap the order of z rotation and translation, you will have Tran(433, 250,600)Rot(z, 30)Rot(y,90). Please verify.

Example

The power lies when you deal with multiple links. Consider the first two joints in a SCARA robot. Top view.



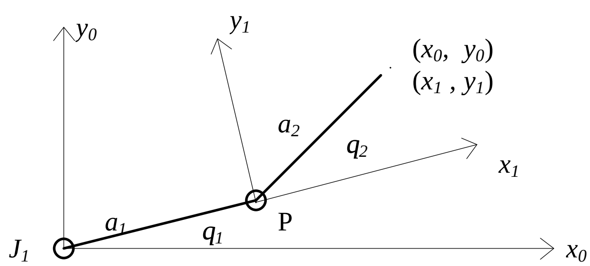
We know the length of L_1 and L_2 are fixed at a_1 and a_2. (x_0, y_0) are functions of q_1 and q_2.

$$x_0 = a_1 \cos q_1 + a_2 \cos(q_1 + q_2)$$

$$y_0 = a_1 \sin q_1 + a_2 \sin(q_1 + q_2)$$

This is rather different from a Cartesian robot. In reality we will also have another rotational, vertical and a twist joint. It could also have revolving joint. We can see that the transformations depend on the configuration.

We can use the transformation method we used earlier. Let's consider a very simple robot We can consider the link 2 being translated to point P then rotated for θ_1.



The translation to P along x_0 and y_0 are: $a_1 \cos q_1$, $a_1 \sin q_1$

The rotation is about z for θ_1.

The vector in the translated coordinate is $v_1 = (a_2 \cos q_2,\ a_2 \sin q_2,\ z,\ 1)'$ Let's use the standard transformation to calculate TCP with respect to (x_0, y_0, z_0).

$$v_0 = A_{tran} A_{rot2} v_1 = \begin{bmatrix} 1 & 0 & 0 & a_1 \cos q_1 \\ 0 & 1 & 0 & a_1 \sin q_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos q_1 & -\sin q_1 & 0 & 0 \\ \sin q_1 & \cos q_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_2 \cos q_2 \\ a_2 \sin q_2 \\ z \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} a_2 \cos q_1 \cos q_2 - a_2 \sin q_1 \sin q_2 + a_1 \cos q_1 \\ a_2 \sin q_1 \cos q_2 + a_2 \cos q_1 \sin q_2 + a_1 \sin q_{12} \\ z \\ 1 \end{bmatrix}$$

Identical, with trigonomitry identities.

If we find the transformation,

$$T = A_{tran} A_{rot} = \begin{bmatrix} \cos q_1 & -\sin q_1 & 0 & a_1 \cos q_1 \\ \sin q_1 & \cos q_1 & 0 & a_1 \sin q \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- you can combine two matrix together if translation is before rotation.
- matrix multiplication is not commutative, or, $\boldsymbol{A_1 A_2}\ {}^{1}\ \boldsymbol{A_2 A_1}$.

This means, that the z axis does not change in orientation or position

x and y changes orientation, and position.

In the first method, if we add a gripper pointing down, it might become difficult. In the 2nd method, we only need to add another transformation and multiply.

Robot kinematic analysis

An industrial robot has 4, 5, 6 or even more dof. The transformation from the base to TCP can be complex. With the transformation introduced earlier, the transformation of the entire robot can be decomposed to successive transformations between connected links. Let joint *n* has transformation A_i. The transformation of a robot with N dof has is

$$\boldsymbol{T_N = A_1 A_2 \ldots A_n}.$$

T_N defines the location and orientation of the tool coordinate system (TCP). In addition, Then, a point in the base coordinate $\mathbf{v}_0 = {}^0\boldsymbol{T}_n\,\mathbf{v}_n$.

In order to be consistent, people in the field follow certain convention in naming links and joints and define the angles. This is called Denavit-Hartenberg convention and homogeneous transformation.

