Robot Dynamics: Euler-Lagrange Formulation

Prof. S.K. Saha

Department of Mechanical Engineering

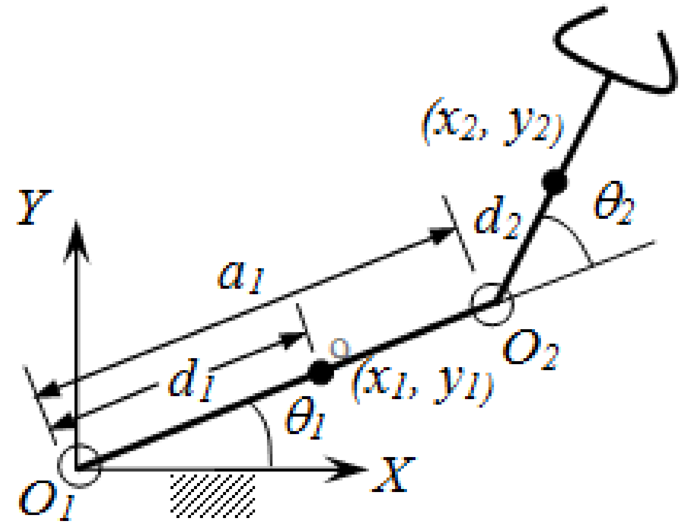
IIT Delhi

Outline

- Generalized coordinates
- Kinetic and potential energy
- Equations of motion
- Inverse and Forward dynamics

Generalized Coordinates and Forces

- Independent coordinates that specify the configuration, i.e., the position and orientation, of all the bodies or links of a robot manipulator completely are called generalized coordinates



- Generalized coordinates can have several representations, e.g., Cartesian (x, y of a point), Polar (r, θ of a point), relative, etc.
- Generalized forces are those which cause change in generalized coordinates, e.g., force to change x or y, *or* moment change θ.

Illustration

- The 2-link planar robot arm requires 6 coordinates, i.e., (x_1, y_1, θ_1) and (x_2, y_2, θ_2)
- Note: (x_1, y_1) and (x_2, y_2) define the mass centers of the links, whereas θ_1 and θ_2 denote their orientations
- d_1 and d_2 are mass center locations from the joints
- 6-coordinates are not independent, as 2-joints restrict the motion of the links

Constraints and Independent Coordinates

- Four constraints

$$x_1 = d_1 \cos\theta_1$$
$$y_1 = d_1 \sin\theta_1$$
$$x_2 = a_1 \cos\theta_1 + d_2 \cos\theta_{12}$$
$$y_2 = a_1 \sin\theta_1 + d_2 \sin\theta_{12}$$

where $\theta_{12} \equiv \theta_1 + \theta_2$

- Needs only 6–4 = 2 independent coordinates (θ_1 and θ_2)

Generalized Coordinates vs. DOF

- The independent coordinates are “generalized coordinates”
- Number of generalized coordinates is equal to Degrees-of-freedom (DOF)
- The planar robot arm has 2-DOF
- Any two can be generalized coordinates
- In robots, θ_1 and θ_2 are most convenient as they are actuator-controlled

Spatial Robots

- Each rigid link requires 6 coordinates to specify its position and orientation
- If a robot manipulator has m moving links, it will require $6m$ coordinates
- Links cannot move freely as joints put restrictions
- If c constraints exist, then n (DOF) = $6m - c$
- With 6 moving links with revolute or prismatic joints: $n = 6 \times 6 - 6 \times 5 = 6$

Euler-Lagrange Formulation

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} = \phi_i$$

L (Lagrangian) = $T - U$;

T: Kinetic energy; U: Potential energy;

q_i: Generalized coordinate;

ϕ_i : Generalized force.

Kinetic and Potential Energies

- Kinetic Energy

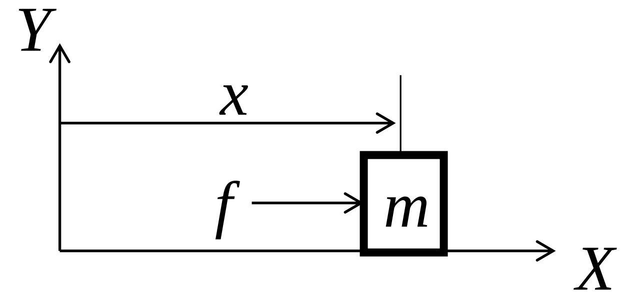
$$T = \sum_{i=1}^{n} T_i = \sum_{i=1}^{n} \frac{1}{2}\left(m_i \dot{\mathbf{c}}_i^T \dot{\mathbf{c}}_i + \boldsymbol{\omega}_i^T \mathbf{I}_i \boldsymbol{\omega}_i \right)$$

