Equations of Motion and Articulated Robots

Robert Stengel
Robotics and Intelligent Systems
MAE 345, Princeton University, 2015

Equations of Motion
MATLAB, Simulink, SimMechanics
Articulated Robots

Inertia Matrix Expressed in Inertial Frame is Not Constant if Body is Rotating

Inertia Matrix Expressed in Inertial Frame is Not Constant if Body is Rotating

Newton's 2^{nd} Law, applied to rotational motion (in inertial frame)

Rate of change of angular momentum = applied moment (or torque), m

$$\boxed{\frac{d\mathbf{h}}{dt} = \frac{d(\boldsymbol{I}\boldsymbol{\omega})}{dt} = \mathbf{m} \quad \text{[moment vector]}}$$

$$\boxed{\begin{bmatrix} \dot{h}_x \\ \dot{h}_y \\ \dot{h}_z \end{bmatrix} = \begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix}}$$

Chain Rule

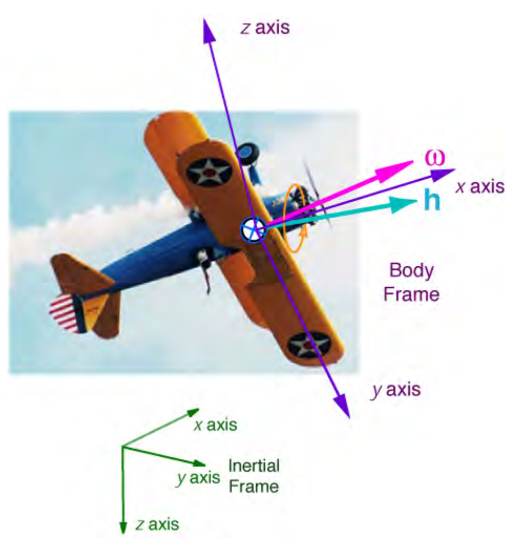
$$\boxed{\frac{d(\boldsymbol{I}\boldsymbol{\omega})}{dt} = \dot{\mathbf{h}} = \dot{\boldsymbol{I}}\boldsymbol{\omega} + \boldsymbol{I}\dot{\boldsymbol{\omega}}}$$

... and in an inertial frame

$$\boxed{\dot{\boldsymbol{I}}_I = \frac{d\boldsymbol{I}_I}{dt} \neq 0}$$

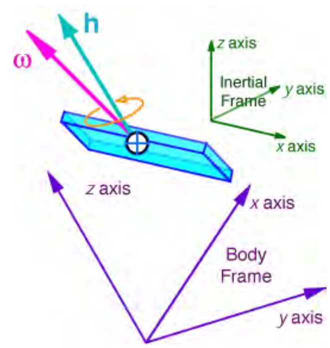
How do We get Rid of dI_I / dt in the Angular Momentum Equation?

- **Write the dynamic equation in body-referenced frame**
  - **With constant mass, inertial properties are unchanging in body reference frame**
  - **... but the frame is "non-Newtonian" or "non-inertial"**



Dynamic equation modified to account for rotating frame

Angular Momentum and Rate are Vectors



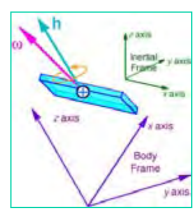
Can be expressed in either an inertial or body frame

Frames are transformed by the rotation matrix and its inverse

$$\mathbf{h}_B = \mathbf{H}_I^B \mathbf{h}_I \qquad \boldsymbol{\omega}_B = \mathbf{H}_I^B \boldsymbol{\omega}_I$$

$$\mathbf{h}_I = \mathbf{H}_B^I \mathbf{h}_B \qquad \boldsymbol{\omega}_I = \mathbf{H}_B^I \boldsymbol{\omega}_B$$

Vector Derivative Expressed in a Rotating Frame



Chain Rule

$$\dot{\mathbf{h}}_I = \mathbf{H}_B^I \dot{\mathbf{h}}_B + \dot{\mathbf{H}}_B^I \mathbf{h}_B$$

Alternatively

$$\dot{\mathbf{h}}_I = \mathbf{H}_B^I \dot{\mathbf{h}}_B + \boldsymbol{\omega}_I \times \mathbf{h}_I = \mathbf{H}_B^I \dot{\mathbf{h}}_B + \tilde{\boldsymbol{\omega}}_I \mathbf{h}_I$$

Cross-product-equivalent matrix of angular rate:

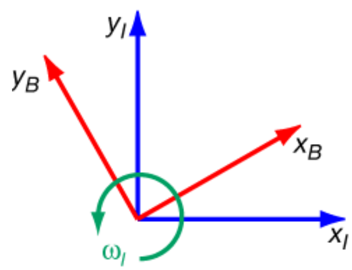
$$\tilde{\boldsymbol{\omega}} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}$$

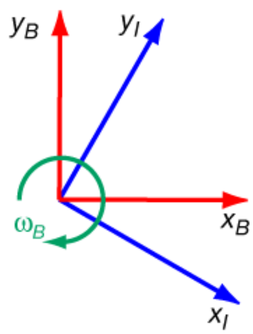
Consequently

$$\dot{\mathbf{H}}_B^I \mathbf{h}_B = \tilde{\boldsymbol{\omega}}_I \mathbf{h}_I = \tilde{\boldsymbol{\omega}}_I \mathbf{H}_B^I \mathbf{h}_B$$

Rate of Change of Angular Momentum due to External Moment

Positive rotation of Frame B w.r.t. Frame A is a negative rotation of Frame A w.r.t. Frame B

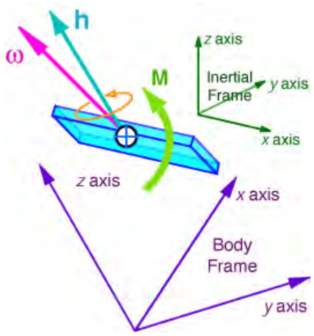




Rate of Change of Angular Momentum due to External Moment

Angular momentum change in the body frame

$$
\begin{aligned}
\dot{\mathbf{h}}_B &= \mathbf{H}_I^B\dot{\mathbf{h}}_I + \dot{\mathbf{H}}_I^B\mathbf{h}_I = \mathbf{H}_I^B\dot{\mathbf{h}}_I - \boldsymbol{\omega}_B \times \mathbf{h}_B \\
&= \mathbf{H}_I^B\dot{\mathbf{h}}_I - \tilde{\boldsymbol{\omega}}_B\mathbf{h}_B = \mathbf{H}_I^B\mathbf{m}_I - \tilde{\boldsymbol{\omega}}_B\boldsymbol{I}_B\boldsymbol{\omega}_B \\
&= \mathbf{m}_B - \tilde{\boldsymbol{\omega}}_B\boldsymbol{I}_B\boldsymbol{\omega}_B
\end{aligned}
$$



Inertial-frame moments (torques) transformed to body-frame moments

$$
\mathbf{m}_I = \begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix}_I ; \quad \mathbf{m}_B = \mathbf{H}_I^B\mathbf{m}_I = \begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix}_B
$$

Rate of Change of Body-Referenced Angular Rate due to External Moment

Angular momentum rate of change expressed in the body frame

$$\boxed{\dot{\mathbf{h}}_B = \boldsymbol{I}_B \dot{\boldsymbol{\omega}}_B = \mathbf{m}_B - \tilde{\boldsymbol{\omega}}_B \boldsymbol{I}_B \boldsymbol{\omega}_B}$$

Angular velocity rate of change

$$\boxed{\dot{\boldsymbol{\omega}}_B = \boldsymbol{I}_B^{-1}\left(\mathbf{m}_B - \tilde{\boldsymbol{\omega}}_B \boldsymbol{I}_B \boldsymbol{\omega}_B\right)}$$

Translational Dynamics

Rate of change of the center of mass's translational position

$$\boxed{\dot{\mathbf{r}}_I = \mathbf{v}_I = \mathbf{H}_B^I \mathbf{v}_B}$$

Express translational dynamics in the body frame of reference

$$\boxed{\dot{\mathbf{v}}_I = \frac{1}{m}\mathbf{f}_I}$$

$$\dot{\mathbf{v}}_B = \mathbf{H}_I^B \dot{\mathbf{v}}_I - \tilde{\boldsymbol{\omega}}_B \mathbf{v}_B = \frac{1}{m}\mathbf{H}_I^B \mathbf{f}_I - \tilde{\boldsymbol{\omega}}_B \mathbf{v}_B$$

$$= \frac{1}{m}\mathbf{f}_B - \tilde{\boldsymbol{\omega}}_B \mathbf{v}_B$$

Body-axis force and velocity vectors

$$\mathbf{f}_B = \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix}_B \qquad \mathbf{v}_B = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}_B = \begin{bmatrix} u \\ v \\ w \end{bmatrix}$$

Same form as the body-axis angular rate equation

$$\boxed{\dot{\boldsymbol{\omega}}_B = \boldsymbol{I}_B^{-1}\left(\mathbf{m}_B - \tilde{\boldsymbol{\omega}}_B \boldsymbol{I}_B \boldsymbol{\omega}_B\right)}$$

Numerical Solutions Using MATLAB and Simulink

Task : Calculate $x_1(t)$ and $x_2(t)$ for $t = 1$ to 10 sec

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} x_2(t) \\ -x_1(t) - x_2(t) \end{bmatrix}$$

with initial conditions

$$\begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} = \begin{bmatrix} 0 \\ 10 \end{bmatrix}$$

MATLAB Models of Dynamic Systems

Systems are described by *instructions*

Main Script

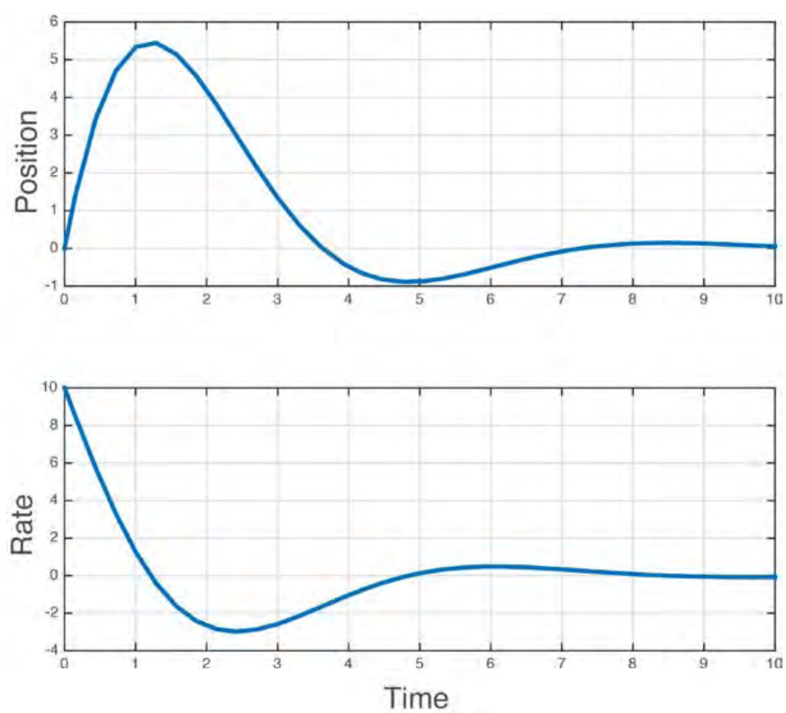
```
% Linear 2nd-Order Example
clear
tspan  =  [0 10];
xo = [0, 10];
[t,x] = ode23('Lin',tspan,xo);

subplot(2,1,1)
plot(t,x(:,1))
ylabel('Position'), grid
subplot(2,1,2)
plot(t,x(:,2))
xlabel('Time'), ylabel('Rate'), grid
```

Function

```
function   xdot   =  Lin(t,x)
%   Linear Ordinary Differential Equation
% x(1) = Position
% x(2) = Rate
xdot = [x(2)
        -x(1) - x(2)];
```

