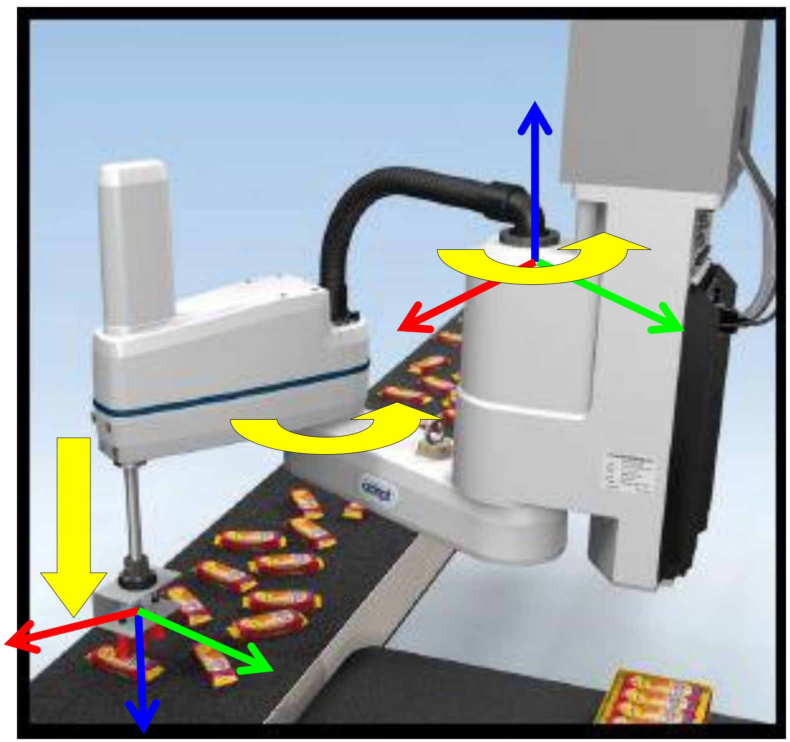
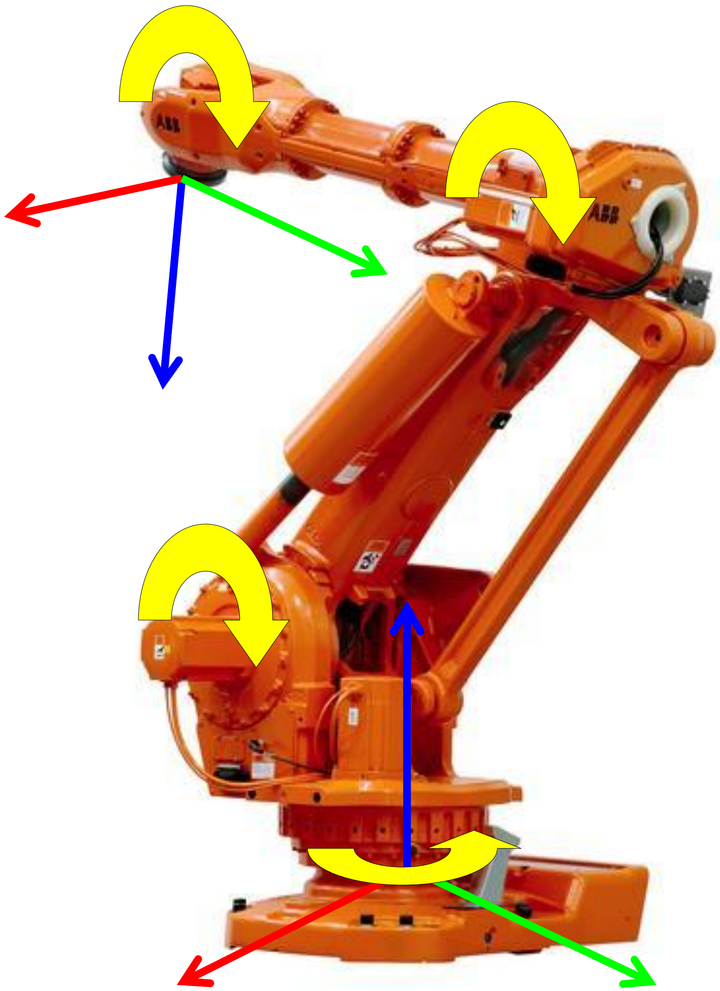
Robot Kinematics

Robot Manipulators

- A robot manipulator is typically moved through its joints
  - Revolute: rotate about an axis
  - Prismatic: translate along an axis

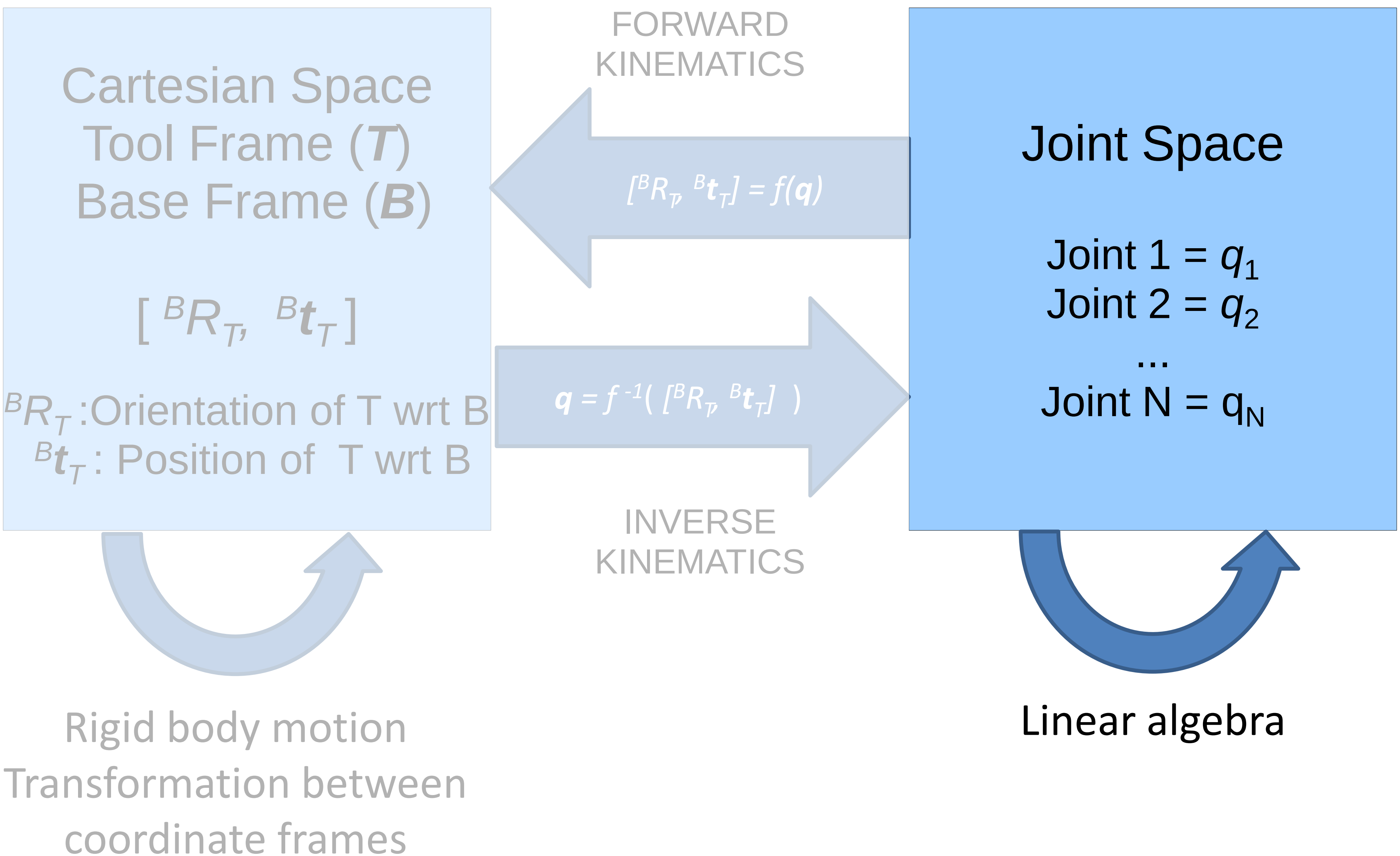


SCARA



6 axes robot arm

Kinematics


Cartesian Space
Tool Frame (***T***)
Base Frame (***B***)
$[\,{}^{B}R_{T},\ {}^{B}\mathbf{t}_{T}\,]$
${}^{B}R_{T}$:Orientation of T wrt B
${}^{B}\mathbf{t}_{T}$: Position of T wrt B
FORWARD KINEMATICS
$[{}^{B}R_{T},\ {}^{B}\mathbf{t}_{T}] = f(\mathbf{q})$
$\mathbf{q} = f^{-1}(\ [{}^{B}R_{T},\ {}^{B}\mathbf{t}_{T}]\ )$
INVERSE KINEMATICS
Joint Space
Joint 1 = q_1
Joint 2 = q_2
...
Joint N = q_N
Rigid body motion
Transformation between coordinate frames
Linear algebra