Physically, what does this mean? Let's look at the example. If rotation proceeds translation, the translation should be along x axis for a_1. Let's see if they will be equal.

$$v_0 = A_{rot}A_{tran}v_1 = \begin{bmatrix} \cos q_1 & -\sin q_1 & 0 & 0 \\ \sin q_1 & \cos q_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_2\cos q_2 \\ a_2\sin q_2 \\ z \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} a_2\cos q_1\cos q_2 - a_2\sin q_1\sin q_2 + a_1\cos q_1 \\ a_2\sin q_1\cos q_2 + a_2\cos q_1\sin q_2 + a_1\sin q_{12} \\ z \\ 1 \end{bmatrix}$$

Identical. Also check the unit vectors.

$$v_{n-2} = A_{tran}A_{rot}v_n = \begin{bmatrix} 1 & & & 0 \\ & 1 & & 6 \\ & & 1 & 0 \\ & & & 1 \end{bmatrix} \left(\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos 30 & -\sin 30 & 0 \\ 0 & \sin 30 & \cos 30 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 3 \\ 7 \\ 5 \\ 1 \end{bmatrix} \right) = \begin{bmatrix} 3 \\ 8.348 \\ 7{,}83 \\ 1 \end{bmatrix}$$

Most robots have 4, 5, 6 or more joints. The location and orientation of the end effector becomes cumbersome if found directly. People in the field developed conventions and step-by-step method to cope with the complexity.

Steps:

1. Finding transformation between link n-1 and n, A_n,
2. Finding the transformation of the robot, ${}^0T_N$.
3. $v_0 = {}^0T_N v_N$, where ${}^0T_N = A_1 A_2 \ldots A_N$, For TCP, $v_N = (0, 0, 0, 1)$, v_0 is called Cartesian coordinate in Rhino.

What is the purpose?

Robots often have to find the relationships between its tool coordinate system and its base coordinate system.

Homogeneous transformation and Denavit-Hartenberg convention

An n dof robot has N mobile links linked by N joints. A link may be translated, rotated with respect to its previous link through their joint. Two joint coordinate system for 2 consecutive joints can be setup (x_{n-1}, y_{n-1}, z_{n-1}) and (x_n, y_n, z_n). We allow 4 out of 6 independent motions,

1. Rotate q about z_{n-1}
2. Translate d along z_{n-1}
3. Translate a along x_n
4. Rotate t about x_n

Then, the transformation from (x_{n-1}, y_{n-1}, z_{n-1}) to (x_n, y_n, z_n) will be

$$A = \mathrm{Rot}(z_{n-1}, \theta)\ \mathrm{Tran}(z_{n-1}, d)\ \mathrm{Tran}(x_n, a)\ \mathrm{Rot}(x_n, t)$$

After multiplication, we have

$$A = \begin{bmatrix} \cos q & -\sin q \cot t & \sin q \sin t & a \cos q \\ \sin q & \cos q \cos t & -\cos q \sin t & a \sin q \\ 0 & \sin t & \cos t & d \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

In case that a mechanism can not be modeled with this method because the convention allows only 4 motions, one can add an intermediate coordinate system to handle.

CONSTRUCTION PROCEDURE (transparency).

1. Number the links and joints, starting from the base. The stationary base is link 0. Link n rotates and/or translates with respect to link n-1 through joint n. The last joint is N.
2. Definition of coordinate systems for each link.
   - z_{n-1}: axis of motion of joint n. If joint n is rotational, link n rotates about z_{n-1}. If joint n is linear, link n translates along z_{n-1}.
   - x_n: along common perpendicular ($z_{n-1} \rightarrow z_n$) perpendicular and passing both.
3. Define the joint parameters.
   - q_n is the rotation from x_{n-1} to x_n about z_{n-1}.
   - d_n is the translation along z_{n-1}.
   - a_n: the translation along x_n.
   - t_n is the rotation from z_{n-1} to z_n about x_n.
4. Form the homogeneous displacement matrix A_n.