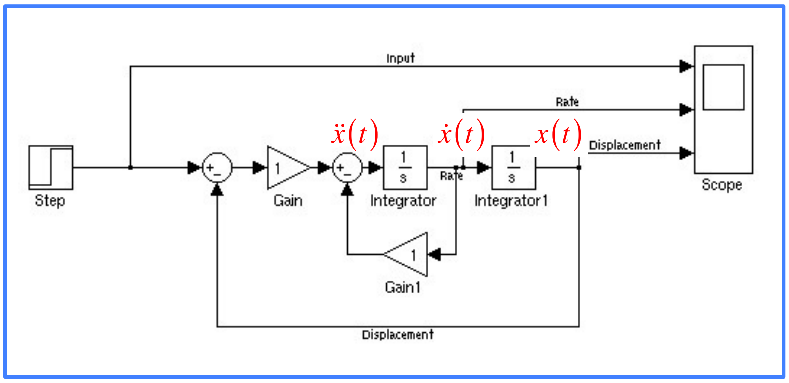

MATLAB Initial-Condition Output



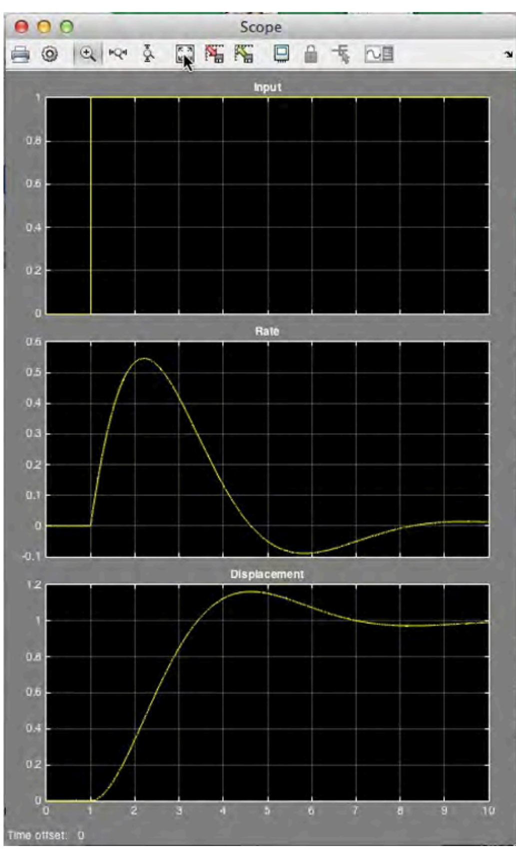
Simulink Models of Dynamic Systems

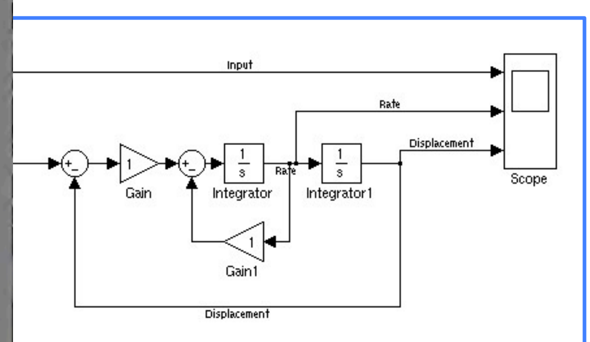
Systems are described by *block diagrams*

$$\frac{d^2x(t)}{dt^2} = \ddot{x}(t) = -x(t) - \dot{x}(t) + u(t)$$

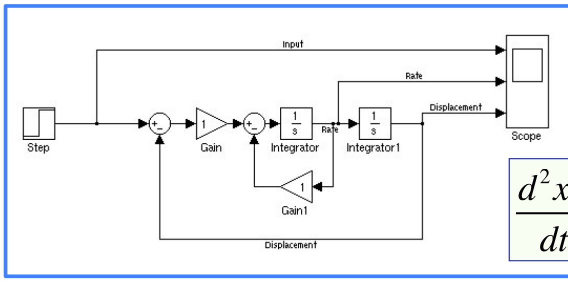


Simulink Step Response





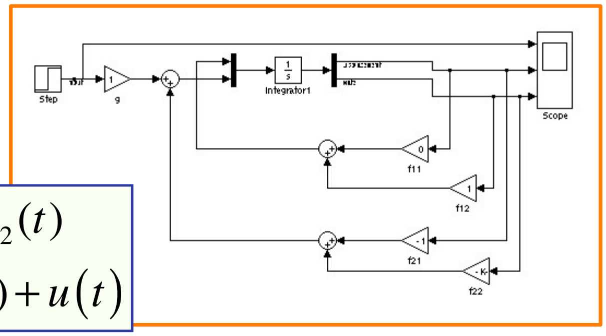
Alternative Simulink Models of 2^{nd}-Order Systems



Single 2^{nd}-order model, with step input and damping

$$\frac{d^2x(t)}{dt^2} = \ddot{x}(t) = -x(t) - \dot{x}(t) + u(t)$$

State-space model (two 1^{st}-order equations), with step input and damping



$$\dot{x}_1(t) = (0)x_1(t) + (1)x_2(t)$$
$$\dot{x}_2(t) = -(1)x_1(t) - Kx_2(t) + u(t)$$

Dynamics of Angular Position

Relationship Between Euler-Angle Rates and Body-Axis Rates

- **Body-axis angular rate vector components are orthogonal**

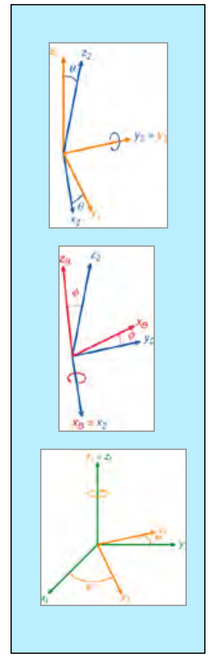
$$\boldsymbol{\omega}_B = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}_B = \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

- **Euler angles form a non-orthogonal vector**

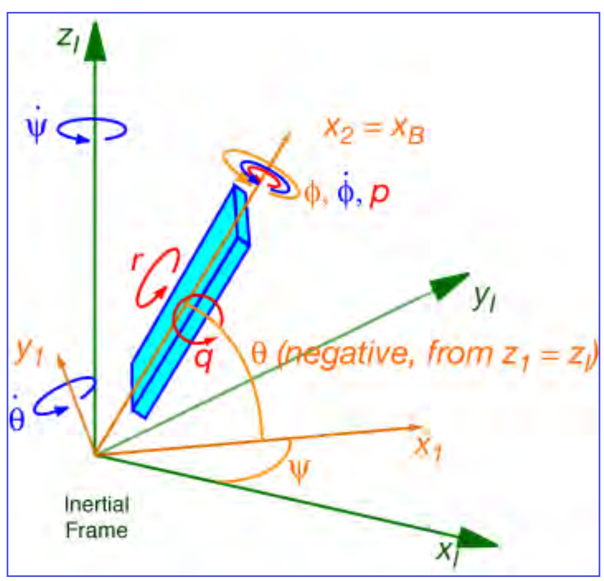
$$\boldsymbol{\Theta} \triangleq \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}$$

- **Euler-angle rate vector is not orthogonal**

$$\dot{\boldsymbol{\Theta}} = \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \neq \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}_I = \boldsymbol{\omega}_I$$



Transformation From Euler-Angle Rates to Body-Axis Rates



- $\dot{\psi}$ **is measured in the Inertial Frame**
- $\dot{\theta}$ **is measured in Intermediate Frame #1**
- $\dot{\phi}$ **is measured in Intermediate Frame #2**

Sequential Transformations from Euler-Angle Rates to Body-Axis Rates

$\dot{\psi}$ is measured in the Inertial Frame

$\dot{\theta}$ is measured in Intermediate Frame #1

$\dot{\phi}$ is measured in Intermediate Frame #2

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \mathbf{H}_2^B \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + \mathbf{H}_2^B \mathbf{H}_1^2 \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix}$$

... which is

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \triangleq \mathbf{L}_I^B \dot{\mathbf{\Theta}}$$

Inversion to Transform Body-Axis Rates to Euler-Angle Rates

Transformation is not orthonormal

$$\mathbf{L}_I^B = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{bmatrix}$$

Inverse transformation is not the transpose

$$\left(\mathbf{L}_I^B\right)^{-1} \triangleq \mathbf{L}_B^I \neq \left(\mathbf{L}_I^B\right)^T$$

$$\left(\mathbf{L}_I^B\right)^{-1} = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{bmatrix}^{-1} = \frac{Adj\left(\mathbf{L}_I^B\right)}{\det\left(\mathbf{L}_I^B\right)}$$

Inverse Transformation for Euler-Angle Rates

$$\left(\mathbf{L}_I^B\right)^{-1} = \frac{Adj\left(\mathbf{L}_I^B\right)}{\det\left(\mathbf{L}_I^B\right)} = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{bmatrix}^{-1} = \boxed{\begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{bmatrix}}$$

Euler-angle rates from body-axis rates

$$\boxed{\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \mathbf{L}_B^I \boldsymbol{\omega}_B}$$

Can the inversion become singular?
What does this mean?

Summary of Six-Degree-of-Freedom (Rigid Body) Equations of Motion

$$\dot{\mathbf{r}}_I = \mathbf{H}_B^I \mathbf{v}_B$$

$$\dot{\mathbf{v}}_B = \frac{1}{m}\mathbf{f}_B - \tilde{\boldsymbol{\omega}}_B \mathbf{v}_B$$

$$\dot{\boldsymbol{\Theta}} = \mathbf{L}_B^I \boldsymbol{\omega}_B$$

$$\dot{\boldsymbol{\omega}}_B = \boldsymbol{I}_B^{-1}\left(\mathbf{m}_B - \tilde{\boldsymbol{\omega}}_B \boldsymbol{I}_B \boldsymbol{\omega}_B\right)$$

Translational position and velocity

$$\mathbf{r}_I = \begin{bmatrix} x \\ y \\ z \end{bmatrix}; \quad \mathbf{v}_B = \begin{bmatrix} u \\ v \\ w \end{bmatrix}$$

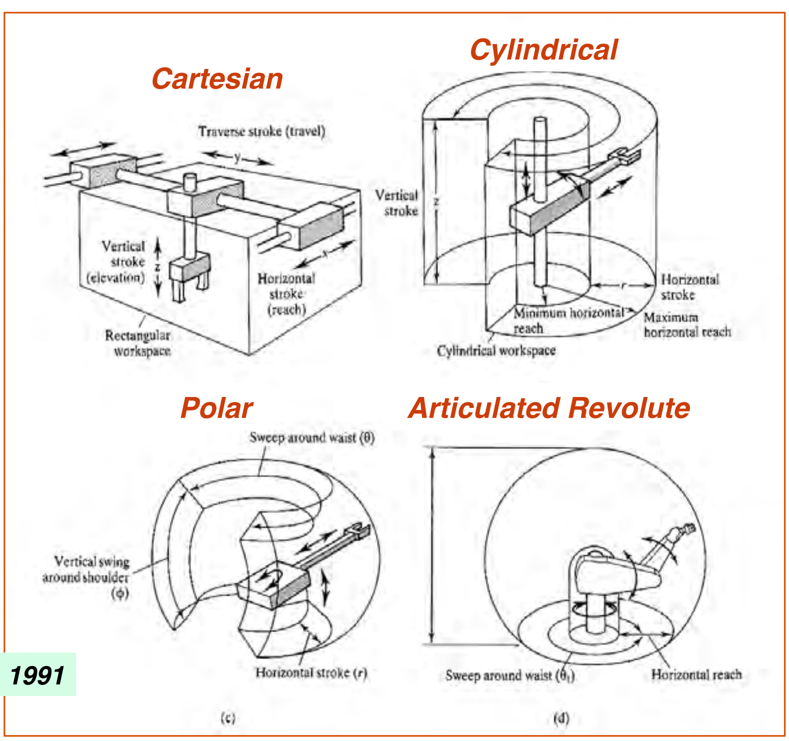
Rotational position and velocity

$$\boldsymbol{\Theta} = \begin{bmatrix} \varphi \\ \theta \\ \psi \end{bmatrix}; \quad \boldsymbol{\omega}_B = \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