Transformation Within Joint Space

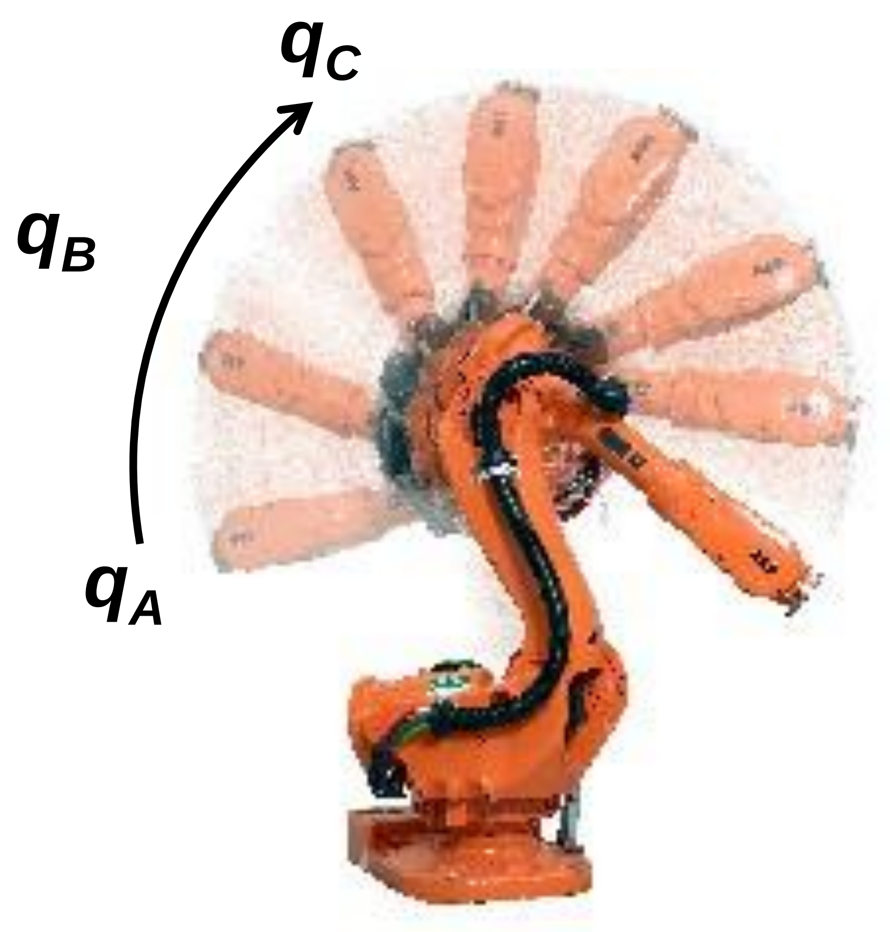
- Joint spaces are typically defined in $\boldsymbol{R}^n$

  Thus for a vector

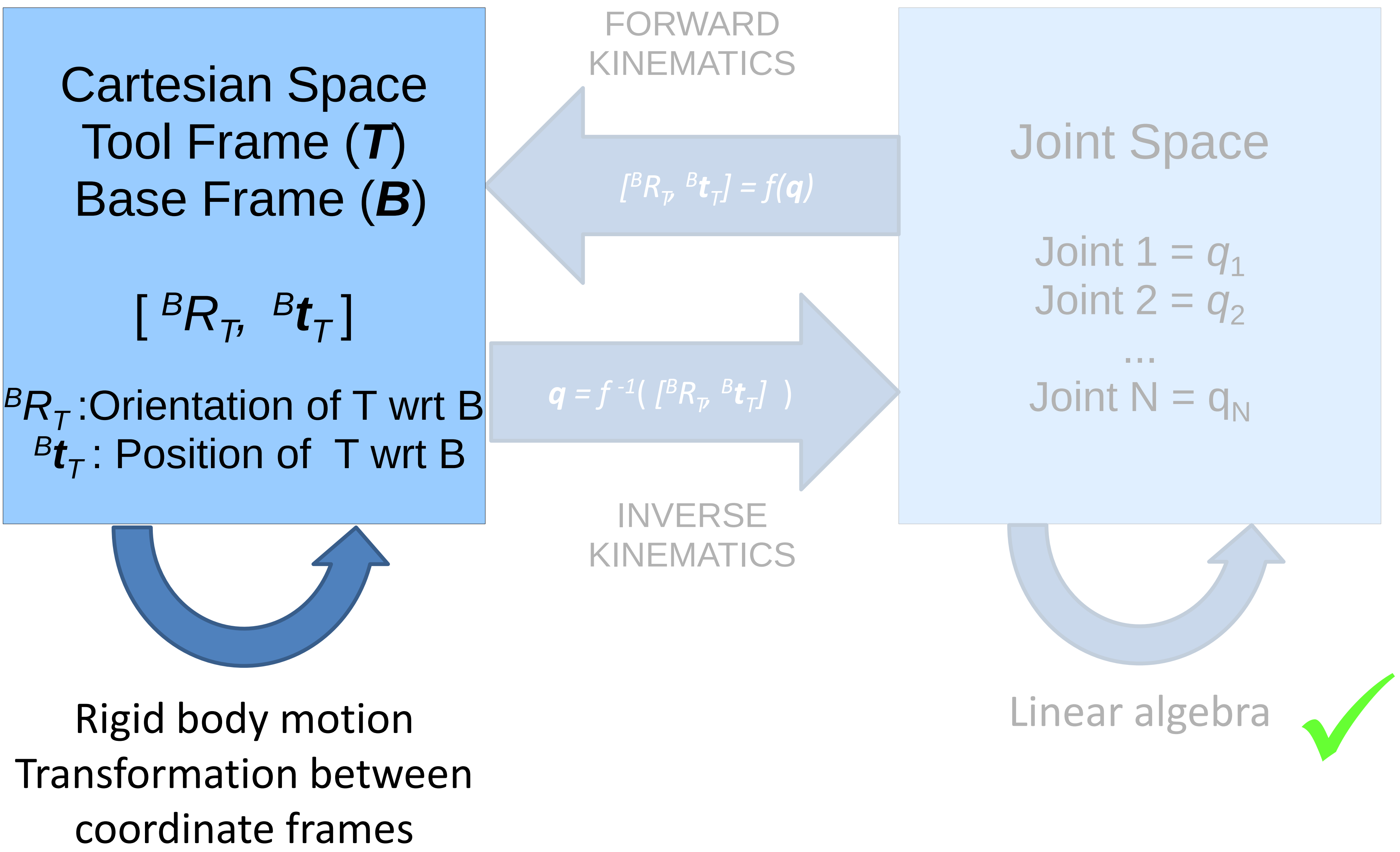
$$\mathbf{q} = \begin{bmatrix} q_1 & \dots & q_n \end{bmatrix}$$

  we can use additions subtractions

$$\mathbf{q}_c = \mathbf{q}_a + \mathbf{q}_b$$



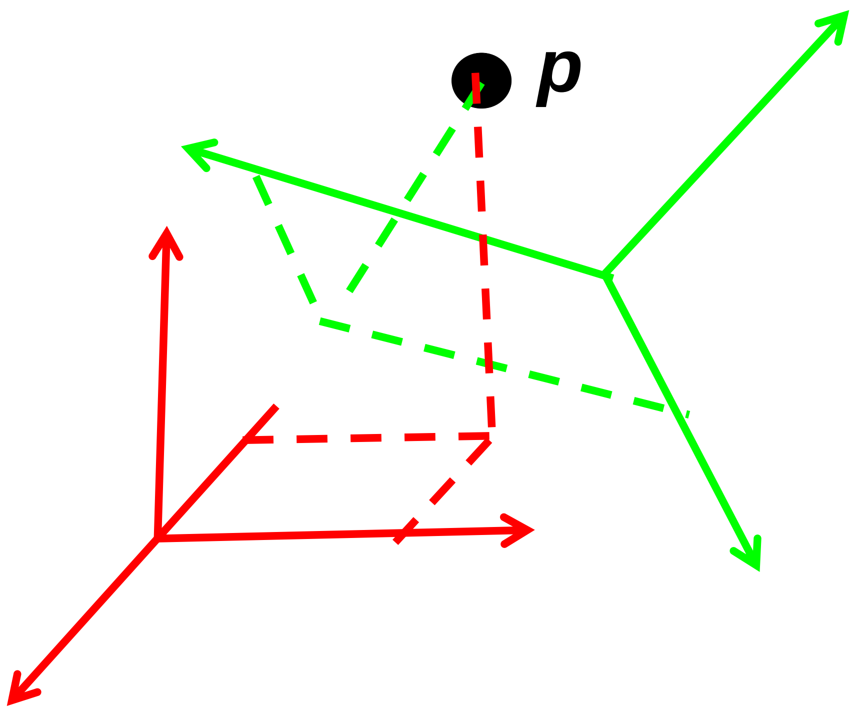
Kinematics


Cartesian Space
Tool Frame (*T*)
Base Frame (*B*)
[$^{B}R_{T}$, $^{B}\mathbf{t}_{T}$]
$^{B}R_{T}$:Orientation of T wrt B
$^{B}\mathbf{t}_{T}$: Position of T wrt B
FORWARD KINEMATICS
$[^{B}R_{T}, ^{B}\mathbf{t}_{T}] = f(\mathbf{q})$
$\mathbf{q} = f^{-1}([^{B}R_{T}, ^{B}\mathbf{t}_{T}])$
INVERSE KINEMATICS
Joint Space
Joint 1 = q_1
Joint 2 = q_2
...
Joint N = q_N
Rigid body motion
Transformation between coordinate frames
Linear algebra

Cartesian Transformation
Position and Orientation

- Combine position and orientation:
  - Special Euclidean Group: *SE(3)*

$$SE(3) = \{(\mathbf{t}, R) : \mathbf{t} \in \mathbb{R}^3, R \in SO(3)\} = \mathbb{R}^3 \times SO(3)$$



$${}^{A}\boldsymbol{p} = {}^{A}R_{B}\,{}^{B}\boldsymbol{p} + {}^{A}\mathbf{t}_{B}$$

$${}^{A}\boldsymbol{p} = {}^{A}E_{B}\,{}^{B}\boldsymbol{p}$$

•3D Rotations

- LOTS of different ways of representing them:
  - Quaternion, Euler angles, axis/angle, Rodrigues
- ONE concept