Example: A 1 dof sliding joint mechanism. This is a translation about z_0 with variable $\underline{d_1}$, and a constant -90^0 rotation about x_0 (t_1).

1. Find the transformation.
2. A point in the 2nd coordinate system is at (2, 0, 5). What are the values in 1st coordinate system. The slide moved 10 units up.

1.

Link	*a*	*t*	*d*	*q*
1	0	-90	d_1	$\underline{0}$

$$T = A_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

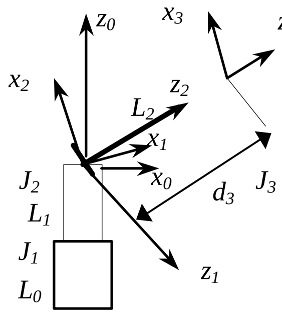
We can check on unit vectors and right hand rule.

Let's see if this makes sense. If we have a point in the 1st link coordinate system (0, 0, 1), it will be.

$$= \qquad = \begin{bmatrix} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 10 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 0 \\ 5 \\ 1 \end{bmatrix} = \begin{bmatrix} 2 \\ 5 \\ 10 \\ 1 \end{bmatrix}$$

Example:

The first 3 axes of a spherical robot,



Step 1. Number the links and joints

Step 2. Set up joint coordinate systems

Step 3. Find link parameters.

Link	a	t	d	q
1	0	90	0	<u>q_1</u>
2	0	90	0	<u>q_2</u>
3	0*	0*	<u>d_3</u>	0 (const)

t_1 differ in sign from book. The only effects are different x orientation and different q values. Otherwise, you get the same result. This way, q is more intuitive.

Step 4. Find As. Short hands: c_n is cosq_n, s_n is sinq_n,

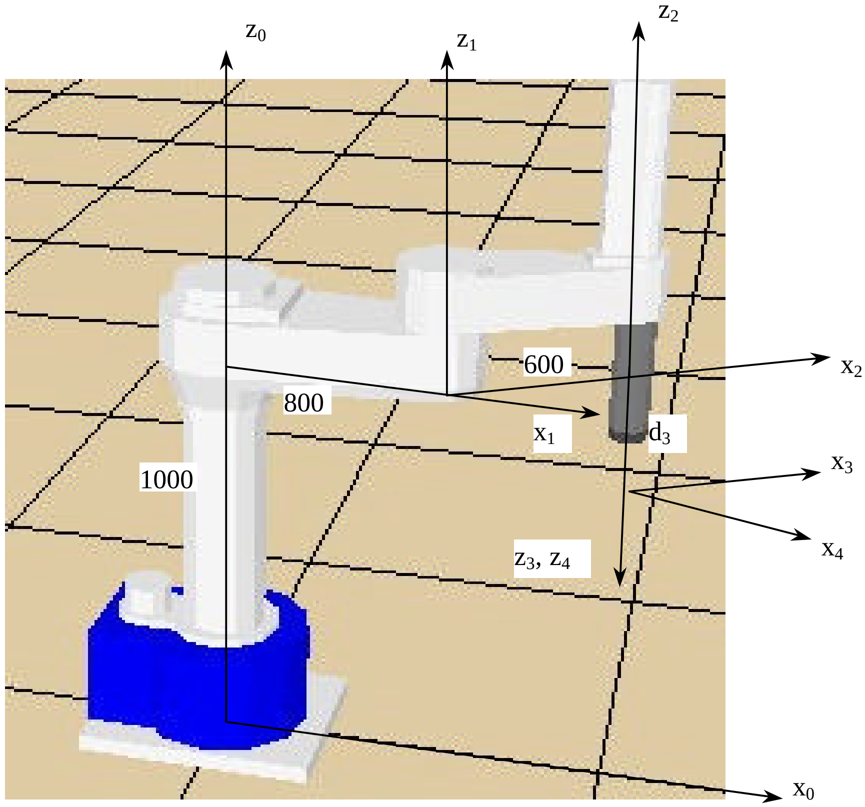
$$A_1 = \begin{bmatrix} c_1 & -s_1c(t_1) & s_1s(t_1) & a_1c_1 \\ s_1 & c_1 & -c_1s(t_1) & a_1s_1 \\ 0 & s(t_1) & c(t_1) & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_1 & 0 & s_1 & 0 \\ s_1 & 0 & -c_1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

y_1 becomes z_0, it makes sense that they are both vertically up.