Rigid-body dynamic equations are nonlinear

Articulated Robots

Assembly Robot Configurations

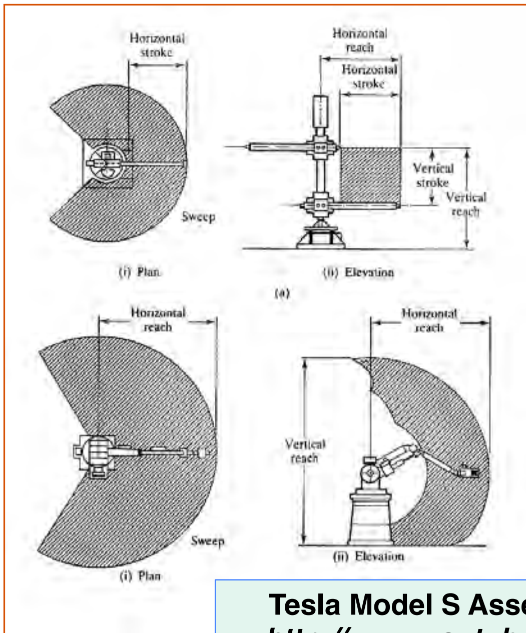


McKerrow, 1991

Assembly Robot Workspaces

Cylindrical

Articulated Revolute



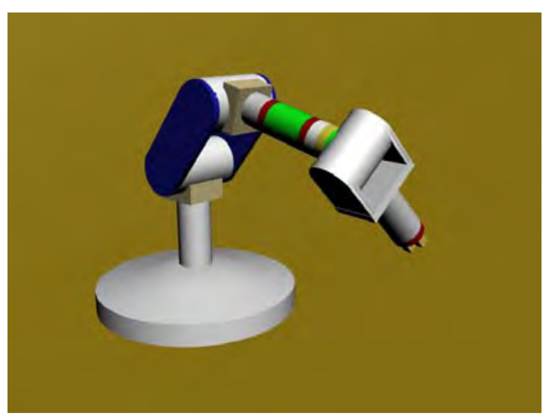
McKerrow, 1991

Tesla Model S Assembly
http://www.youtube.com/watch?v=8_lfxPI5ObM

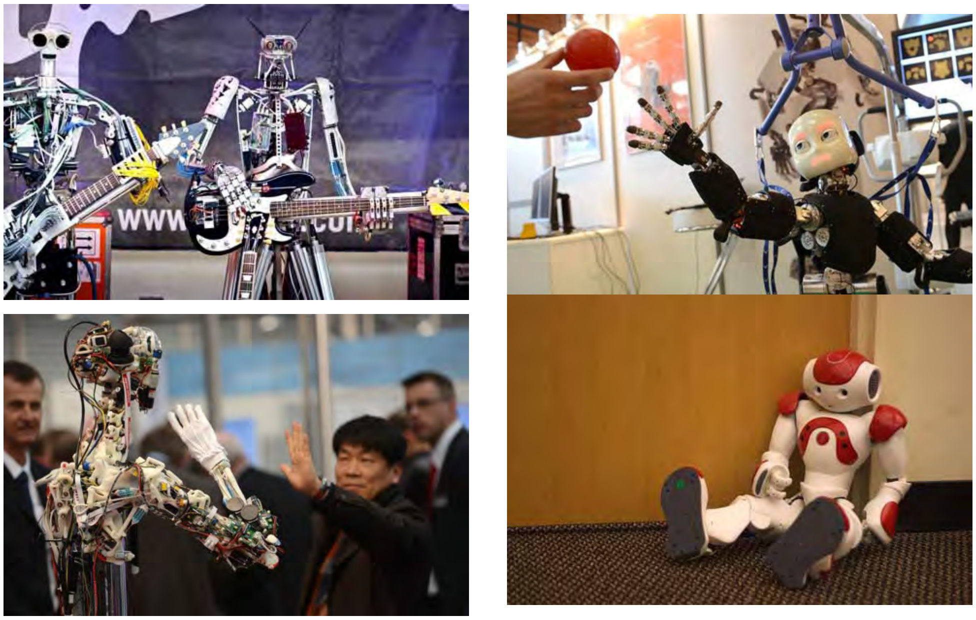
Serial Robotic Manipulators

Proximal link: closer to the base
Distal link: farther from the base

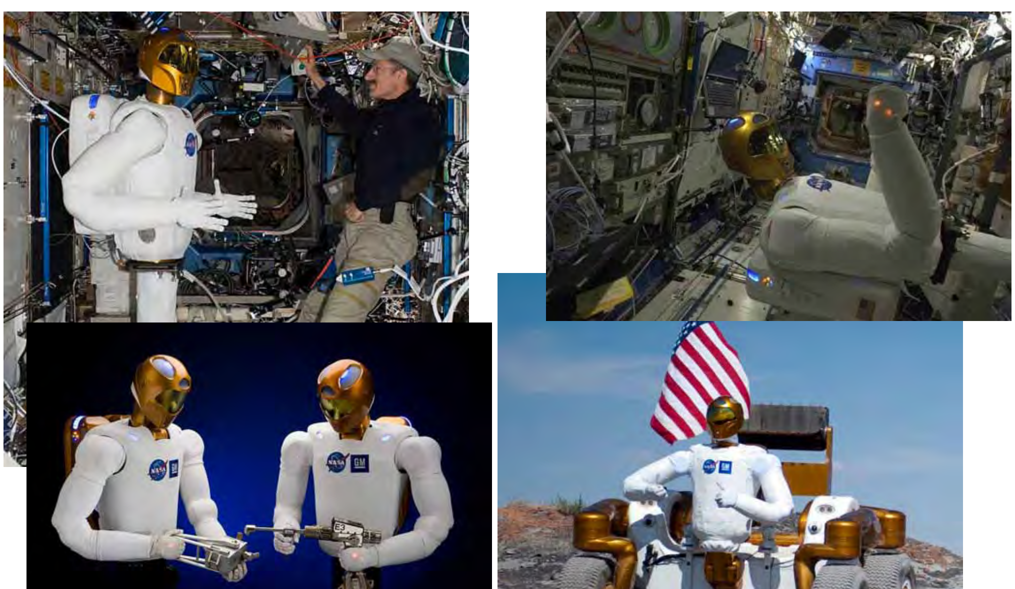
- **Serial chain of robotic links and joints**
  - **Large workspace**
  - **Low stiffness**
  - **Cumulative errors from link to link**
  - **Proximal links carry the weight and load of distal links**
  - **Actuation of proximal joints affects distal links**
  - **Limited load-carrying capability at end effecter**



Humanoid Robots



NASA/GM Robonaut

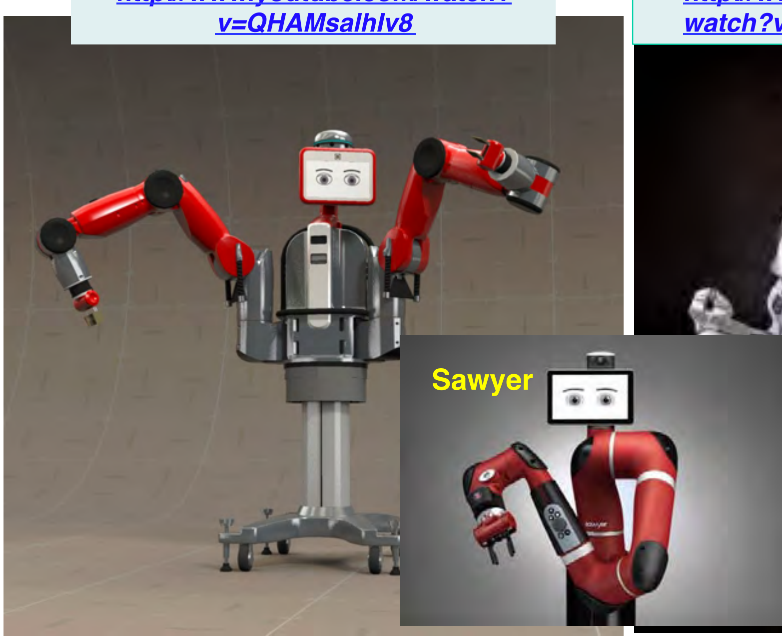


Robonaut
http://www.youtube.com/watch?v=g3u48T4Vx7k

Disney Audio-Animatronics, 1967

Baxter, Sawyer, and the PR2

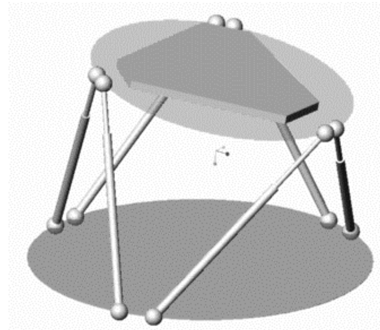
Baxter
http://www.youtube.com/watch?v=QHAMsalhlv8



PR2
http://www.youtube.com/watch?v=HMx1xW2E4Gg

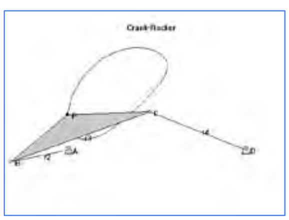
Parallel Robotic Mechanisms

- **End plate is directly actuated by multiple links and joints (*kinematic chains*)**
  - **Restricted workspace**
  - **Common link-joint configuration**
  - **Light construction**
  - **Stiffness**
  - **High load-carrying capacity**

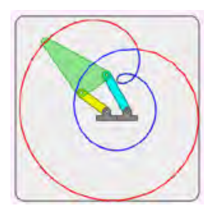


Stewart Platform
http://www.youtube.com/watch?v=QdKo9PYwGaU

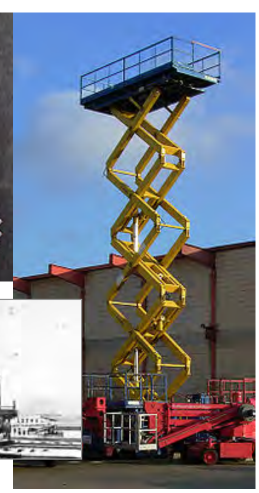
Pick-and-Place Robot
http://www.youtube.com/watch?v=i4oBExI2KiQ

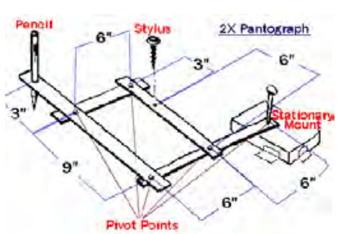


Four-Bar Linkage



- **Closed-loop structure**
- **Rotational joints**
- **Planar motion**
- **Proportions of link lengths determine pattern of motion**
- **Examples**
  - **Double wishbone suspension**
  - **Pantograph**
  - **Scissor lift**