  A 3x3 rotation matrix that

$$^{A}\mathbf{p} = {}^{A}R_{B}\,{}^{B}\mathbf{p}$$

  Where $({}^{A}R_{B}{}^{T})\,{}^{A}R_{B} = {}^{A}R_{B}\,({}^{A}R_{B}{}^{T}) = I$

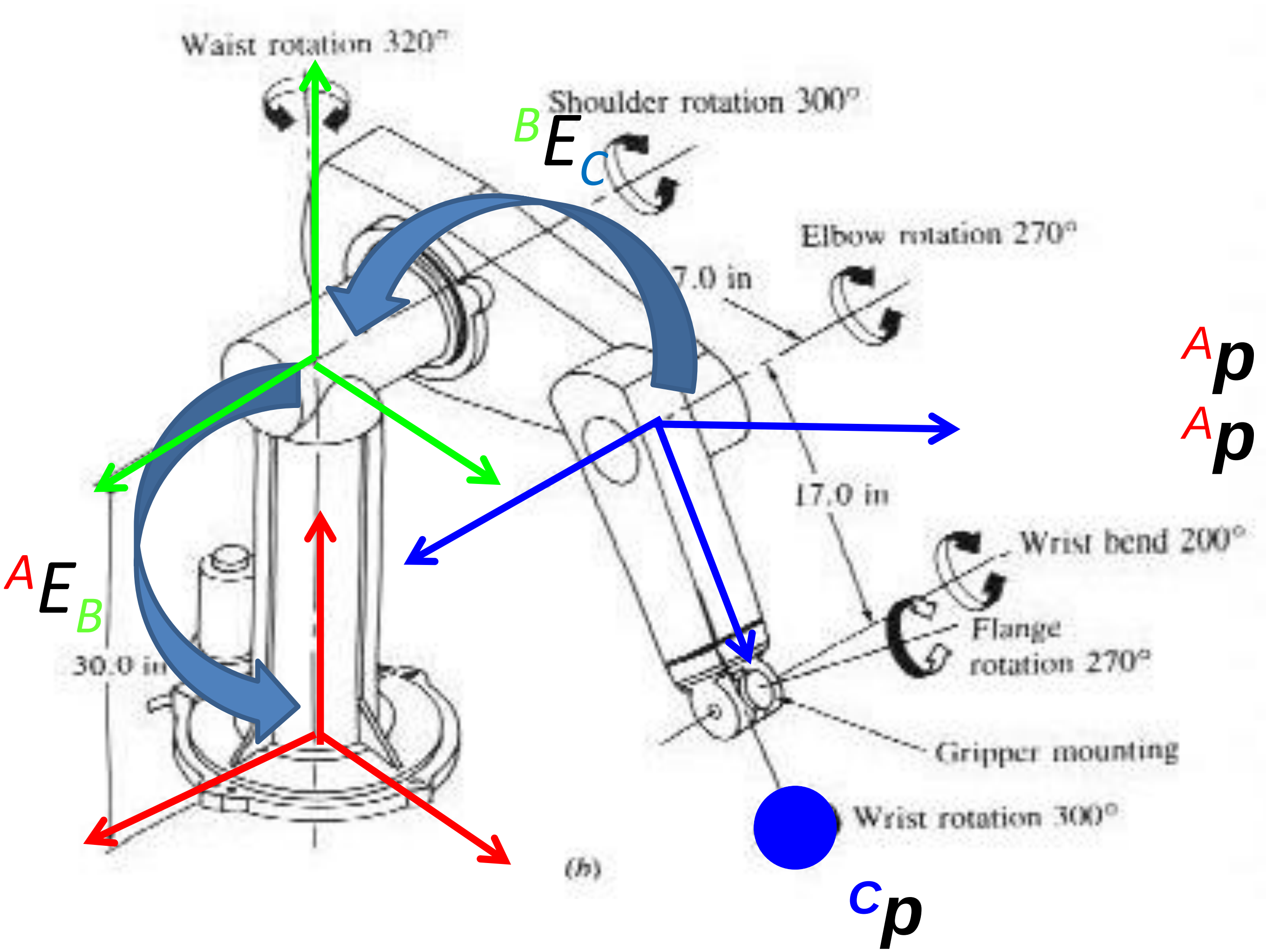
- Elementary rotations

  Rotation about x *Rotation about y* *Rotation about z*

$$R_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\varphi \\ 0 & \sin\varphi & \cos\varphi \end{bmatrix} \quad R_y(\beta) = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \quad R_z(\alpha) = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