- Potential Energy

$$U = -\sum_{i=1}^{n} m_i \mathbf{c}_i^T \mathbf{g}$$

Illustration: A Moving Mass

- Generalized Coordinate: x
- Kinetic Energy: $T = \frac{1}{2} m \dot{x}^2$
- Potential Energy: $U = 0$
- Lagrangian: $L = T - U$



$$\frac{\partial L}{\partial \dot{x}} = m\dot{x}; \frac{\partial L}{\partial x} = 0 \qquad \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right) = m\ddot{x}$$

- Generalized Force: f

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right) - \frac{\partial L}{\partial x} = f \qquad \boxed{m\ddot{x} = f}$$

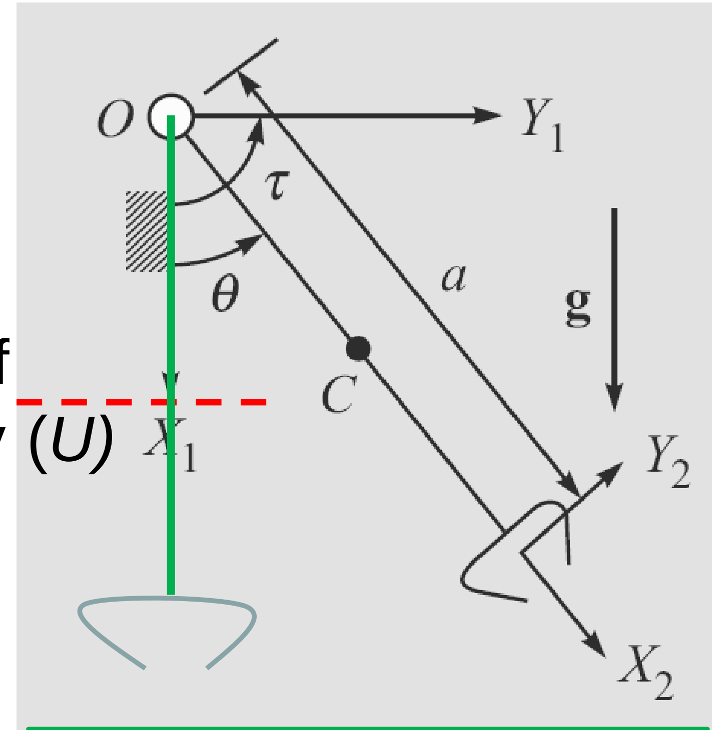
(Dynamic)
Equation of Motion
or
Dynamic Model

One-DOF Arm

$$T \equiv \frac{1}{2}m(\frac{a}{2}\dot{\theta})^2 + \frac{1}{2}\frac{ma^2}{12}\dot{\theta}^2;$$

$$U = mg(\frac{a}{2} - \frac{a}{2}c\theta)$$

Reference of Potential Energy (*U*)



$$\boxed{L = T - U \equiv \frac{ma^2}{6}\dot{\theta}^2 - mg\frac{a}{2}(1 - c\theta)}$$

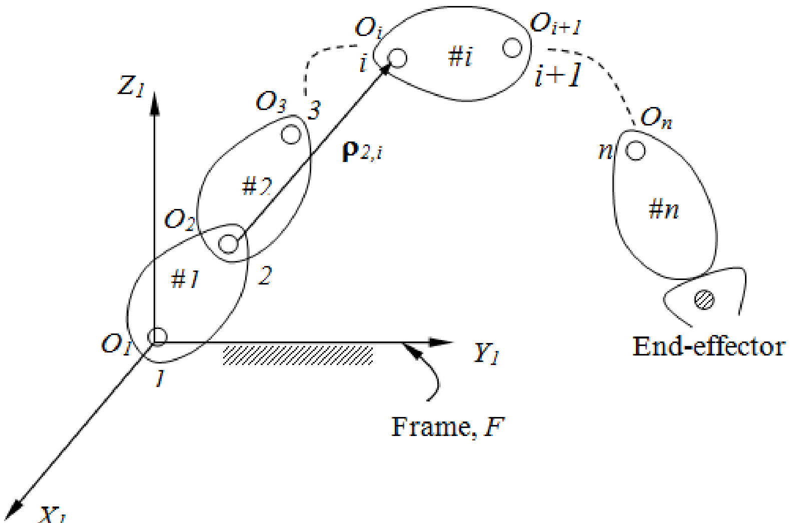
$$\frac{d}{dt}(\frac{\partial L}{\partial \dot{\theta}}) = \frac{1}{3}ma^2\ddot{\theta}; \frac{\partial L}{\partial \theta} = -\frac{1}{2}mgas\theta;$$