$$A_2 = \begin{bmatrix} c_2 & -s_2c(t_2) & s_2s(t_2) & a_2c_2 \\ s_2 & c_2 & -c_2s(t_2) & a_{\backslash 2}s_2 \\ 0 & s(t_2) & c(t_2) & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_2 & 0 & s_2 & 0 \\ s_2 & 0 & -c_2 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

E.g. Adept 1 robot. SCARA, 4 axis. Please find the transformation. x_1 and x_2 are at the same height of 1000 mm. The shoulder link is 800 mm long while the elbow link is 600 mm long.



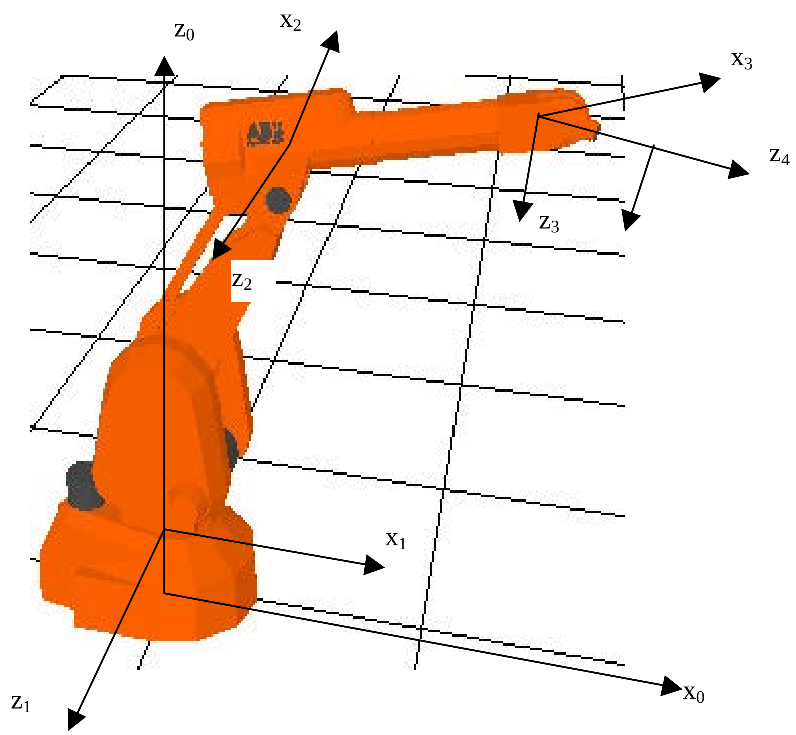
Solution:

1. label the links
2. 2. Coordinate system as indicated
3. parameters

	A	T	D	θ
1	800	0	1000	θ_1
2	600	0	0	θ_2
3	0	180	d_3	0
4	0	0	0	θ_4

4. The matrices.

E.g. Jointed arm robot with pitch (Let's simplify by not cosiderting first roll and last roll). The height of the shoulder is 400 mm. The length of the shoulder link is 1200mm. The length of the elbow link is 1000mm.