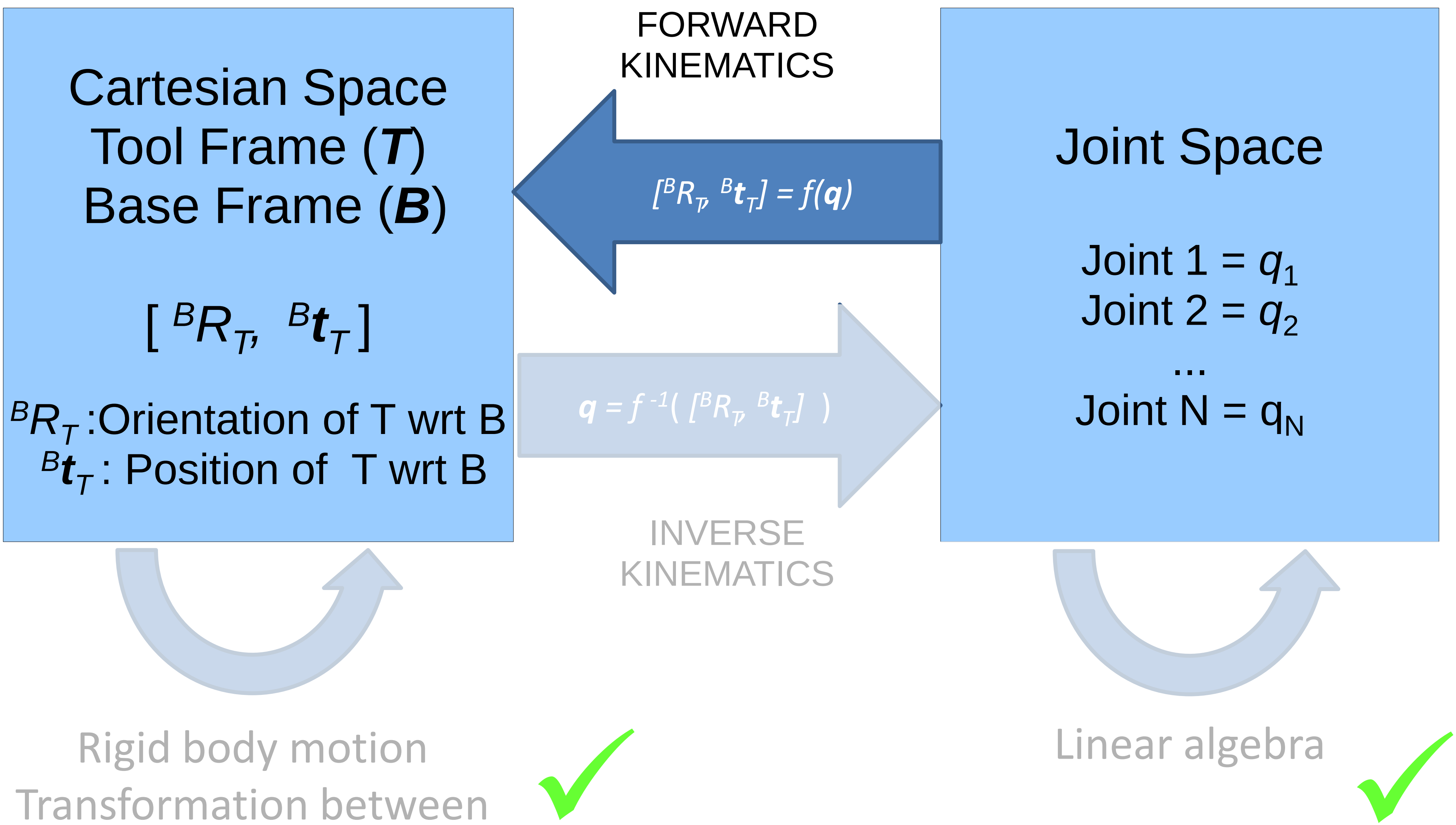
$$R = R_x R_y R_z \neq R_z R_y R_x$$

Cartesian Transformation
Kinematic Chain

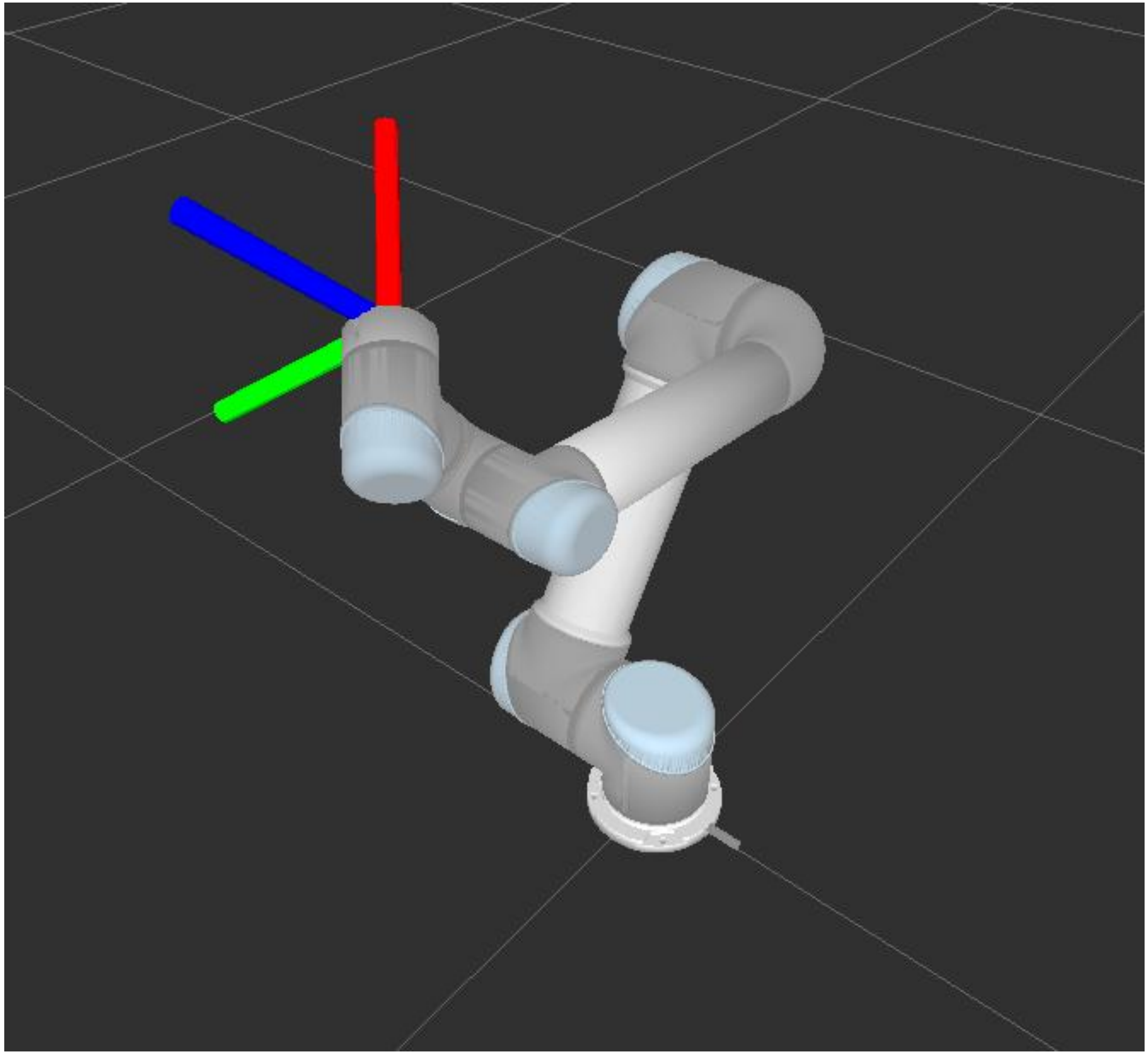


$$^{A}\mathbf{p} = {}^{A}E_{B}\ {}^{B}E_{C}\ {}^{C}\mathbf{p}$$
$$^{A}\mathbf{p} = {}^{A}E_{C}\ {}^{C}\mathbf{p}$$

Kinematics



Cartesian Space
Tool Frame (*T*)
Base Frame (*B*)
$[\,^{B}R_{T},\ ^{B}\mathbf{t}_{T}\,]$
$^{B}R_{T}$:Orientation of T wrt B
$^{B}\mathbf{t}_{T}$: Position of T wrt B
FORWARD KINEMATICS
$[^{B}R_{T},\ ^{B}\mathbf{t}_{T}] = f(\mathbf{q})$
$\mathbf{q} = f^{-1}(\ [^{B}R_{T},\ ^{B}\mathbf{t}_{T}]\ )$
INVERSE KINEMATICS
Joint Space
Joint 1 = q_1
Joint 2 = q_2
...
Joint N = q_N
Rigid body motion
Transformation between coordinate frames
Linear algebra



Forward Kinematics

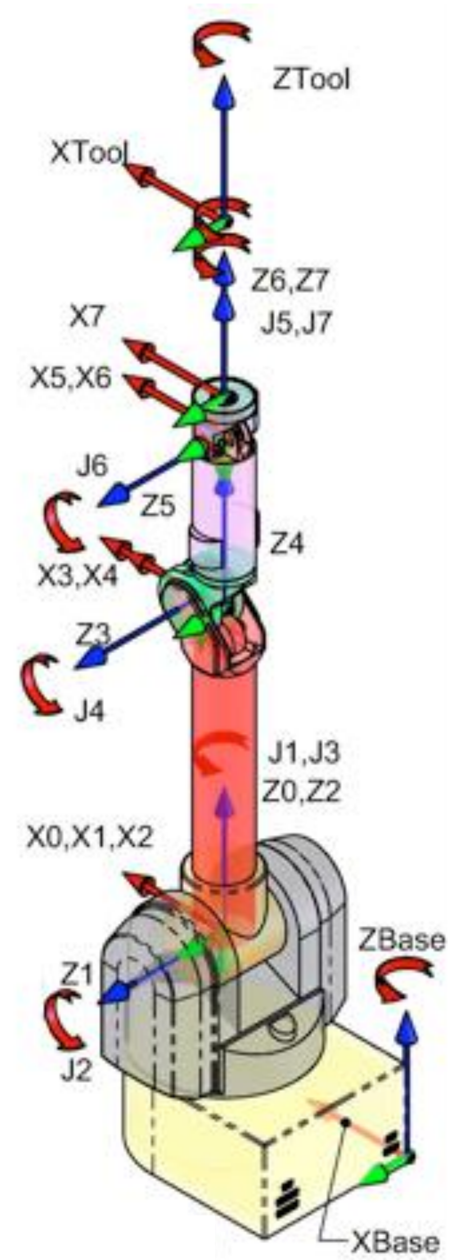
Guidelines for assigning frames to robot links:

- There are several conventions
  - Denavit Hartenberg (DH), modified DH, Hayati, etc.
  - These are conventions (habits), not laws!

1) Choose the base and tool coordinate frame
  - Make your life easy!

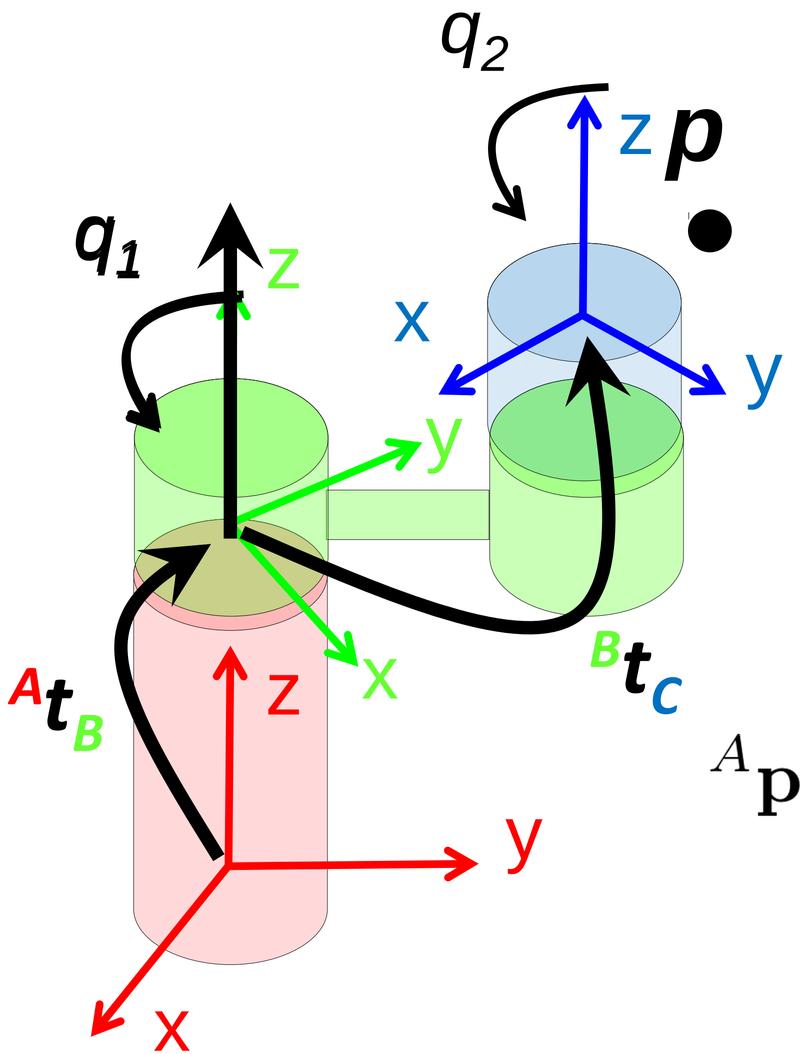
2) Start from the base and move towards the tool
  - Make your life easy!
  - In general each link has a coordinate frame.