$$\boxed{\frac{1}{3}ma^2\ddot{\theta} + \frac{1}{2}mgas\theta = \tau}$$

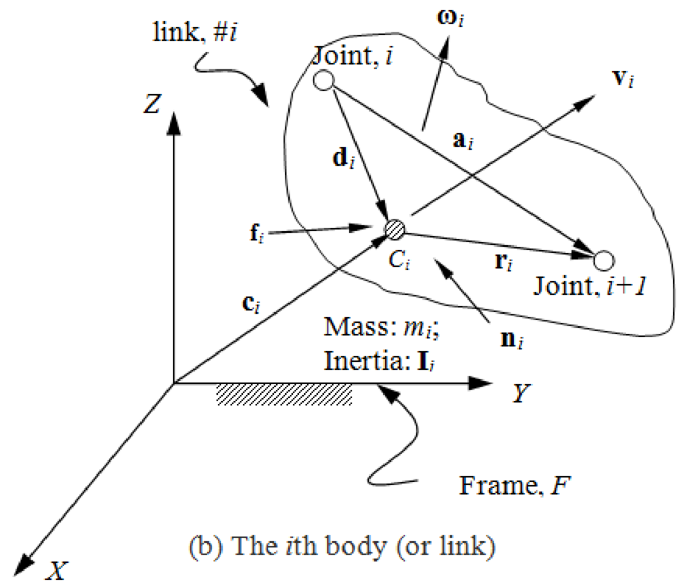
Alternatively,

$$\boxed{U = -\sum_{i=1}^{n} m_i \mathbf{c}_i^T \mathbf{g} = -m\begin{bmatrix} \frac{a}{2}c\theta & \frac{a}{2}s\theta & 0 \end{bmatrix}\begin{bmatrix} g \\ 0 \\ 0 \end{bmatrix} = -mg\frac{a}{2}c\theta}$$

Velocities



(a) A serial-chain robot manipulator



(b) The ith body (or link)

$$\boldsymbol{\omega}_i = \boldsymbol{\omega}_{i-1} + \dot{\theta}_i \mathbf{e}_i = \mathbf{e}_1 \dot{\theta}_1 + \mathbf{e}_2 \dot{\theta}_2 + \ldots + \mathbf{e}_i \dot{\theta}_i$$

$$\dot{\mathbf{c}}_i = \dot{\mathbf{c}}_{i-1} + \boldsymbol{\omega}_{i-1} \times \mathbf{r}_{i-1} + \boldsymbol{\omega}_i \times \mathbf{d}_i = \mathbf{e}_1 \times \boldsymbol{\rho}_{1i}\, \dot{\theta}_1 + \cdots + \mathbf{e}_i \times \boldsymbol{\rho}_{ii}\, \dot{\theta}_i$$

Velocities of the i[th] link in terms of all the n joint rates as

$$\boldsymbol{\omega}_i = \mathbf{J}_{\omega,i}\dot{\boldsymbol{\theta}} \qquad \text{where} \qquad \mathbf{J}_{\omega,i} \equiv [\mathbf{j}_{\omega,\mathrm{i}}^{(1)}\ \mathbf{j}_{\omega,\mathrm{i}}^{(2)} \cdots \mathbf{j}_{\omega,\mathrm{i}}^{(\mathrm{i})}\ \mathbf{0} \ldots \mathbf{0}]$$

$$\dot{\mathbf{c}}_i = \mathbf{J}_{c,i}\dot{\boldsymbol{\theta}} \qquad \text{where} \qquad \mathbf{J}_{c,\mathrm{i}} \equiv [\mathbf{j}_{c,\mathrm{i}}^{(1)}\ \mathbf{j}_{c,\mathrm{i}}^{(2)} \cdots \mathbf{j}_{c,\mathrm{i}}^{(\mathrm{i})}\ \mathbf{0} \ldots \mathbf{0}]$$