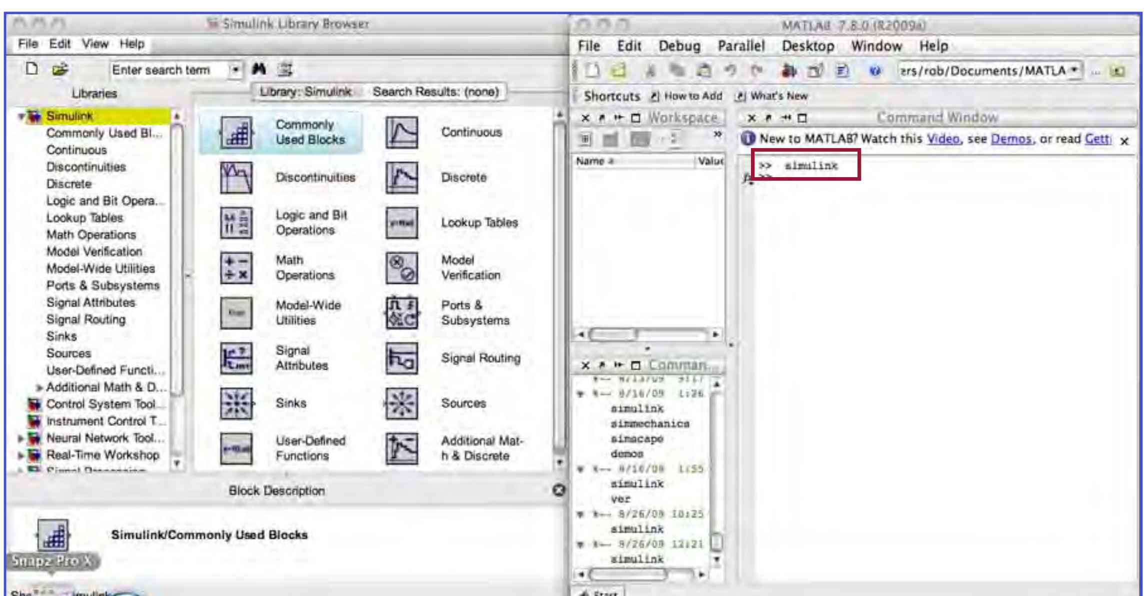




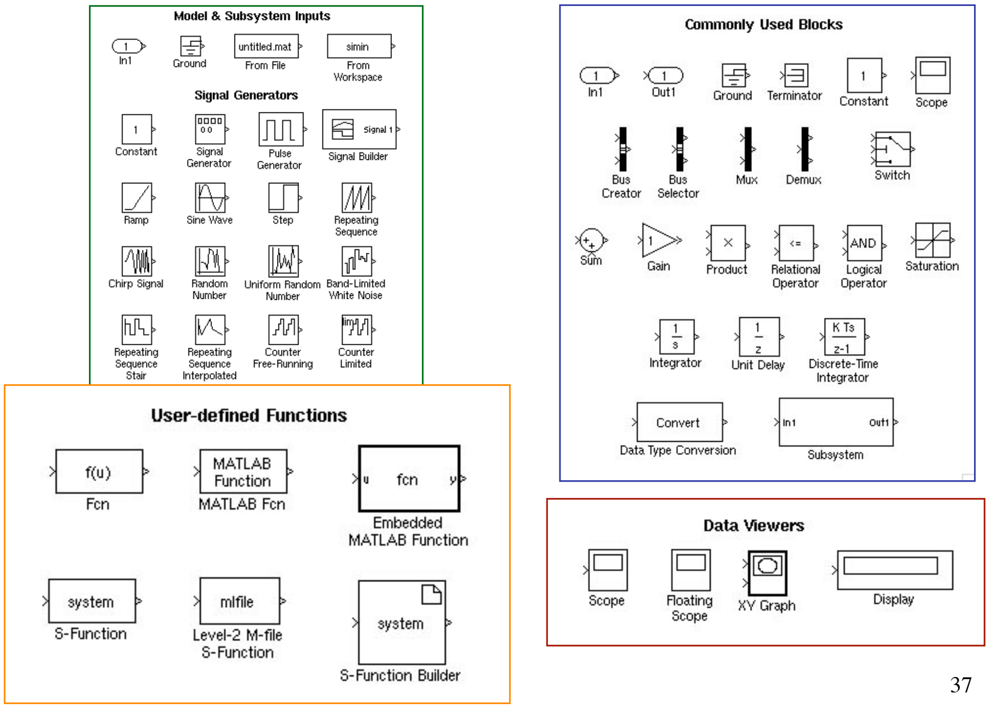
More on Simulink

Simulink

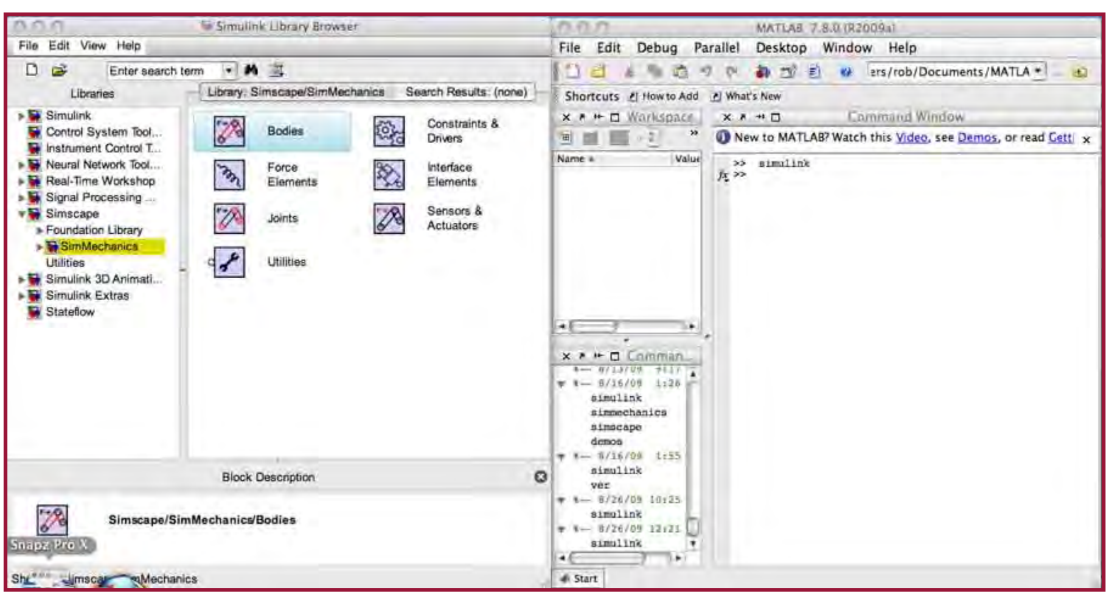
Library of blocks, sources, and sinks



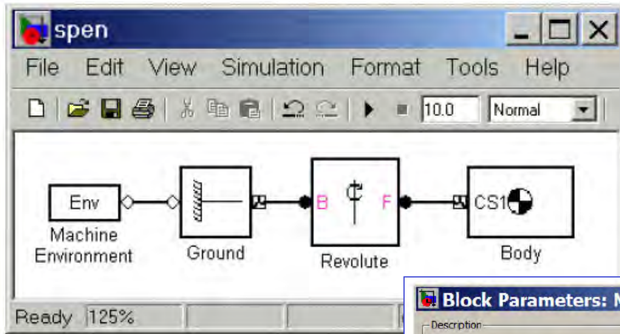
Simulink Blocks

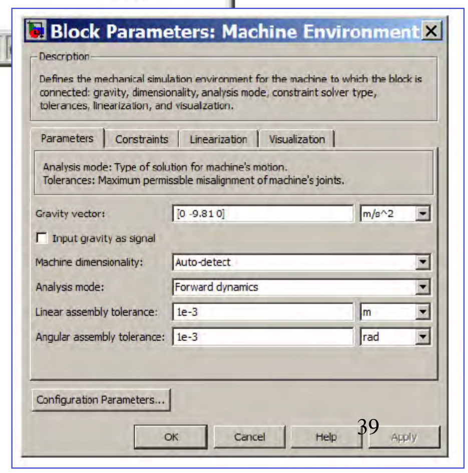


SimMechanics
Called from Simulink

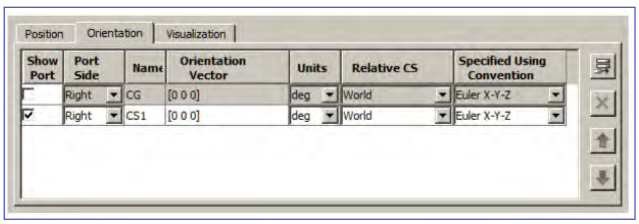


Simple Pendulum



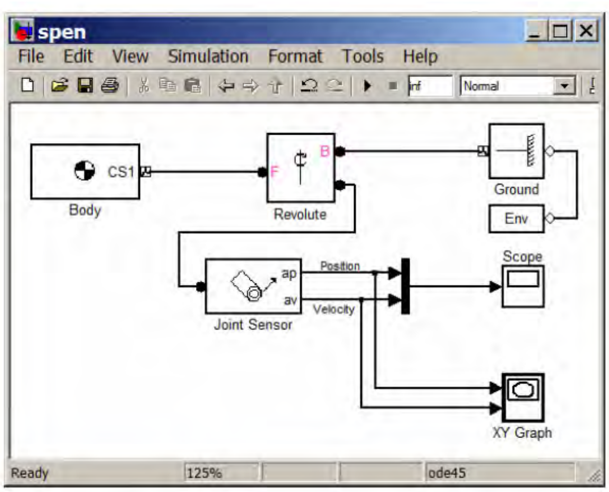


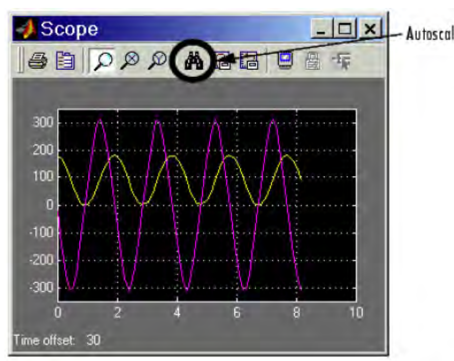
Specifying Body Coordinate System



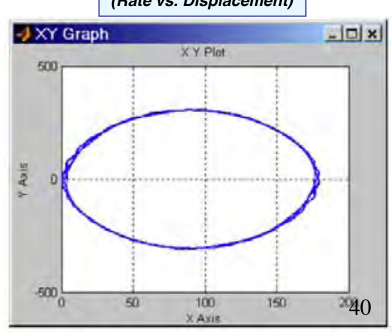
39

Simple Pendulum with Scope and XY Graph

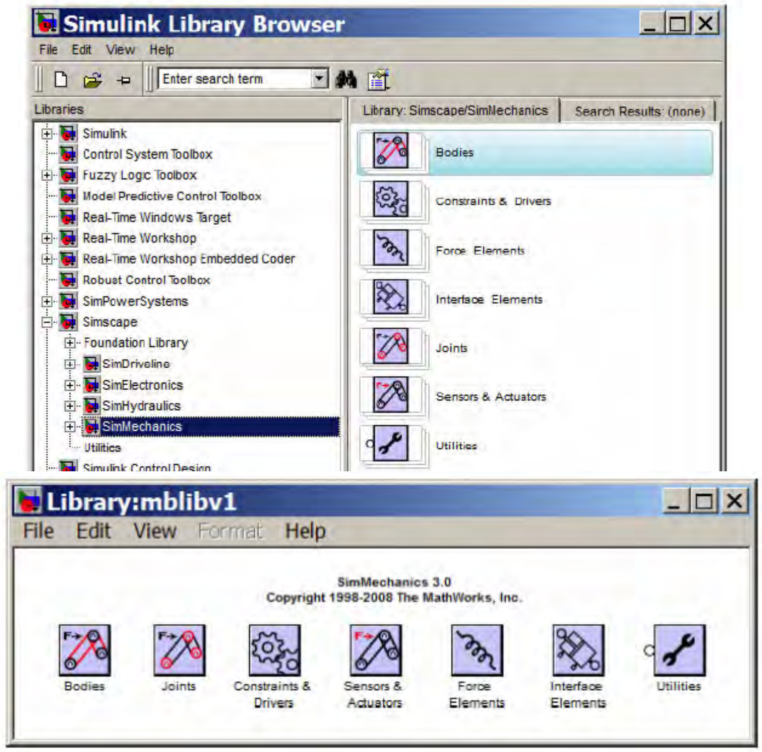




Phase-Plane Plot (Rate vs. Displacement)



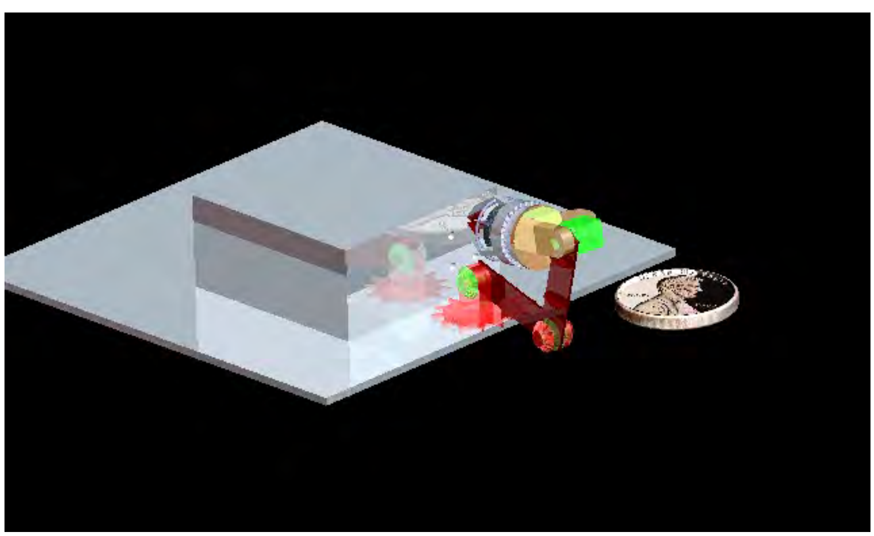
40



More examples in *Supplemental Material*

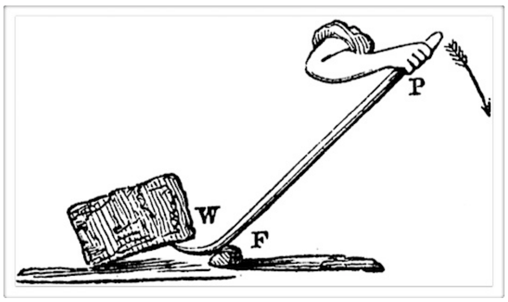
Simulink Demonstration of 1-Inch Robot

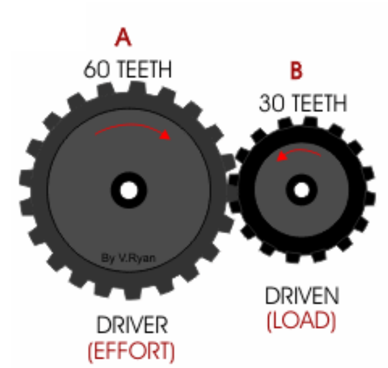
(MAE 345 Mid-Term Project, 2009)