3) Align each coordinate frame with a joint actuator
  - *Conventionally* it's the "Z" axis but this is **not** necessary and any axis can be use to represent the motion of a joint



Barrett WAM

Forward Kinematics 3D



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

$$^A\mathbf{p} = ^A E_B \ ^B\mathbf{p}$$

$$= \begin{bmatrix} R_z(q_1) & ^A\mathbf{t}_B \\ \mathbf{0} & 1 \end{bmatrix}$$

$$^B\mathbf{p} = ^B E_C \ ^C\mathbf{p}$$

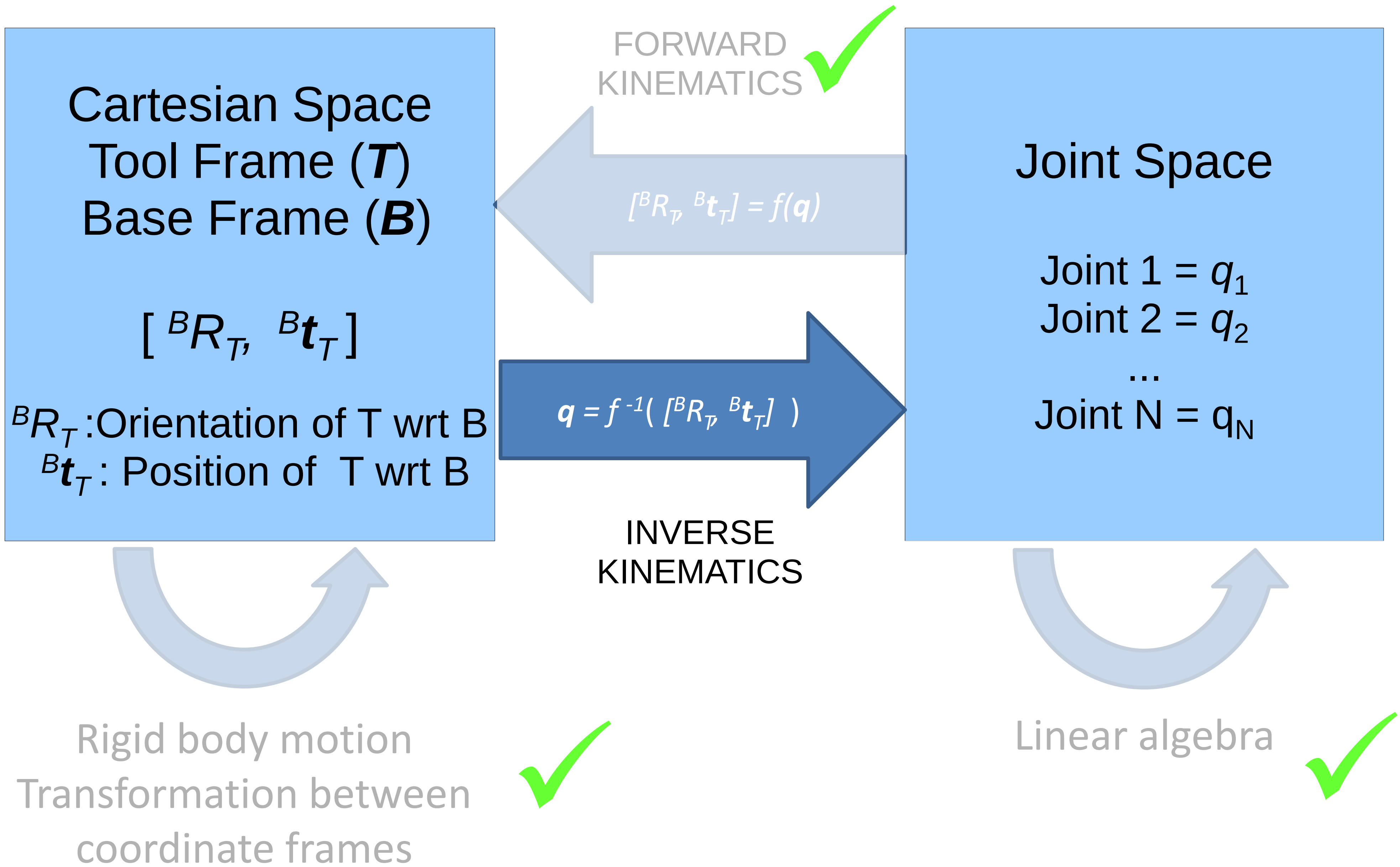
$$= \begin{bmatrix} R_z(q_2) & ^B\mathbf{t}_C \\ \mathbf{0} & 1 \end{bmatrix}$$

$$^A\mathbf{p} = ^A E_B \ ^B E_C^C \mathbf{p}$$

$$\begin{bmatrix} R_z(q_1)R_z(q_2) & ^A\mathbf{t}_B + R_z(q_1)^B\mathbf{t}_C \\ \mathbf{0} & 1 \end{bmatrix}$$

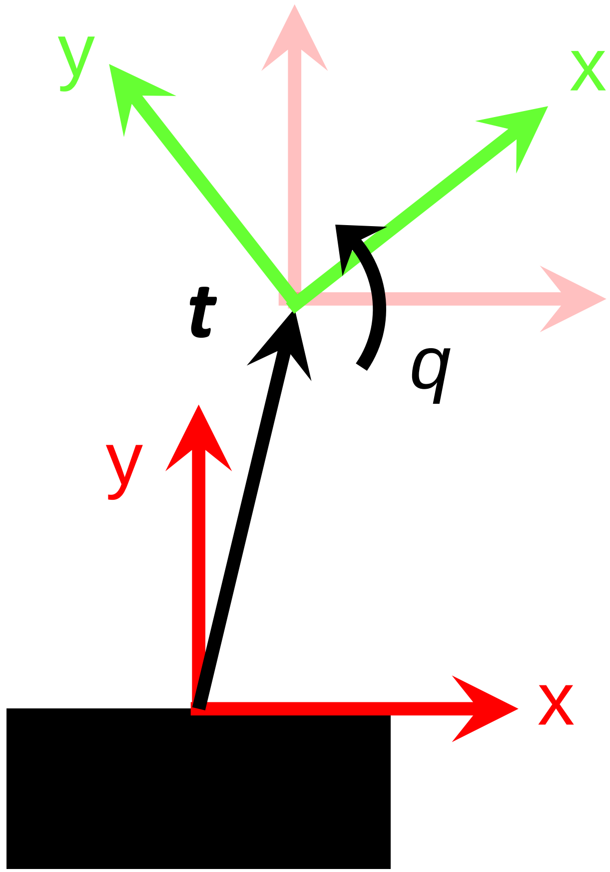
Forward Kinematics

Kinematics



Cartesian Space
Tool Frame (T)
Base Frame (B)
$[\,^{B}R_{T},\ ^{B}\boldsymbol{t}_{T}\,]$
$^{B}R_{T}$:Orientation of T wrt B
$^{B}\boldsymbol{t}_{T}$: Position of T wrt B
FORWARD KINEMATICS
$[^{B}R_{T},\ ^{B}\boldsymbol{t}_{T}] = f(\boldsymbol{q})$
$\boldsymbol{q} = f^{-1}(\ [^{B}R_{T},\ ^{B}\boldsymbol{t}_{T}]\ )$
INVERSE KINEMATICS
Joint Space
Joint 1 = q_1
Joint 2 = q_2
...
Joint N = q_N
Rigid body motion
Transformation between coordinate frames
Linear algebra

Inverse Kinematics 2D

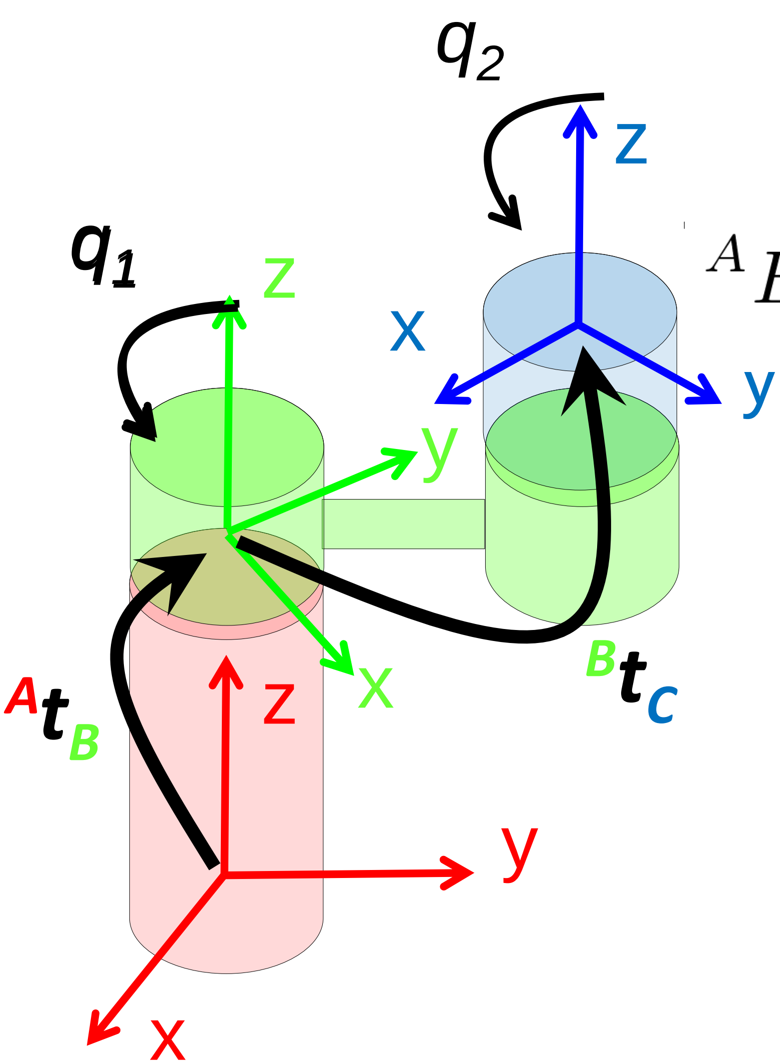


$$\begin{bmatrix} {}^{A}x \\ {}^{A}y \\ 1 \end{bmatrix} = \begin{bmatrix} \cos q & -\sin q & t_x \\ \sin q & \cos q & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} {}^{B}x \\ {}^{B}y \\ 1 \end{bmatrix}$$

Given ${}^{A}R_{B}$ and ${}^{A}\boldsymbol{t}_{B}$ find q

q only appears in ${}^{A}R_{B}$ so solving R for q is pretty easy. With several joints, the inverse kinematics gets very messy.