$$\boxed{T = \frac{1}{2}\sum_{i=1}^{n}(m_i \dot{\mathbf{c}}_i^{\mathrm{T}} \dot{\mathbf{c}}_i + \boldsymbol{\omega}_i^T \mathbf{I}_{\mathrm{i}} \boldsymbol{\omega}_{\mathrm{i}}) = \frac{1}{2}\sum_{1}^{n} \dot{\boldsymbol{\theta}}^T \bar{\mathbf{I}}_i \dot{\boldsymbol{\theta}}}$$

where $$\boxed{\bar{\mathbf{I}}_i = m_i \mathbf{J}_{c,i}^T \mathbf{J}_{c,i} + \mathbf{J}_{\omega,i}^T \mathbf{I}_i \mathbf{J}_{\omega,i}}$$

I is called the *generalized inertia matrix* (GIM)

In a different reference frame, say, the frame, i, tensor $\mathbf{I}_i$ can be obtained as:

$$[\mathbf{I}_i]_i = \mathbf{Q}_i [\mathbf{I}_i]_{i+1} \mathbf{Q}_i^T$$

Equations of Motion

Lagrangian:

$$L = T - U = \sum_{i=1}^{n} [\frac{1}{2}\dot{\boldsymbol{\theta}}^{\mathrm{T}}\, \bar{\mathbf{I}}_i\, \dot{\boldsymbol{\theta}} + m_i \mathbf{c}_i^{\mathrm{T}} \mathbf{g}]$$

Dynamic equations of motion are obtained as

$$\sum_{j=1}^{n} i_{ij} \ddot{\theta}_{ij} + h_i + \gamma_i = \tau_i$$

$$h_i \equiv \sum_{j=1}^{n} \sum_{k=1}^{n} \left(\frac{\partial i_{ij}}{\partial \theta_k} - \frac{1}{2} \frac{\partial i_{jk}}{\partial \theta_i} \right) \dot{\theta}_j \dot{\theta}_k \qquad \gamma_i \equiv -\sum_{j=1}^{n} [\mathbf{j}_{c,i}^{(j)}]^T (m_j \mathbf{g})$$

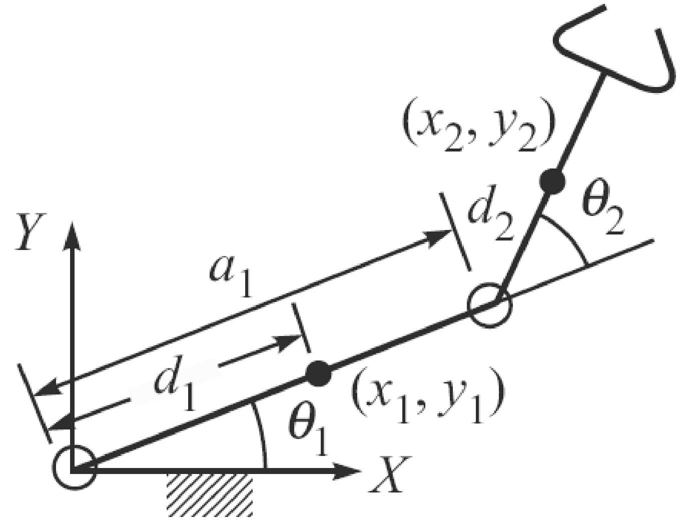
$$\boxed{\mathbf{I}\,\ddot{\boldsymbol{\theta}} + \mathbf{h} + \boldsymbol{\gamma} = \boldsymbol{\tau}}$$

Example: Dynamic Model of a 2-link Arm

;

$$\mathbf{J}_{\omega,1} \equiv \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{bmatrix} \qquad \mathbf{J}_{\omega,2} \equiv \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{bmatrix}$$

;



$$\mathbf{J}_{c,1} \equiv \begin{bmatrix} -\frac{a_1}{2}s_1 & 0 \\ \frac{a_1}{2}c_1 & 0 \\ 0 & 0 \end{bmatrix} \qquad \mathbf{J}_{c,2} \equiv \begin{bmatrix} -a_1 s_1 - \frac{a_2}{2}s_{12} & -\frac{a_2}{2}s_{12} \\ a_1 c_1 + \frac{a_2}{2}c_{12} & \frac{a_2}{2}c_{12} \\ 0 & 0 \end{bmatrix}$$