5. label the links
6. 2. Coordinate system as indicated
7. parameters

	A	T	D	θ
1	0	90	300	θ_1
2	1200	0	0	θ_2
3	1000	0	0	θ_4
4	0	90	0	θ_4

8. The matrices.

Transformation for the robot, forward kinematics

The beauty of this method for a robot that has N joints, the location and orientation of tool (TCP) in base coordinate system is defined by

$$\boldsymbol{{}^{0}T_{N} = A_{1}A_{2} \ldots A_{N}}$$

In the polar robot example, we have

$$
{}^{0}T_{3} = A_{1}A_{2}A_{3}\begin{bmatrix} c_1 & 0 & s_1 & 0 \\ s_1 & 0 & -c_1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} c_2 & 0 & s_2 & d_3 s_2 \\ s_2 & 0 & -c_2 & d_3 c_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_1 c_2 & s_1 & c_1 s_2 & d_3 c_1 s_2 \\ s_1 c_2 & -c_1 & s_1 s_2 & d_3 s_1 s_2 \\ s_2 & 0 & -c_2 & -d_3 c_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

Forward (direct) kinematic analysis

For each robot, you can find T_N with joint parameters as constants and variables embedded in the matrix. For any robot stop, all joint variables are known to the robot. Substitute in the variables in the transformation, you can find tool (TCP)

- Location
- Orientation

The meaning of A and T

A_n defines the location and orientation of the (x_n, y_n, z_n) in terms of (x_{n-1}, y_{n-1}, z_{n-1}).

If we write A_n in form

$$A_n = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Where

p vector defines the location of the origin in n^th^ coordinate system

a is a unit vector defines direction of z_n,

o is a unit vector defines direction of y_n

n is a unit vector defines x_n

E.g. Please fill in the missing parameters in the following matrix and indicate TCP.

$$\begin{bmatrix} 1 & A & B & 5 \\ C & .707 & D & -3 \\ E & -.707 & F & 0 \\ G & H & I & J \end{bmatrix}$$

A = B = C = E = … = 0. J = 1

x_n = (1, 0, 0)

y_n = (0, .707, -.707)

z_n = (0, .707, .707)

E.g. When $\theta_1 = 0$, we have

$$A_1 = \begin{bmatrix} c_1 & 0 & s_1 & 0 \\ s_1 & 0 & -c_1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The location of the second coordinate system in the base coordinate system is (0, 0, 0)

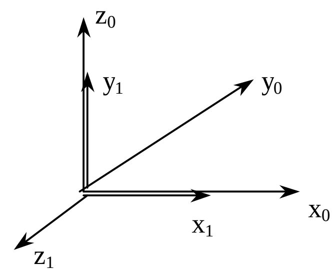
Orientation

$z_1 \to -y_0$

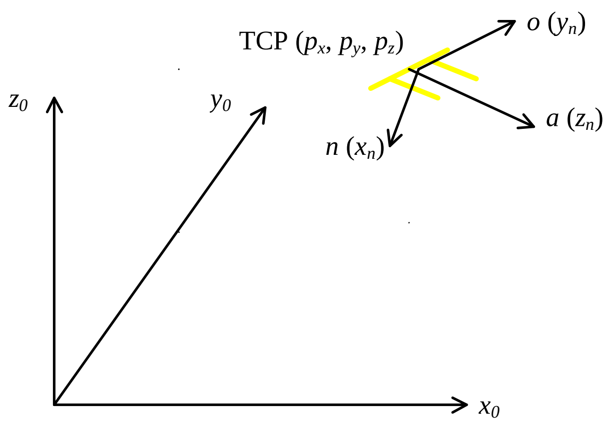
$y_1 \to z_0$

$x_1 \to x_0$

Let's see if this is correct.



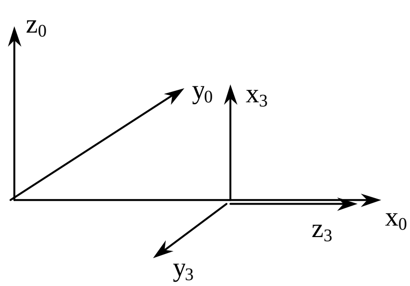
This is right and both are right hand coordinate system. Please draw when $\theta_1 = 90$.