Gearing and Leverage

Force multiplication
Displacement ratios

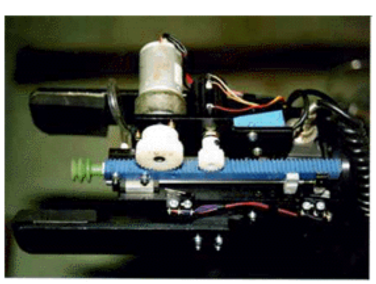


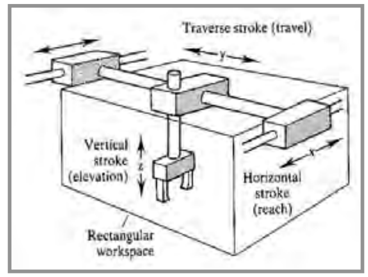


End Effectors

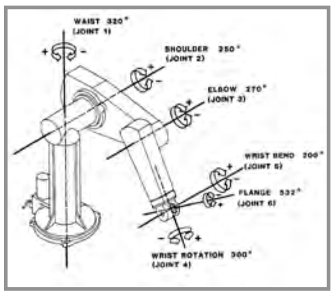
- **Machine tools**
  - Grinding, sanding
  - Inserting screws
  - Drilling
  - Hammering
- **Paint sprayer**
- **Gripper, clamp**
- **Multi-digit hand**

DARPA Prosthetic Hand
http://www.youtube.com/watch?v=QJg9igTnjlo&feature=related



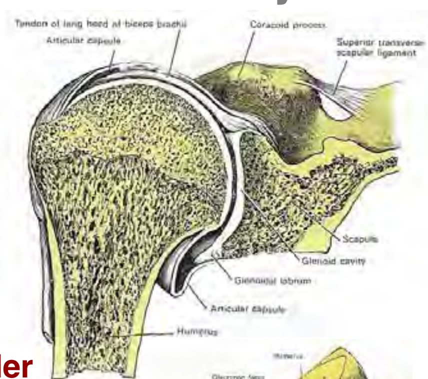


Links and Joints

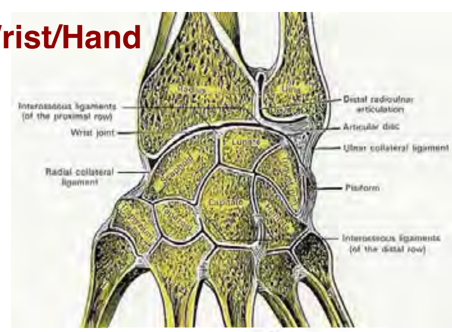


Human Joints

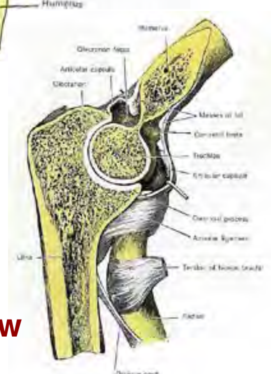
Gray's Anatomy, 1858



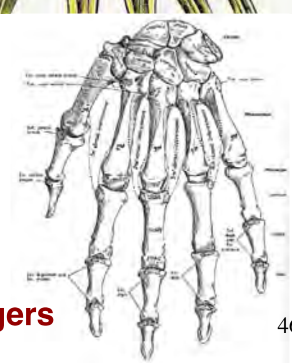
Shoulder



Wrist/Hand

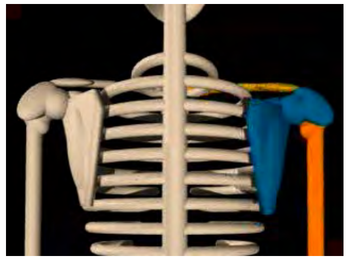


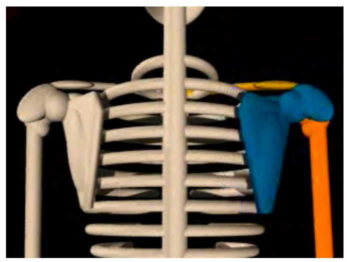
Elbow

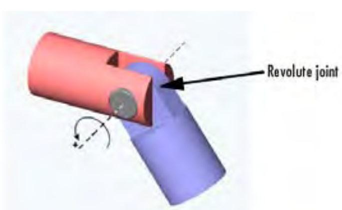


Hand/Fingers

Skeleton and Muscle-Induced Motion



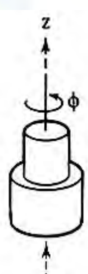


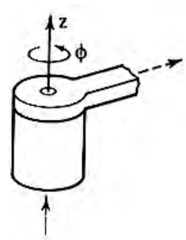


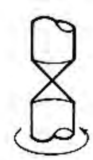
Revolute Robotic Joints

Rotation about a single axis

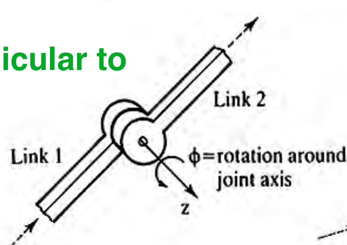
Parallel to Link

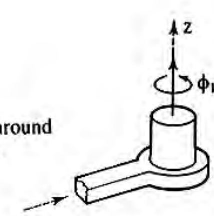


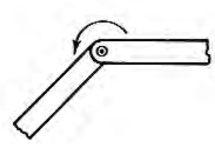




Perpendicular to Link

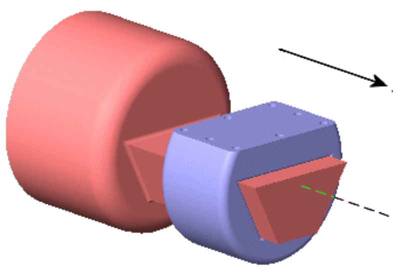


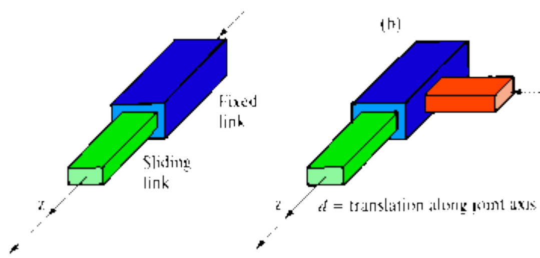




Prismatic Robotic Joints

Sliding along a single axis

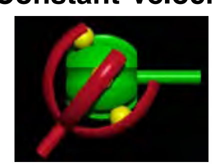




Universal

Other Robotic Joints

Constant-Velocity

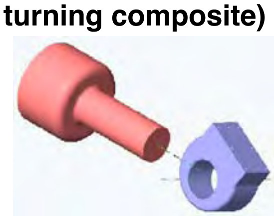


Flexible

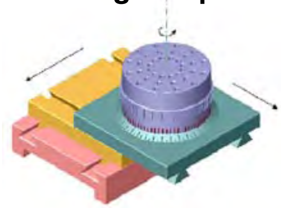
Spherical (or ball)

Roller Screw

Cylindrical (sliding and turning composite)



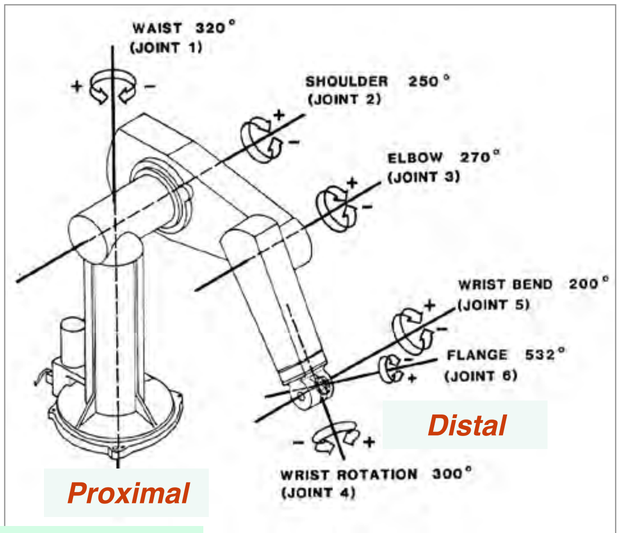
Planar (sliding and turning composite)



Characteristic Transformation of a Link

Link: solid structure between two joints

- **Each link type has a characteristic transformation matrix relating the proximal joint to the distal joint**
- **Link *n* has**
  - **Proximal end: Joint *n*, coordinate frame *n* – 1**
  - ***Distal end*: Joint *n* + 1, coordinate frame *n***



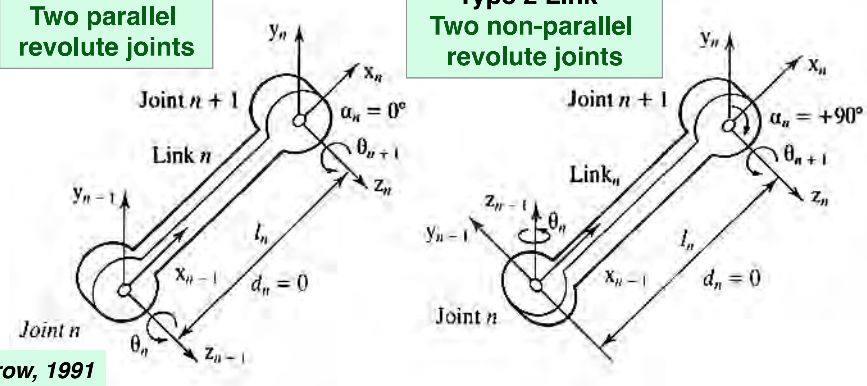
McKerrow, 1991

Links Between Revolute Joints

- **Link: solid structure between two joints**
  - **Proximal end:** closer to the base
  - **Distal end:** farther from the base
- **4 Link Parameters**
  - **Length of the link** between rotational axes, *l*, along the common normal
  - **Twist angle** between axes, α
  - **Angle between** 2 links, θ (revolute)
  - **Offset between** links, *d* (prismatic)
- **Joint Variable: single link parameter that is free to vary**

Type 1 Link
Two parallel revolute joints

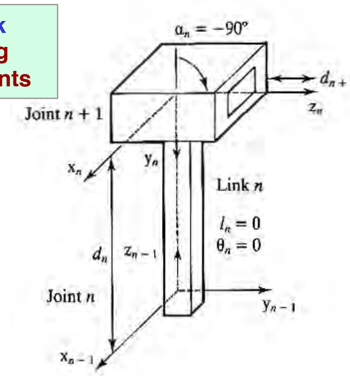
Type 2 Link
Two non-parallel revolute joints



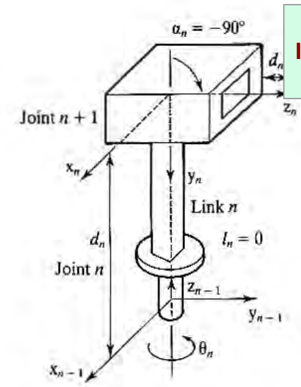
McKerrow, 1991

Links Involving Prismatic Joints

Type 5 Link
Intersecting prismatic joints



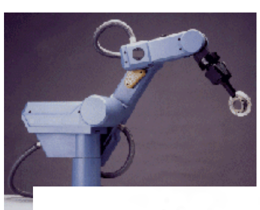
Type 6 Link
Intersecting revolute and prismatic joints



- Link n extends along z_{n-1} axis
  - $l_n = 0$, along x_{n-1}
  - d_n = length, along z_{n-1} (variable)
  - $\theta_n = 0$, about z_{n-1}
  - α_n = fixed orientation of $n + 1$ prismatic axis about x_{n-1}

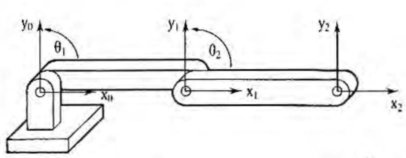
- Link n extends along z_{n-1} axis
  - $l_n = 0$, along x_{n-1}
  - d_n = length, along z_{n-1} (fixed)
  - θ_n = variable joint angle n about z_{n-1}
  - α_n = fixed orientation of $n + 1$ prismatic axis about x_{n-1}

McKerrow, 1991

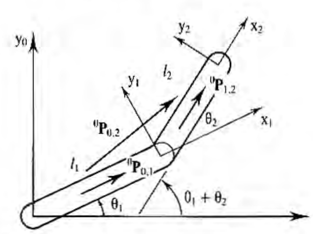


Two-Link/Three-Joint Manipulator

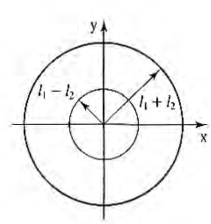
Parallel Rotation Axes



Manipulator in zero position



Assignment of coordinate frames



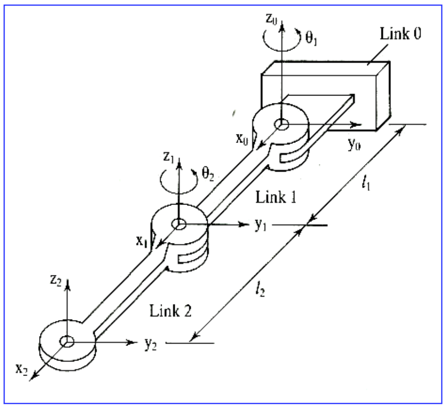
Workspace

Parameters and Variables for 2-link manipulator
- Link lengths (fixed)
- Joint angles (variable)