Inverse Kinematics 3D



Likewise, in 3D we want to solve for the position and orientation of the last coordinate frame: Find q_1 and q_2 such that

$$^{A}E_C = \begin{bmatrix} R_z(q_1)R_z(q_2) & {}^{A}\mathbf{t}_B + R_z(q_1)\ {}^{B}\mathbf{t}_C \\ \mathbf{0} & 1 \end{bmatrix}$$

Solving the inverse kinematics gets messy fast!

A) For a robot with several joints, a symbolic solution can be difficult to get

B) A numerical solution (Newton's method) is more generic

Note that the inverse kinematics is NOT

$$^{A}E_C^{-1} = {}^{C}E_A$$

Kinematics

Cartesian Space
Tool Frame (***T***)
Base Frame (***B***)

$[\ {}^{B}v_{T},\ \ {}^{B}w_{T}\ ]$

${}^{B}v_{T}$:linear vel. of T wrt B
${}^{B}w_{T}$: angular vel. of T wrt B

JACOBIAN

$[\ v, w\ ]^{T} = J(\boldsymbol{q})\ \dot{\boldsymbol{q}}$

$\dot{\boldsymbol{q}} = J^{-1}(\boldsymbol{q})\ [\ v, w\ ]^{T}$

INVERSE
JACOBIAN

Joint Space

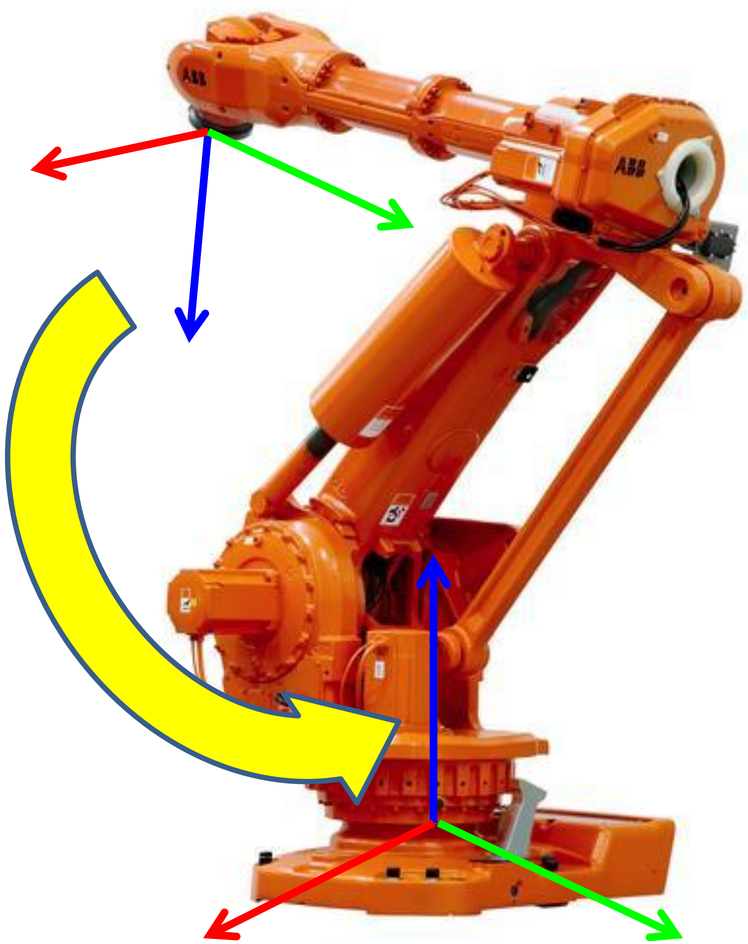
Joint 1 = $\dot{q}_1$
Joint 2 = $\dot{q}_2$
...
Joint N = $\dot{q}_N$

Rigid body motion
Transformation between
coordinate frames

Linear algebra ✓

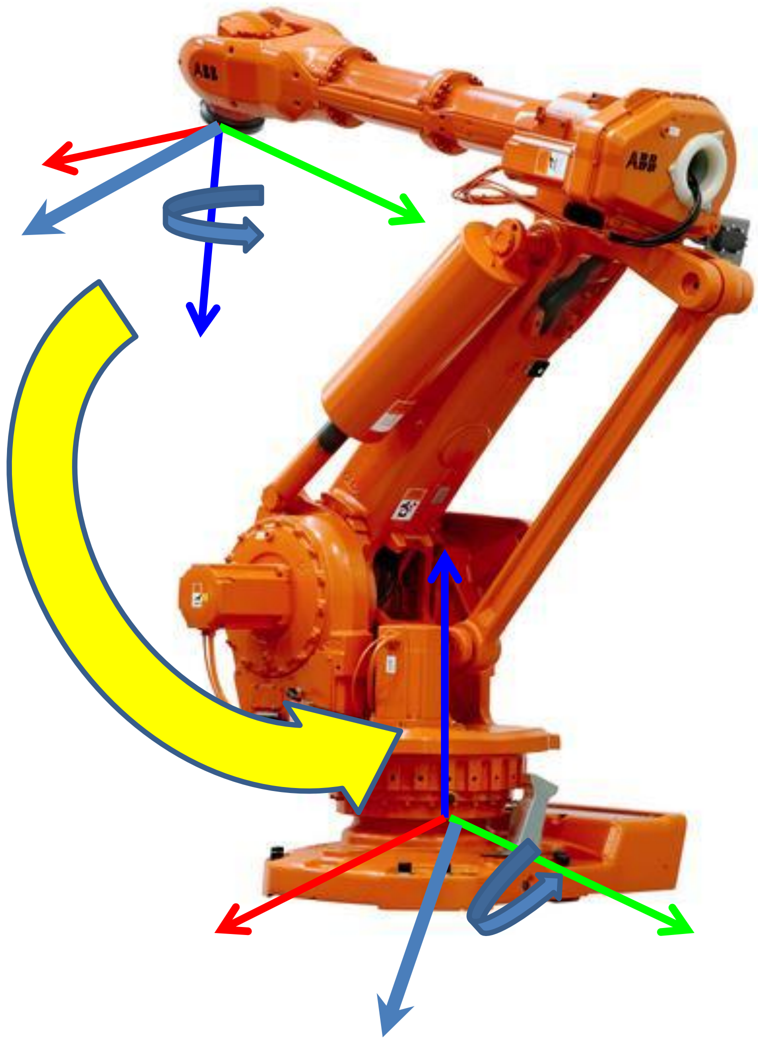
Rigid Body Transformation

Relates two coordinate frames



Rigid Body Velocity

Relate a 3D velocity in one coordinate frame to an equivalent velocity in another coordinate frame



Rotational Velocity

We note that a rotation relates the coordinates of 3D points with

$$^{A}p(t) = {^{A}R_{B}}(t)\,{^{B}p}$$

Deriving on both sides with respect to time we get

$$v_{^{A}p}(t) = \frac{d\,{^{A}p(t)}}{dt} = {^{A}\dot{R}_{B}}\,{^{B}p}$$

$$v_{^{A}p}(t) = {^{A}\dot{R}_{B}}\left({^{A}R_{B}^{-1}}\,{^{A}R_{B}}\right){^{B}p}$$

$$v_{^{A}p}(t) = \left({^{A}\dot{R}_{B}}\,{^{B}R_{A}}\right){^{A}p}$$

Rotational Velocity

$^A\dot{R}_B \, {}^A R_b^{-1}$ is skew symmetric $\hat{a} = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix}$

And the instantaneous spatial angular velocity is defined by

$$^A\hat{\omega}_B := \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} = {}^A\dot{R}_B \, {}^A R_B^{-1}$$

Where $^A w_B = [\ w_x \ \ w_y \ w_z\ ]^T$ form a vector that represents the angular velocity of a body.

Spatial Velocity

Velocity of a rigid body as seen from another frame (here called the "spatial" frame)

$$^{A}E_{B}(t)=\begin{bmatrix} ^{A}R_{B}(t) & ^{A}\mathbf{t}_{B}(t) \\ 0 & 1 \end{bmatrix}$$

$$^{A}\dot{E}_{B}\,^{A}E_{B}^{-1}=\begin{bmatrix} ^{A}\dot{R}_{B} & ^{A}\dot{\mathbf{t}}_{B} \\ 0 & 0 \end{bmatrix}\begin{bmatrix} ^{A}R_{B}^{T} & -^{A}R_{B}^{T}\,^{A}\mathbf{t}_{B} \\ 0 & 1 \end{bmatrix}=\begin{bmatrix} ^{A}\dot{R}_{B}\,^{A}R_{B}^{T} & -^{A}\dot{R}_{B}\,^{A}R_{B}^{T}\,^{A}\mathbf{t}_{B}+^{A}\dot{\mathbf{t}}_{B} \\ 0 & 0 \end{bmatrix}$$

The "***s" patial velocity*** is defined by

$$^{A}\hat{V}_{B}^{s}=^{A}\dot{E}_{B}\,^{A}E_{B}^{-1}$$

Where the linear velocity is defined by

$$^{A}v_{B}^{s}=-^{A}\dot{R}_{B}\,^{A}R_{B}^{T}\,^{A}\mathbf{t}_{B}+^{A}\dot{\mathbf{t}}_{B}$$

And the angular velocity is define as before by

$$^{A}\hat{\omega}_{B}^{s}=^{A}\dot{R}_{B}\,^{A}R_{B}^{T}$$

Body Velocity

Velocity of a rigid body with respect to its own frame

$$^{A}E_{B}(t)=\begin{bmatrix} ^{A}R_{B}(t) & ^{A}\mathbf{t}_{B}(t) \\ 0 & 1 \end{bmatrix}$$

$$^{A}\hat{V}_{B}^{b}={}^{A}E_{B}^{-1}\,{}^{A}\dot{E}_{B}=\begin{bmatrix} ^{A}R_{B}^{T}\,{}^{A}\dot{R}_{B} & ^{A}R_{B}^{T}\,{}^{A}\dot{\mathbf{t}}_{B} \\ 0 & 0 \end{bmatrix}$$