E.g. For the simple polar robot, let $\theta_1 = 0$, $\theta_2 = 90$, $d_3 = 10$.

$$
{}^0T_3 = \begin{bmatrix} c_1c_2 & s_1 & c_1s_2 & d_3c_1s_2 \\ s_1c_2 & -c_1 & s_1s_2 & d_3s_1s_2 \\ s_2 & 0 & -c_2 & -d_3c_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 10 \\ 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

This matrix says that the tool is at (10, 0, 0), pointing (z) to x_0, Its x is up.



E.g. For the polar robot, what is TCP if $q_1 = 90^0$, $q_2 = 30^0$ and and $d_3 = 5$”?

$$
{}^0T_3 = \begin{bmatrix} c_1c_2 & s_1 & c_1s_2 & d_3c_1s_2 \\ s_1c_2 & -c_1 & s_1s_2 & d_3s_1s_2 \\ s_2 & 0 & -c_2 & -d_3c_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ .866 & 0 & .5 & 2.5 \\ .5 & 0 & -.866 & -4.33 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

That means, the TCP is at (0, 2.5, -4.33), The gripper is pointing down&front. The gripper opens along y, so it is along x axis.

Forward kinematics can also be used to relate positions in tool coordinate and base coordinate.

$$\mathbf{v}_{\boldsymbol{0}} = {}^{\boldsymbol{0}}\boldsymbol{T}_{\boldsymbol{N}}\, \mathbf{v}_{\boldsymbol{N}}$$

For Rhino:

v_0 are called cartesian coordinates.

v_N is called tool coordinates (0, 0, 0) is TCP.

Joint coordinates are B (roll), C (pitch), ...

E.g. If a point in the scope link coordinate is (0, 0, 1), where is TCP and what is its orientation if $q_1 = 90^0$, $q_2 = 30^0$ and $d_3 = 5$"?

$$v_0 = {}^0T_3 v_3 = \begin{bmatrix} c_1c_2 & s_1 & c_1s_2 & d_3c_1s_2 \\ s_1c_2 & -c_1 & s_1s_2 & d_3s_1s_2 \\ s_2 & 0 & -c_2 & -d_3c_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 1 & 0 & 0 \\ .866 & 0 & .5 & 2.5 \\ .5 & 0 & -.866 & -5*.866 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 3 \\ -5.196 \\ 1 \end{bmatrix}$$

This means, a point (0, 0, 1) in the gripper coordinate is at (0, 3, -5.196) in base coordinate.

Inverse Kinematics

Given: position and orientation of the tool

Find: joint values

Useful in: offset, Cartesian incremental movement, linear interpolation, etc.

$$^{0}T_{N}(\text{var}\,iables) = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

If we know the location and orientation, then we know the values of of p, a, o and n. this matrix is known. We can setup up to 12 simultaneous equations.

Example

For the polar robot, If we want the robot TCP to go to (0, 5, 3)

$$^{0}T_{3} = \begin{bmatrix} c_1c_2 & s_1 & c_1s_2 & d_3c_1s_2 \\ s_1c_2 & -c_1 & s_1s_2 & d_3s_1s_2 \\ s_2 & 0 & -c_2 & -d_3c_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_1c_2 & s_1 & c_1s_2 & 0 \\ s_1c_2 & -c_1 & s_1s_2 & 5 \\ s_2 & 0 & -c_2 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

ps are fixed. The orientation is fixed at any given location. Effectively, we can only solve for 3 joint values.

$$\begin{aligned} d_3 \cos q_1 \sin q_2 &= 0 \\ d_3 \sin q_1 \sin q_2 &= 5 \\ -d_3 \cos q_2 &= 3 \end{aligned}$$

3 equations and 3 unknowns.

In 3, d_3 can not be zero.

In 1, $q_1 = 90^0$ or $q_2 = 0^0$,

In 2, $q_2 \neq 0^0$. Therefore, $q_1 = 90^0$.