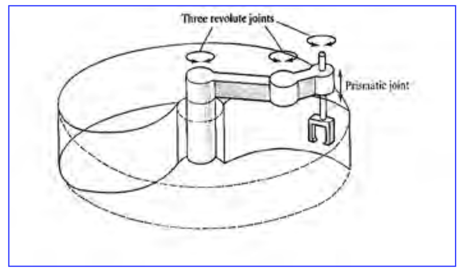
McKerrow, 1991

Four-Joint (SCARA*) Manipulator

Arm with Three Revolute Link Variables (Joint Angles)



Operation
http://www.youtube.com/watch?v=3-sbtCCyJXo



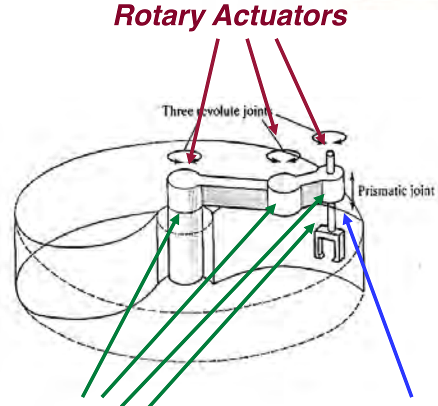
McKerrow, 1991

**Selective Compliant Articulated Robot Arm*

Joint Variables Must Be Actuated and Observed for Control

- **Frames of Reference for Actuation and Control**
  - World coordinates
  - Actuator coordinates
  - Joint coordinates
  - Tool coordinates

Rotary Actuators



Sensors May Observe Joints Directly, Indirectly, or Not At All

Linear Actuator

Next Time: Transformations, Trajectories, and Path Planning

Supplemental Material

By-Passing the Euler Angle Singularity at $\theta = \pm 90°$

Replace Euler angles as primary definition of angular attitude

More than 3 angle parameters are necessary to avoid singularity

Two alternatives to Euler angles

Direction cosine (rotation) matrix [9 parameters]

Quaternions [4 parameters]

(See *Supplemental Material*)

Propagating the Rotation Matrix without Euler Angles [9 Parameters]

Recall that $\dot{\mathbf{H}}_B^I \mathbf{h}_B = \tilde{\boldsymbol{\omega}}_I \mathbf{H}_B^I \mathbf{h}_B$

$$\therefore \dot{\mathbf{H}}_B^I = \tilde{\boldsymbol{\omega}}_I \mathbf{H}_B^I$$

Consequently, $\dot{\mathbf{H}}_I^B(t) = -\tilde{\boldsymbol{\omega}}_B(t)\mathbf{H}_I^B(t)$

$$= -\begin{bmatrix} 0 & -r(t) & q(t) \\ r(t) & 0 & -p(t) \\ -q(t) & p(t) & 0(t) \end{bmatrix}_B \mathbf{H}_I^B(t)$$

Retrieving Euler Angles from the Rotation Matrix

$$\mathbf{H}_I^B = \left[\begin{array}{c|c|c} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ \hline -\cos\phi\sin\psi + \sin\phi\sin\theta\cos\psi & \cos\phi\cos\psi + \sin\phi\sin\theta\sin\psi & \sin\phi\cos\theta \\ \hline \sin\phi\sin\psi + \cos\phi\sin\theta\cos\psi & -\sin\phi\cos\psi + \cos\phi\sin\theta\sin\psi & \cos\phi\cos\theta \end{array}\right] = \left(\mathbf{H}_B^I\right)^T$$

Trigonometry: for example, for *$\theta \neq \pm \pi/2$*

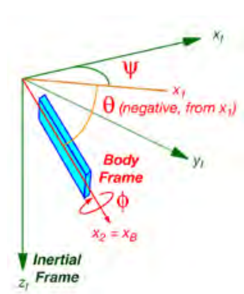
$$\boxed{\begin{aligned} \theta &= -\sin^{-1} h_{1,3} \\ \psi &= -\sin^{-1}\left(h_{1,2}/\cos\theta\right) \\ \phi &= -\sin^{-1}\left(h_{2,3}/\cos\theta\right) \end{aligned}}$$

Ambiguity between *ψ* and *Φ* remains for *$\theta = \pm \pi/2$*

Quaternion Vector

Single rotation from inertial to body frame (4 parameters)

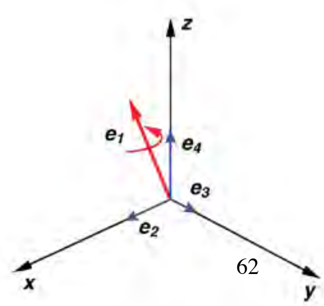
$$\begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \end{bmatrix} = \begin{bmatrix} \text{Rotation angle, } \theta \text{, rad} \\ x\text{-component of rotation axis} \\ y\text{-component of rotation axis} \\ z\text{-component of rotation axis} \end{bmatrix} = \begin{bmatrix} \cos\dfrac{\theta}{2} \\ \sin\dfrac{\theta}{2}\begin{bmatrix} x_R \\ y_R \\ z_R \end{bmatrix} \end{bmatrix}$$



Normality conditions

$$\boxed{e_1^2 + e_2^2 + e_3^2 + e_4^2 = 1}$$

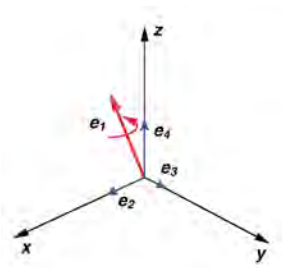
$$\boxed{x_R^2 + y_R^2 + z_R^2 = 1}$$



Quaternion Vector Initialized from Non-Singular Euler Angles

$$\begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \end{bmatrix} = \begin{bmatrix} \cos\frac{\phi}{2}\cos\frac{\theta}{2}\cos\frac{\psi}{2} + \sin\frac{\phi}{2}\sin\frac{\theta}{2}\sin\frac{\psi}{2} \\ \sin\frac{\phi}{2}\cos\frac{\theta}{2}\cos\frac{\psi}{2} - \cos\frac{\phi}{2}\sin\frac{\theta}{2}\sin\frac{\psi}{2} \\ \cos\frac{\phi}{2}\sin\frac{\theta}{2}\cos\frac{\psi}{2} + \sin\frac{\phi}{2}\cos\frac{\theta}{2}\sin\frac{\psi}{2} \\ \cos\frac{\phi}{2}\cos\frac{\theta}{2}\sin\frac{\psi}{2} - \sin\frac{\phi}{2}\sin\frac{\theta}{2}\cos\frac{\psi}{2} \end{bmatrix}$$

Propagating the Quaternion Vector



$$\dot{\mathbf{e}}(t) = \begin{bmatrix} \dot{e}_1(t) \\ \dot{e}_2(t) \\ \dot{e}_3(t) \\ \dot{e}_4(t) \end{bmatrix} = \frac{1}{2}\mathbf{Q}(t)\mathbf{e}(t)$$

$$= \frac{1}{2}\begin{bmatrix} 0 & -p(t) & -q(t) & -r(t) \\ p(t) & 0 & r(t) & -q(t) \\ q(t) & -r(t) & 0 & p(t) \\ r(t) & q(t) & -p(t) & 0 \end{bmatrix}\begin{bmatrix} e_1(t) \\ e_2(t) \\ e_3(t) \\ e_4(t) \end{bmatrix}$$

Then compute rotation matrix from quaternion

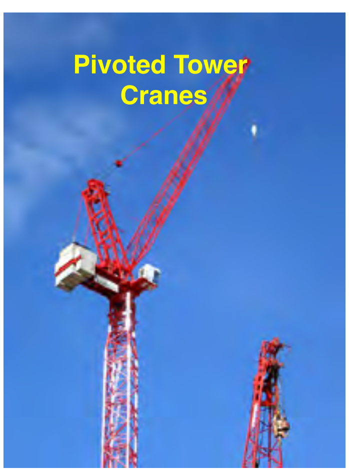
$$\mathbf{H}_I^B = \begin{bmatrix} e_1^2 + e_2^2 - e_3^2 - e_4^2 & 2(e_2e_3 - e_1e_4) & 2(e_2e_4 + e_1e_3) \\ 2(e_2e_3 + e_1e_4) & e_1^2 - e_2^2 + e_3^2 - e_4^2 & 2(e_3e_4 - e_1e_2) \\ 2(e_2e_4 - e_1e_3) & 2(e_3e_4 + e_1e_2) & e_1^2 - e_2^2 - e_3^2 + e_4^2 \end{bmatrix} = \left(\mathbf{H}_B^I\right)^T$$

Euler Angles Retrieved from Quaternion Vector

$$\begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} \text{atan2}\left[2\left(e_1 e_2 + e_3 e_4\right), \left[1 - 2\left(e_2^2 + e_3^2\right)\right]\right] \\ \arcsin\left[2\left(e_1 e_3 - e_2 e_4\right)\right] \\ \text{atan2}\left[2\left(e_0 e_3 + e_1 e_2\right), \left[1 - 2\left(e_3^2 + e_4^2\right)\right]\right] \end{bmatrix}$$

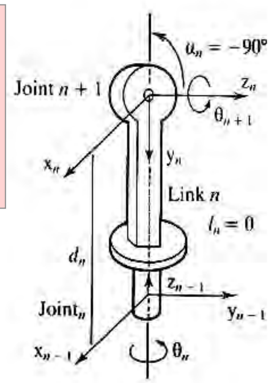
$$\text{atan2}(y,x) \triangleq 2 \arctan \frac{y}{\sqrt{x^2 + y^2} + x}$$

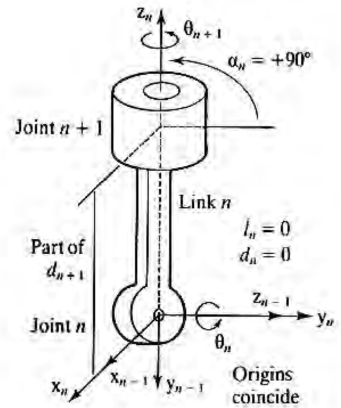
Construction Cranes



Links Between Revolute Joints - 2

Type 3 Link
Two revolute joints with intersecting rotational axes (e.g., shoulder)





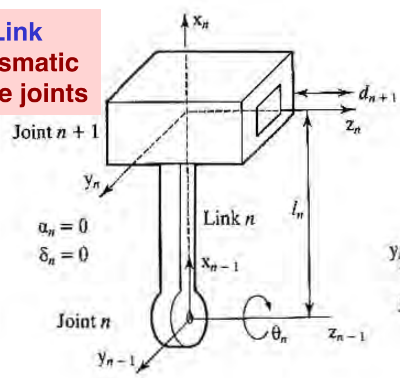
Type 4 Link
Two perpendicular revolute joints with common origin (e.g., elbow-wrist)

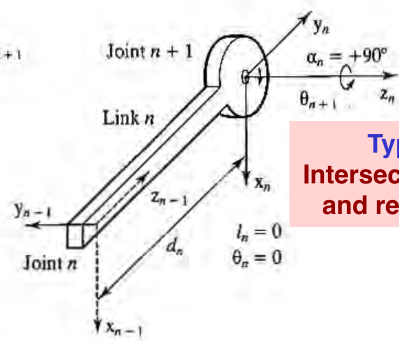
- Link n extends along z_{n-1} axis
  - $l_n = 0$, along x_{n-1}
  - d_n = length, along z_{n-1} (fixed)
  - θ_n = variable joint angle n about z_{n-1}
  - α_n = fixed orientation of $n + 1$ rotational axis about x_{n-1}
- Link n extends along $-z_n$ axis
  - $l_n = 0$, along x_{n-1}
  - $d_n = 0$, along z_{n-1}
  - θ_n = variable joint angle n about z_{n-1}
  - α_n = fixed orientation of $n + 1$ rotational axis about x_{n-1}

McKerrow, 1991

Links Involving Prismatic Joints - 2

Type 7 Link
Parallel prismatic and revolute joints



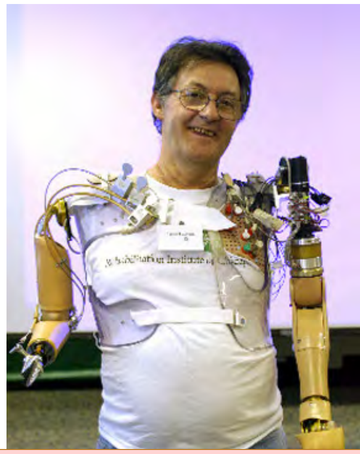


Type 8 Link
Intersecting prismatic and revolute joints

- Link n extends along x_{n-1} axis
  - l_n = length along x_{n-1}
  - $d_n = 0$, along z_{n-1}
  - θ_n = variable joint angle n about z_{n-1}
  - $\alpha_n = 0$, orientation of $n + 1$ prismatic axis about x_{n-1}
- Link n extends along z_{n-1} axis
  - $l_n = 0$, along x_{n-1}
  - d_n = length, along z_{n-1} (variable)
  - $\theta_n = 0$, about z_{n-1}
  - α_n = fixed orientation of $n + 1$ rotational axis about x_{n-1}