The “***b”ody velocity*** is defined by

$$^{A}\hat{V}_{B}^{b}={}^{A}E_{B}^{-1}\,{}^{A}\dot{E}_{B}$$

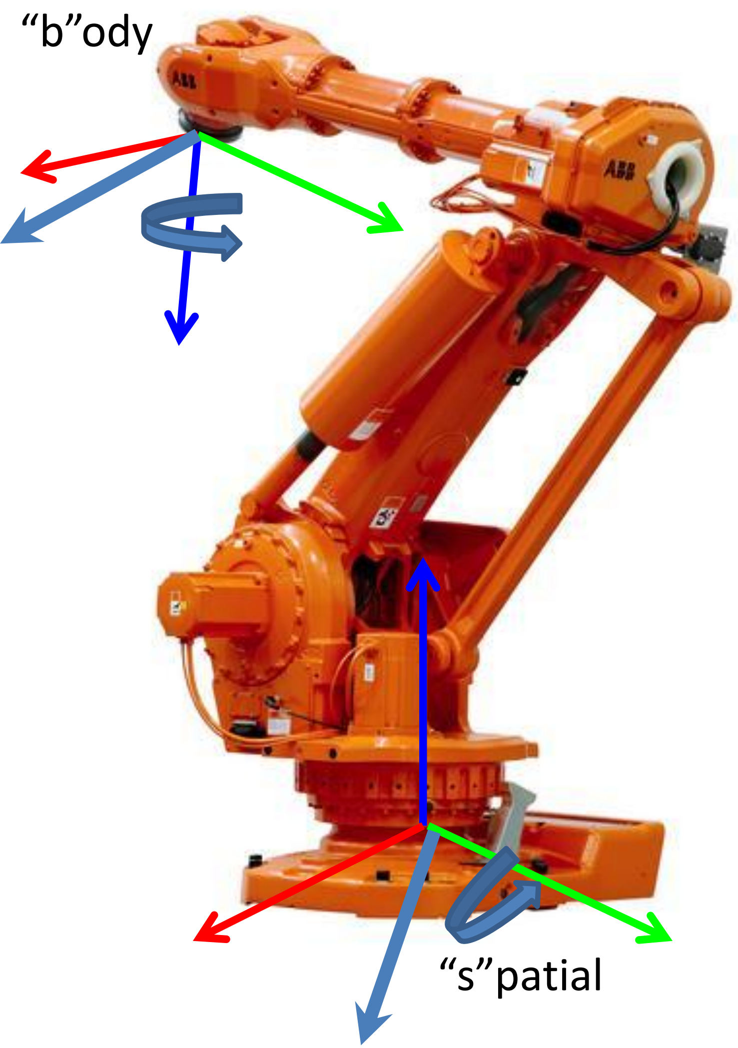
Where the linear velocity is defined by

$$^{A}v_{B}^{b}={}^{A}R_{B}^{T}\,{}^{A}\dot{\mathbf{t}}_{B}$$

And the angular velocity is define as before by

$$^{A}\hat{\omega}_{B}^{b}={}^{A}R_{B}^{T}\,{}^{A}\dot{R}_{B}$$

Transform Body Velocity to Spatial Velocity



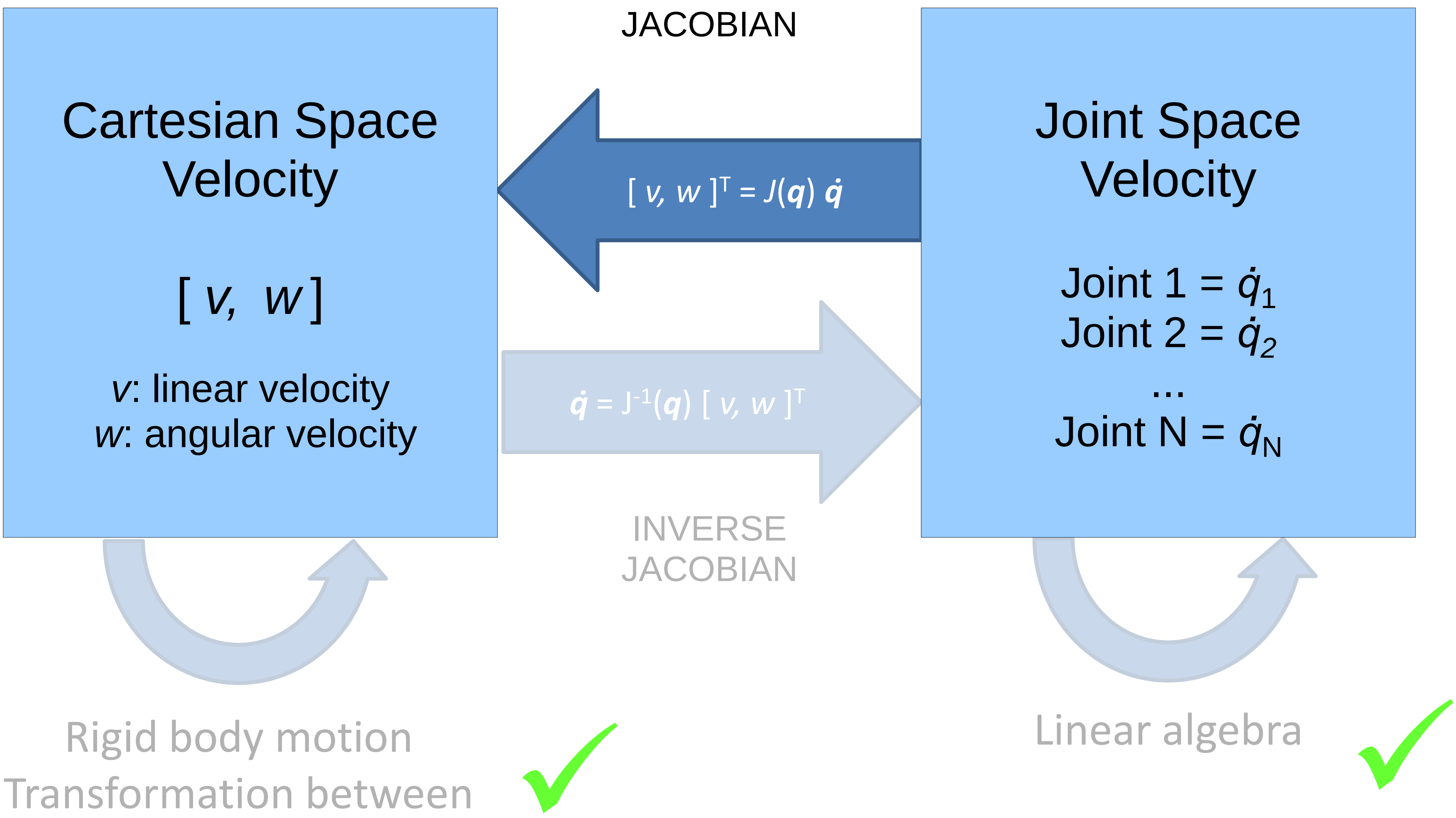
If you are given a "body velocity":

1) Rotate the tool about a given axis (in the tool frame)
2) Drive the tool along a given axis (in the tool frame)

Then you can compute the equivalent velocity in the "spatial" frame according to

$$\begin{bmatrix} {}^{A}v_{B}^{s} \\ {}^{A}\omega_{B}^{s} \end{bmatrix} = \begin{bmatrix} {}^{A}R_{B} & {}^{A}\hat{\mathbf{t}}_{B}\,{}^{A}R_{B} \\ 0 & {}^{A}R_{B} \end{bmatrix} \begin{bmatrix} {}^{A}v_{B}^{b} \\ {}^{A}\omega_{B}^{b} \end{bmatrix}$$

Kinematics


JACOBIAN
Cartesian Space Velocity
$[\,v,\ w\,]$
v: linear velocity
w: angular velocity
$[\,v,\ w\,]^T = J(\boldsymbol{q})\,\dot{\boldsymbol{q}}$
$\dot{\boldsymbol{q}} = J^{-1}(\boldsymbol{q})\,[\,v,\ w\,]^T$
Joint Space Velocity
Joint 1 = $\dot{q}_1$
Joint 2 = $\dot{q}_2$
...
Joint N = $\dot{q}_N$
INVERSE JACOBIAN
Rigid body motion
Transformation between coordinate frames
Linear algebra

Manipulator Jacobian

Spatial velocity of the "T"ool in the "B"ase frame is

$${}^B\hat{V}_T^s = {}^B\dot{E}_T(t)\,{}^BE_T^{-1}(t)$$

We change the time varying trajectory to be a time varying joint trajectory

$${}^B\hat{V}_T^s = {}^B\dot{E}_T(\mathbf{q}(t))\,{}^BE_T^{-1}(\mathbf{q}(t))$$

Derivative of the forward kinematics wrt q_i

Inverse of the forward kinematics

Applying the chain rule

$$\frac{\partial E(q(t))}{\partial t} = \frac{\partial E(q(t))}{\partial q}\frac{\partial q(t)}{\partial t}$$

$${}^B\hat{V}_T^s = \sum_{i=1}^{N}\left(\frac{\partial\, {}^BE_T}{\partial q_i}\dot{q}_i\right){}^TE_B(\mathbf{q}(t))$$

$${}^B\hat{V}_T^s = \sum_{i=1}^{N}\left(\frac{\partial\, {}^BE_T}{\partial q_i}\,{}^TE_B(\mathbf{q}(t))\right)\dot{q}_i$$

Manipulator Jacobian

Lets rewrite the previous result as

$$\begin{bmatrix} {}^{B}v_{T}^{s} \\ {}^{B}\omega_{T}^{s} \end{bmatrix} = J(\mathbf{q})\dot{\mathbf{q}}$$

Where $J(\boldsymbol{q})$ is a 6xN matrix called the manipulator Jacobian that relates joint velocities to Cartesian velocities.
Note that the Jacobian depends on **q** and, therefore, is configuration dependant.