In 2 and 3,

$d_3 \sin q_2 = 5$

$- d_3 \cos q_2 = 3$

$q_2 = \tan^{-1} (5 / (-3) = - 59^0$ using calculator

It should be $180 - 59 = 121^0$

$d_3 = 5 / \sin 121 = 5.833$

$$T^3 = \begin{bmatrix} c_1c_2 & s_1 & c_1s_2 & 0 \\ s_1c_2 & -c_1 & s_1s_2 & 5 \\ s_2 & 0 & -c_2 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -.515 & 0 & .857 & 5 \\ 0.857 & 0 & .515 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Check right hand rule: Check.

This is a trivial case with 3 dof. See handout. In this case, the solution will be unique. Often, the solution is not unique.

For a 6 dof robot, the forward requires 10 transcendental calls, 30 multiplies, and 12 additions. The inverse 13 transcendental calls, 27 multiplies, and 20 additions after manipulations.

Inverse of transformation matrix

In kinematics analysis, you often needs to find an inverse. If

$$T = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}, \text{ then its inverse is } T^{-1} = \begin{bmatrix} n_x & n_y & n_z & -p\cdot n \\ o_x & o_y & o_z & -p\cdot o \\ a_x & a_y & a_z & -p\cdot a \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Where the last column are dot products of vectors.

e.g.

$$A_1 = \begin{bmatrix} c_1 & 0 & s_1 & 0 \\ s_1 & 0 & -c_1 & 0 \\ 0 & 1 & 0 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \text{ and } A_1^{-1} = \begin{bmatrix} c_1 & s_1 & 0 & 0 \\ 0 & 0 & -1 & l_1 \\ s_1 & -c_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Robot Motion and Transformation

When we discuss the motion types, I "claimed" that joint interpolation is simple with smooth path while continuous path control is complex.

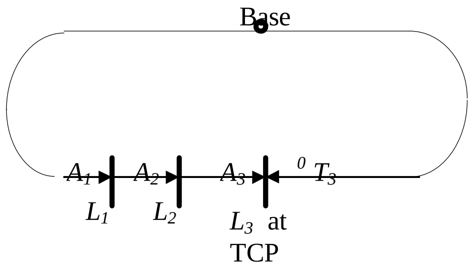
E.g. The polar robot is taught two points: P_1 and P_2. The positions in joint coordinates are (30, 90, 5) and (210, 120, 10). In a section of an application, the robot is at P_1. We instruct the robot to go to P_2 in

- Joint interplation
- Linear interpolation.

Please explain the robot controller function if you use move. The default speeds of the robot in joint coordinates are (180, 120, 10).
In Joint interpolation, the changes in joint coordinates are (180, 30, 5). Using the analysis from before, we find:

Transformation graph

You can express the transformation of a robot in two ways from the base. A transform graph is a directed graph to connect consecutive links of a robot. For the example spherical robot.



We can write the transformation from the base to TCP as $^0\boldsymbol{T}_3$ or $\boldsymbol{A}_1\boldsymbol{A}_2\boldsymbol{A}_3$, and we have

$$^0\boldsymbol{T}_3 = \boldsymbol{A}_1\boldsymbol{A}_2\boldsymbol{A}_3,$$

As a matter of fact, we can write this equality between any two points.

- in the direction of the arrow, use the transformation,
- against the direction, use the inverse.

Let's choose between L_1 and TCP.

$$\boldsymbol{A}_2\,\boldsymbol{A}_3 = \boldsymbol{A}_1^{-1}\,{}^0\boldsymbol{T}_3$$

This is useful in setting different equations. Solving inverse kinematics is often difficult. Some may be better than others. Matrix inversion.

Comprehensive example

The 3 dof polar example is not very useful. Let's add a pitch to it. This can be used for vertical insertion of round pins. Or moving column materials.