McKerrow, 1991

Prosthetic Arms and Hands



Jesse Sullivan, 2007
Rehabilitation Institute of Chicago

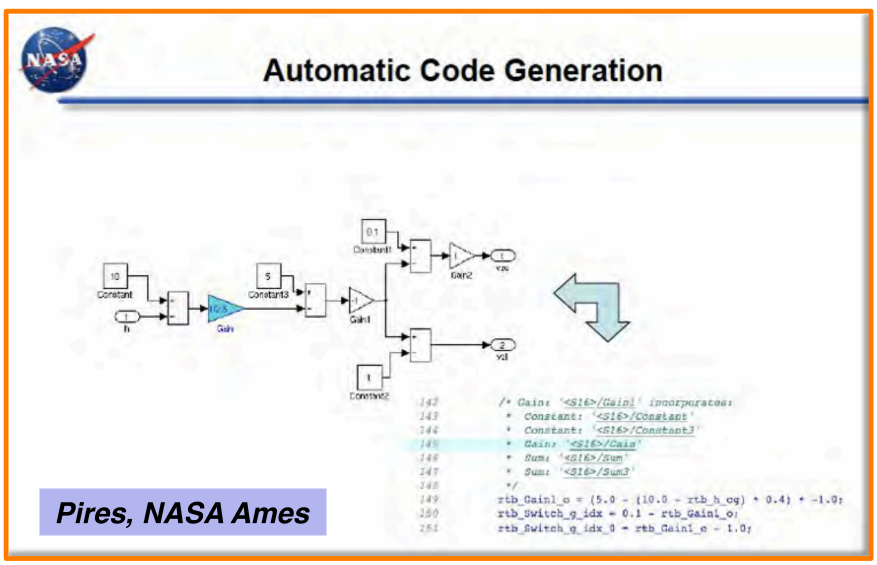
Jesse with the DEKA/DARPA Arm, 2009
http://www.youtube.com/watch?v=ddInW6sm7JE

"Terminator Arm"
https://www.youtube.com/watch?v=_qUPnnROxvY

Open Bionics 3D-Printed Hand
http://techcrunch.com/2015/09/02/open-bionics-wants-to-bring-down-the-cost-of-prosthetics-with-3d-printed-robotic-hands/

Toward the Bionic Man
https://www.youtube.com/watch?v=xBiOQKonkWs

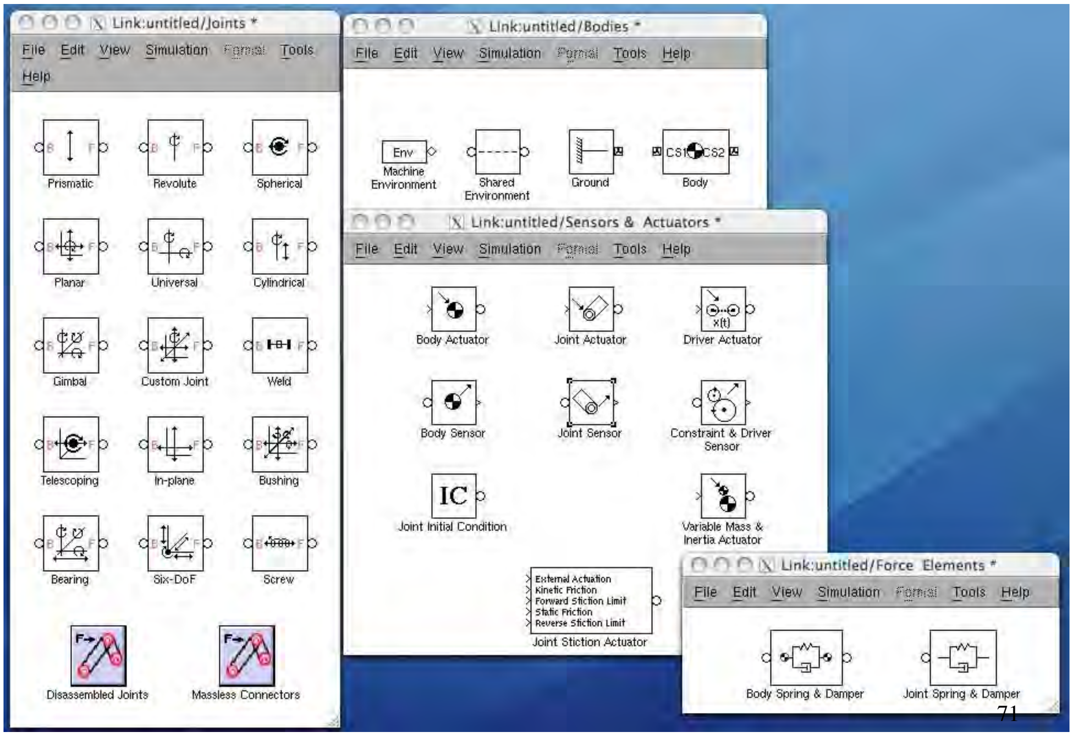
Simulink



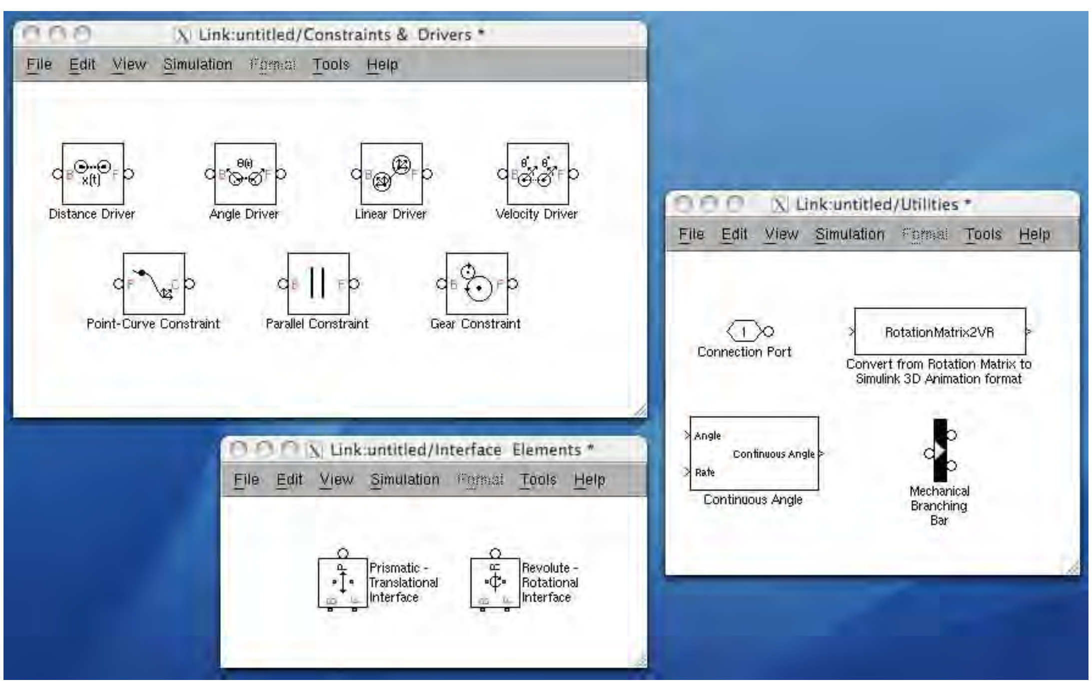
Pires, NASA Ames

Graphic modeling of dynamic systems
Library of functions
Generation of MATLAB code

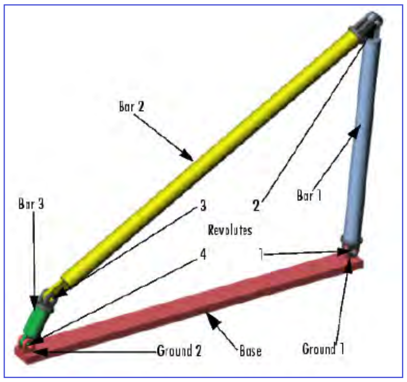
SimMechanics Library - 1

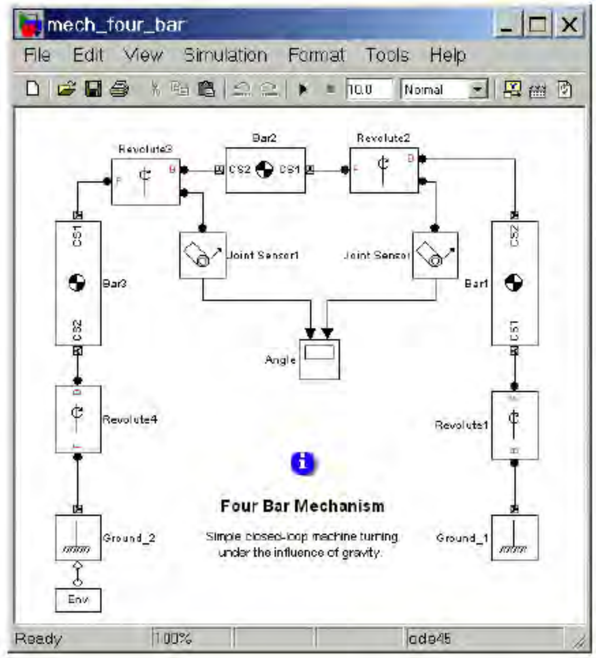


SimMechanics Library - 2



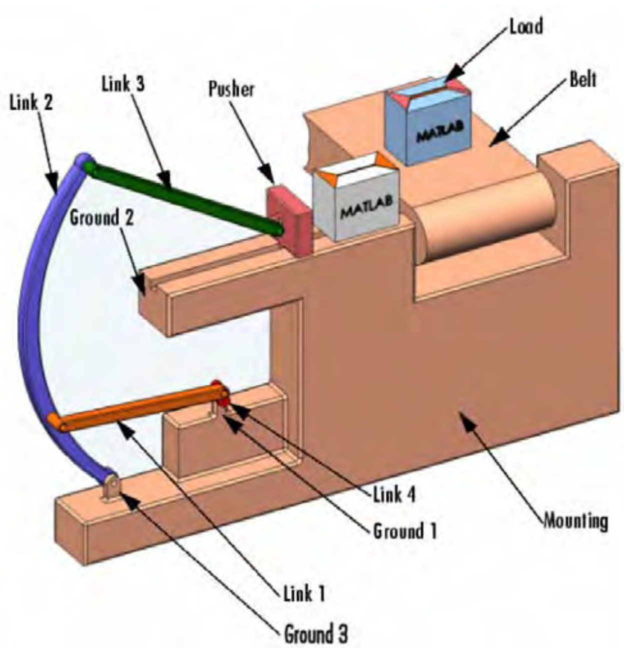
Simulink/SimMechanics Representation of Four-Bar Linkage