Manipulator Jacobian

- Each column of J(q) is given by the linear and angular velocities elements (v_x, v_y, v_z, w_x, w_y, w_z) found in each

$$\frac{\partial^B E_T}{\partial q_i}^T E_B$$

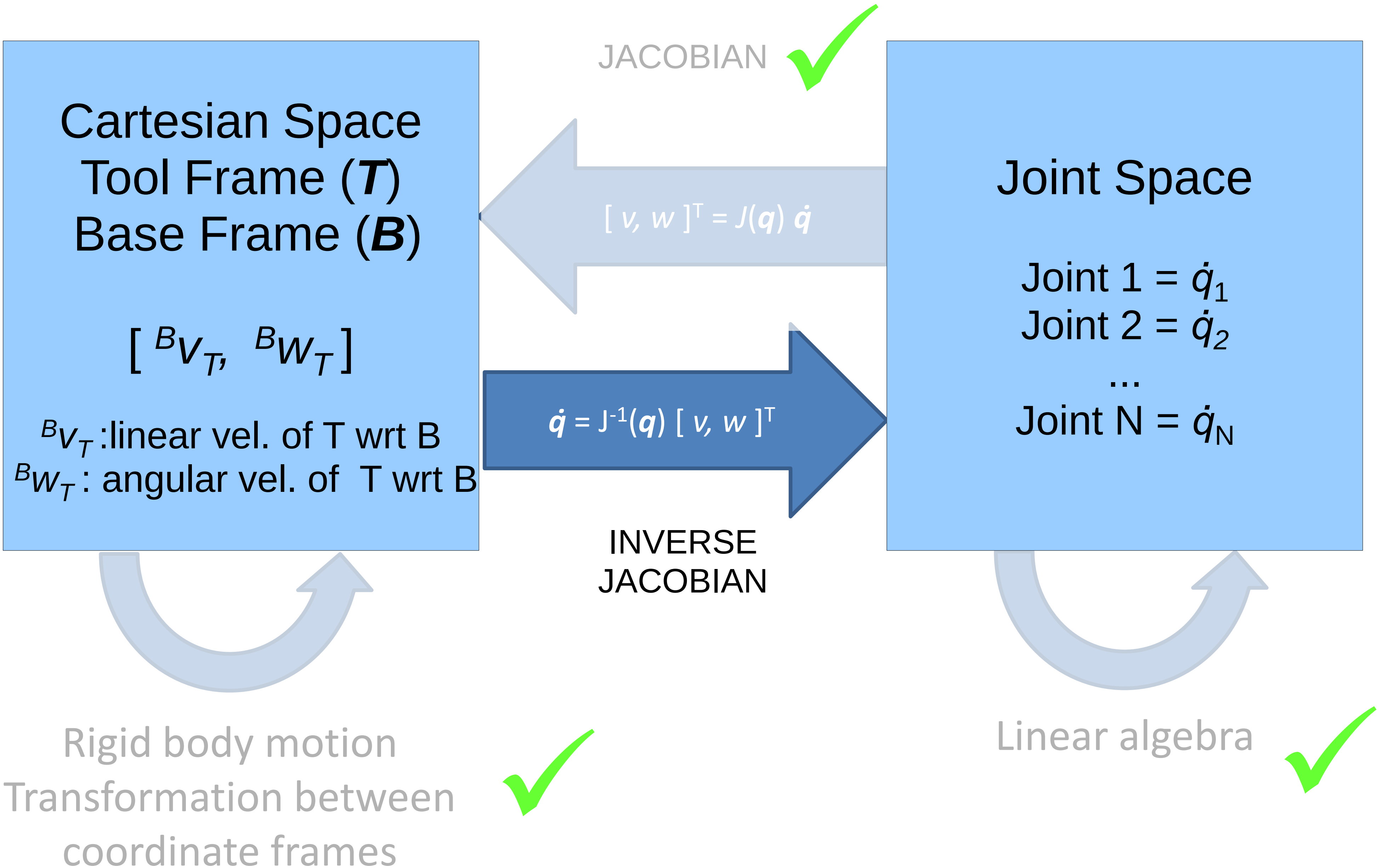
- Thus, given the following “extraction” operator

$$\begin{bmatrix} \hat{\omega} & v \\ 0 & 0 \end{bmatrix}^{\vee} = \begin{bmatrix} v \\ \omega \end{bmatrix}$$

J has the following structure

$$J(\mathbf{q}) = \left[\left(\frac{\partial^B E_T}{\partial q_1}^T E_B \right)^{\vee} \quad \cdots \quad \left(\frac{\partial^B E_T}{\partial q_N}^T E_B \right)^{\vee} \right]$$

Kinematics



JACOBIAN
Cartesian Space
Tool Frame (***T***)
Base Frame (***B***)
$[\ {}^{B}v_{T},\ {}^{B}w_{T}\ ]$
${}^{B}v_{T}$:linear vel. of T wrt B
${}^{B}w_{T}$: angular vel. of T wrt B
$[\ v, w\ ]^{T} = J(\boldsymbol{q})\ \dot{\boldsymbol{q}}$
$\dot{\boldsymbol{q}} = J^{-1}(\boldsymbol{q})\ [\ v, w\ ]^{T}$
Joint Space
Joint 1 = $\dot{q}_1$
Joint 2 = $\dot{q}_2$
...
Joint N = $\dot{q}_N$
INVERSE
JACOBIAN
Rigid body motion
Transformation between
coordinate frames
Linear algebra

Manipulator Jacobian

We just derived that given a vector of joint velocities, the velocity of the tool as seen in the base of the robot is given by

$$\begin{bmatrix} {}^B\mathbf{v} \\ {}^B\boldsymbol{\omega} \end{bmatrix} = J(\mathbf{q})\dot{\mathbf{q}}$$

If, instead we want to tool to move with a velocity expressed in the **base** frame, the corresponding joint velocities can be computed by

$$\dot{\mathbf{q}} = J^{-1}(\mathbf{q}) \begin{bmatrix} {}^B\mathbf{v} \\ {}^B\boldsymbol{\omega} \end{bmatrix}$$

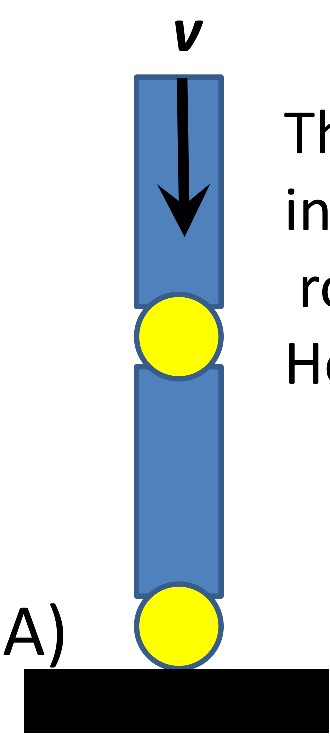
Inverting a matrix is much easier than computing the inverse kinematics!

Manipulator Jacobian

What if the Jacobian has no inverse?

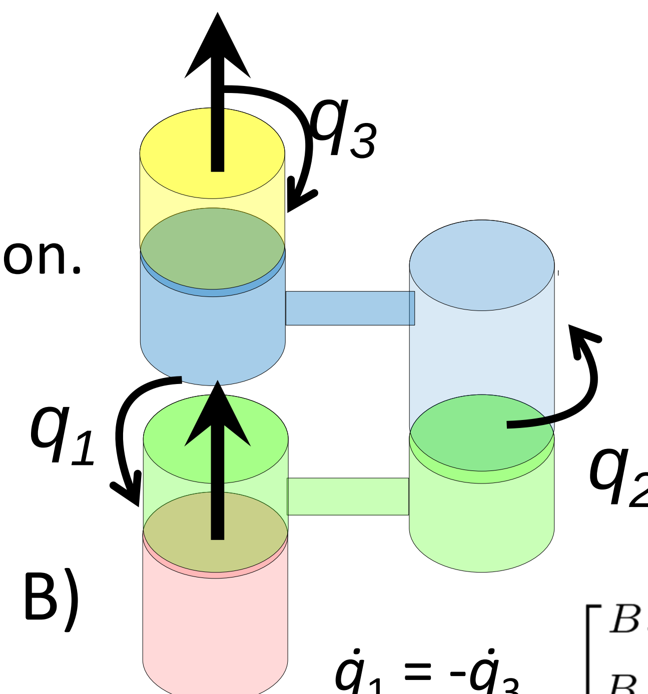
A) No solution: The velocity is impossible

B) Infinity of solutions: Some joints can be moved without affecting the velocity (i.e. when two axes are colinnear)



A)

The robot cannot move in this direction when the robot is in this configuration. Hence $J(\boldsymbol{q})$ is singular.



B)

In this configuration, q_1 and q_3 can counter rotate. Hence $J(\boldsymbol{q})$ is singular.

$\dot{q}_1 = -\dot{q}_3$ $\begin{bmatrix} {}^{B}\mathbf{v} \\ {}^{B}\boldsymbol{\omega} \end{bmatrix} = \mathbf{0}$