Look at the modified axis in blue. The link parameters will be now

Link	*a*	*t*	*D*	*q*
1	0	90	0	$\underline{q_1}$
2	0	90	0	$\underline{q_2}$
	0	90	$\underline{d_3}$	–180
3	0	90	0	$\underline{q_4}$

$$^0T_4 = \begin{bmatrix} -c_1c_2c_4 + c_1s_2s_4 & s_1 & -c_1s_2s_4 - c_1s_2c_4 & d_3c_1s_2 \\ -s_1c_2c_4 + s_1s_2s_4 & -c_1 & -s_1c_2s_4 - s_1s_2c_4 & d_3s_1s_2 \\ -s_2c_4 - c_2s_4 & 0 & -s_2s_4 + c_2c_4 & -d_3c_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The location portion does not change. Because pitch does not affect location the way we assign the coordinates. Rotational portion becomes more complex. You can verify that the vectors in rotational portion are unit vectors.

Forward kinematics.

If $\theta_1 = 90$, $\theta_2 = 90$, $d_3 = 10$, and $\theta_4 = 90$. Location and orientation = ?

$$^0T_4 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 10 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The gripper is at (0, 10, 0), on y_0 axis. Gripper is pointing down. Opens about x axis.

Inverse

We thought that this robot can perform vertical insertion of round pins. What does that mean in this matrix ????????????

The tool coordinate must be able to go down. The approach vector must be (0, 0, -1).

That is we must find solution

Which has solution anywhere in the envelop.

$$-sq_2sq_4 + cq_2cq_4 = -1$$
$$c(q_2 + q_4) = -1, \quad \text{or}$$
$$q_2 + q_4 = 180$$

We know that any position can be reached within envelop by varying θ_1, θ_2, and d_3. By simply choosing $\theta_4 = 180 - \theta_2$, We can maintain vertically down.

How about insertion in x-z plane????

We need approach vector to be in y, or (0, 1, 0). Or,

$$-s_1(c_2s_4 + s_2c_4) = 1$$

In order for this to be true, we must have first and second term be 1 or –1.

That can only be achieved at certain positions.

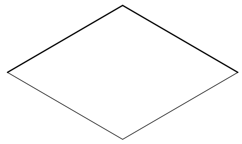
Other kinematics issues

1. Euclidean speed. For the popular articulated robot, there is no direct control of the speed of the end effector, which is important in almost all applications. Speed is the first derivative of position w.r.t the time. Jacobian.
2. Euclidean acceleration. 2^{nd} derivative.
3. Singularity.
4. Trajectory control.
5. Solution uniqueness. Define posture.

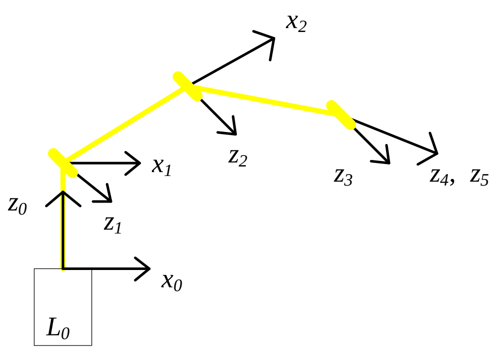
Summary of kinematics analysis

Commercial robots are not necessary built with application coordinates.
It is necessary to find relationships between the joint coordinates and application coordinates.
Forward: known
Inverse:
Robots has 4 - 7 dof. Analysis can be difficult.
Homogeneous transformation make it simple
Finding robot transformation
Forward application: Given joint, find TCP, v_0
Inverse:
Other issues
Usefulness

E.g. in an RR robot,



Example 2: The SCORBOT or Rhino robot.



z’s

x’s

Parameters

Summary of link parameters.

Link	*a*	*t*	*d*	*q*
1	0	90	l_1	q_1
2	L_2	0	0	q_2
3	L_3	0	0	q_3
4	0	90	0	q_4
5	0	0	0	q_5