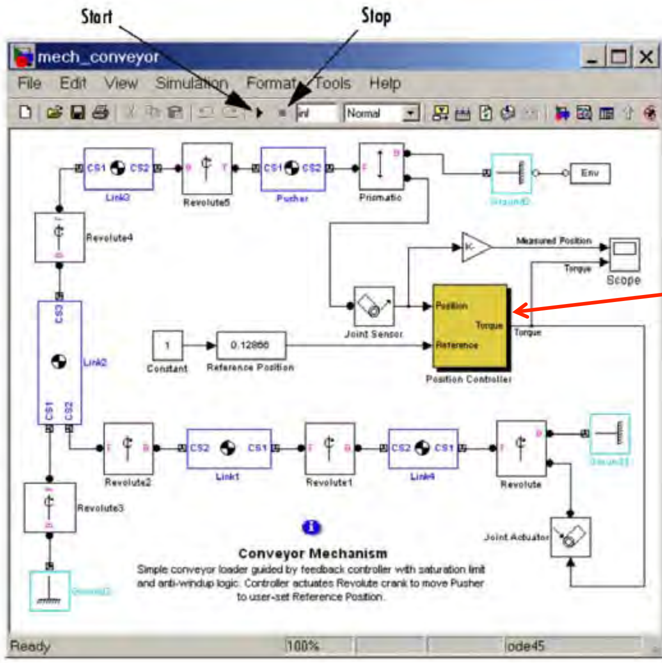
73

Conveyer-Loader Demonstration



74

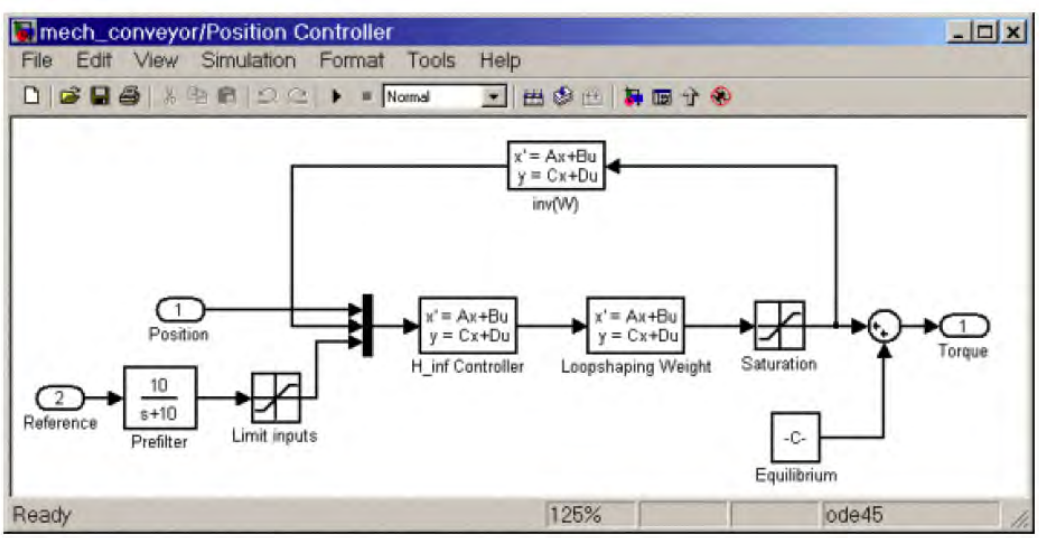
Conveyer-Loader Demonstration

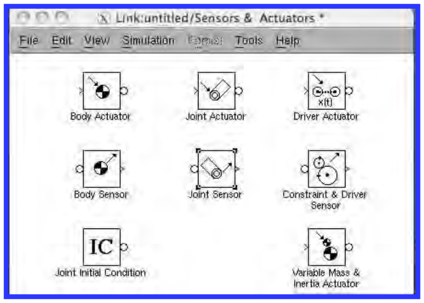


Controller specified within box (See *Supplemental Material*)

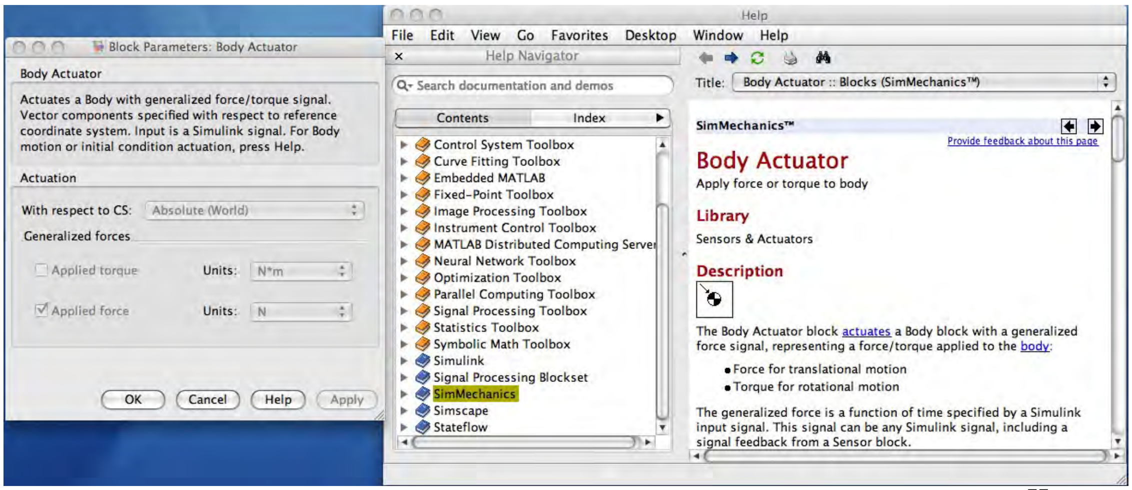
Position Controller for Conveyor-Loader Demonstration

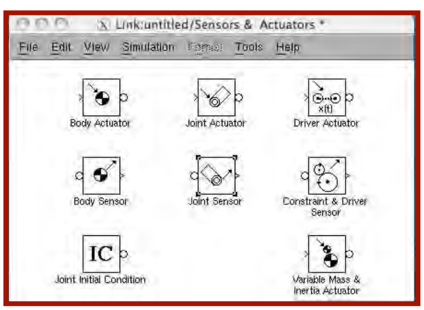
(Simulink)



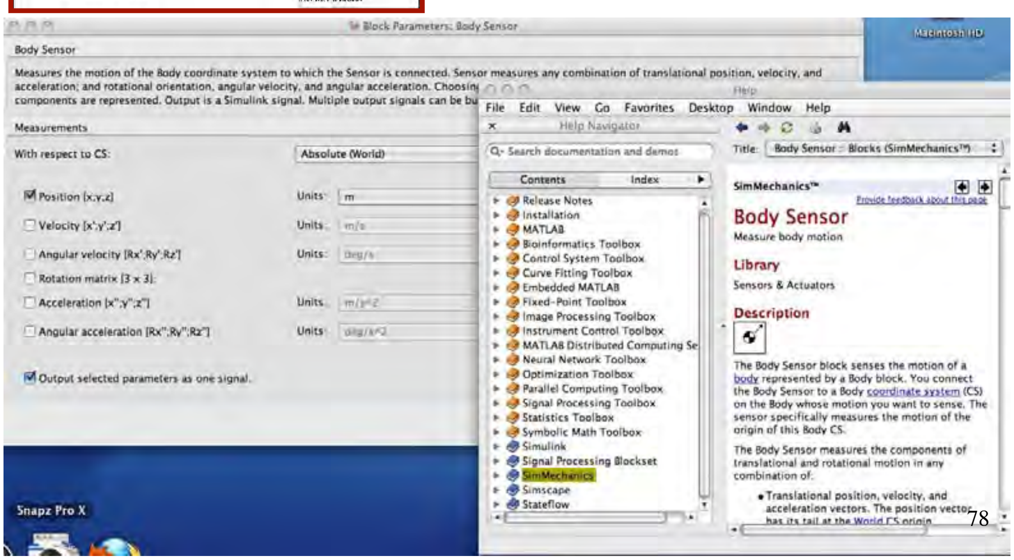


SimMechanics Body Actuator

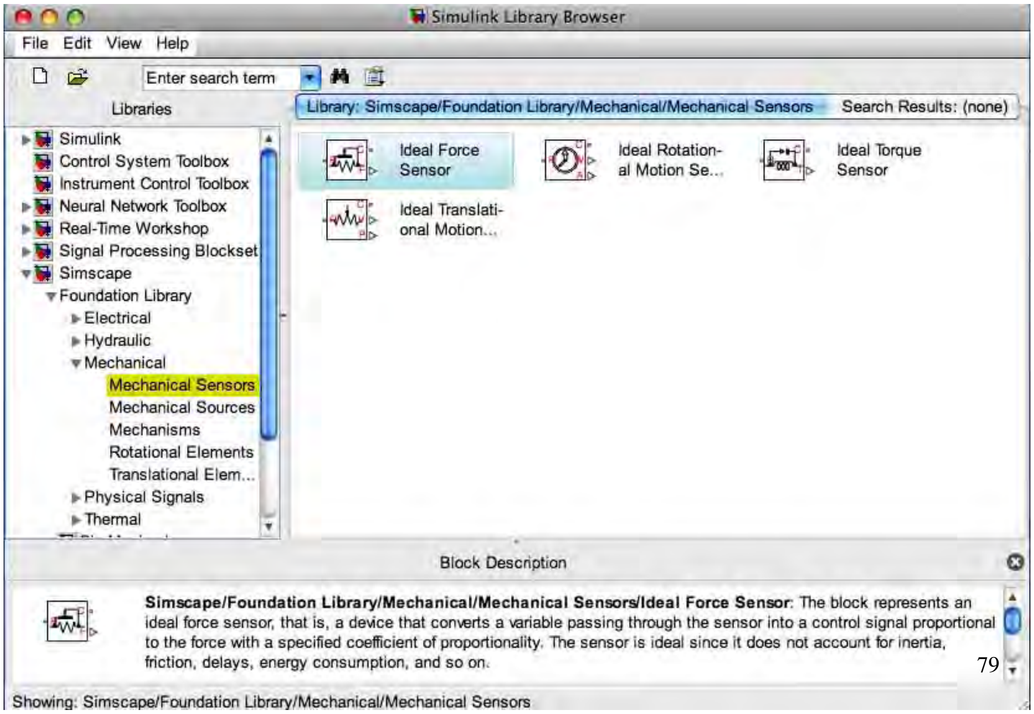




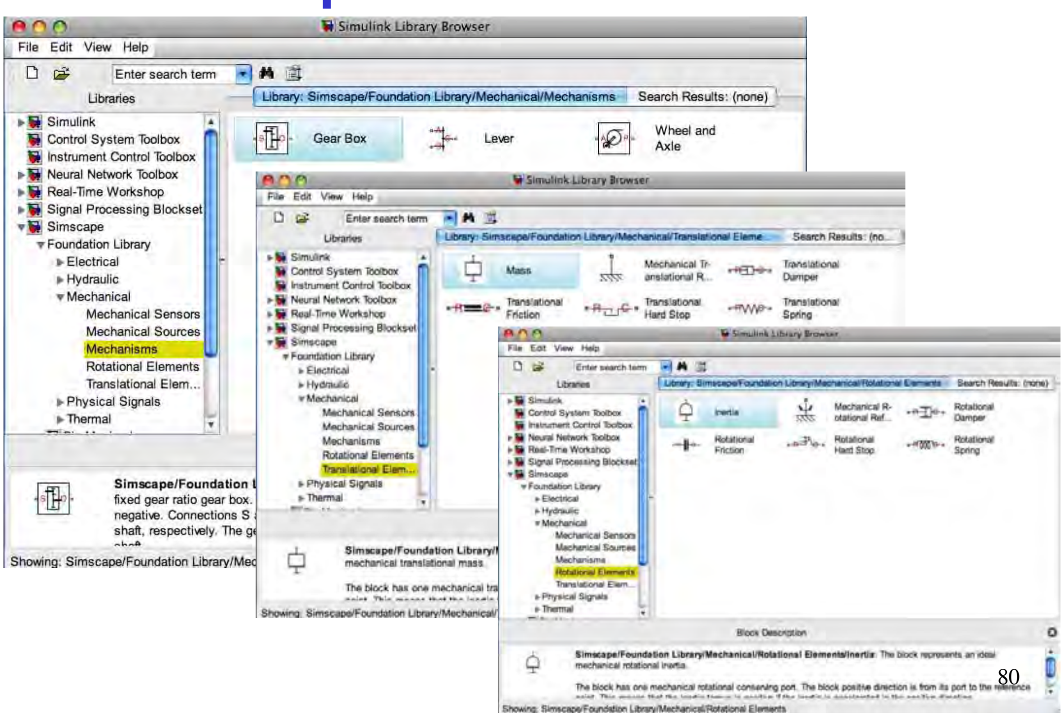
SimMechanics Body Sensor



SimScape Mechanical Sensors

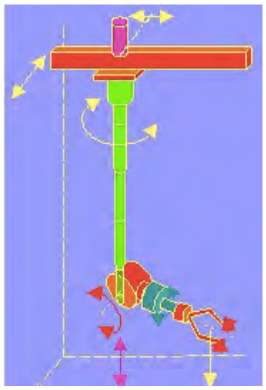


SimScape Mechanism Models

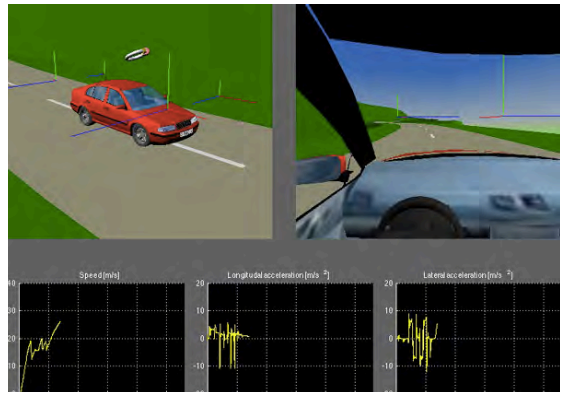


SimMechanics, Simulink 3D Animation 'Product Help' Demos

Robotic Manipulator



Vehicle Dynamics



http://www.mathworks.com/products/simmechanics/demos.html