Equations of Motion of a 2-link Arm

$$\mathbf{I}\ddot{\boldsymbol{\theta}} + \mathbf{h} + \boldsymbol{\gamma} = \boldsymbol{\tau}$$

$$\mathbf{I} \equiv \begin{bmatrix} i_{11} & i_{12} \\ i_{21} & i_{22} \end{bmatrix}$$

$$i_{11} = \frac{m_1}{4} a_1^2 + I_{1,zz} + m_2(a_1^2 + \frac{a_2^2}{4} + a_1 a_2 c_2) + I_{2,zz}$$

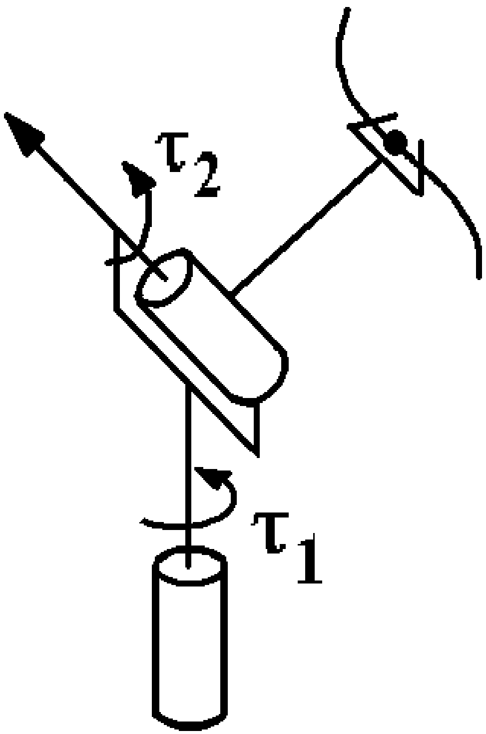
$$i_{22} = m_2 \frac{a_2^2}{4} + I_{2,zz} \qquad i_{12} = i_{21} = m_2(\frac{a_2^2}{4} + \frac{a_1 a_2 c_2}{2}) + I_{2,zz}$$

$$h_1 = -m_2 a_1 a_2 s_2 \dot{\theta}_1 \dot{\theta}_2 - m_2 \frac{a_1 a_2}{2} s_2 \dot{\theta}_2^2 \qquad h_2 = \frac{m_2}{2} a_1 a_2 s_2 \dot{\theta}_1^2$$

$$\gamma_1 = m_1 g \frac{a_1}{2} c_1 + m_2 g(a_1 c_1 + \frac{a_2}{2} c_{12}) \qquad \gamma_2 = m_1 g \frac{a_2}{2} c_{12}$$

Dynamics Algorithms

To control a robot

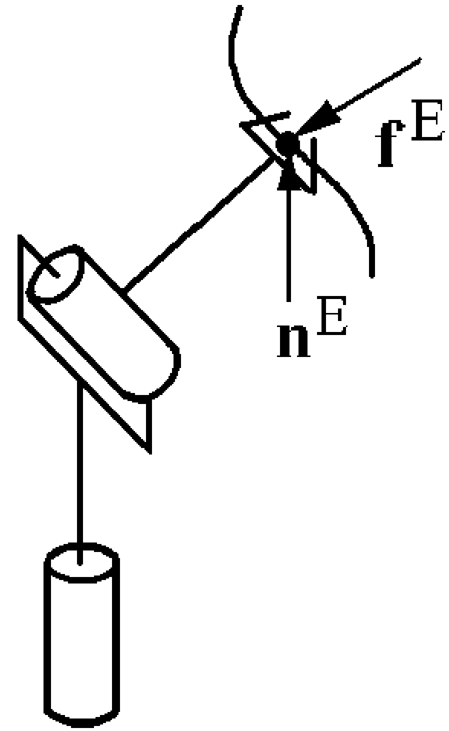


Inverse Dynamics

Find joint torques/forces for given joint motions and end-effector moment/force

Forward Dynamics

Find end-effector motion for known joint torques/forces



To simulate a robot

Inverse Dynamics

$$\mathbf{I}\ddot{\boldsymbol{\theta}} + \mathbf{h} + \boldsymbol{\gamma} = \boldsymbol{\tau}$$

- Joint angles, velocities, and accelerations are inputs as per desired trajectories of the joints
- Calculate the left-hand side
- Requires only multiplications and additions
- Straightforward

Inverse Dynamics of One-link Planar Arm Using MATLAB

Cycloidal trajectory

$$\theta = \theta(0) + \frac{\theta(T) - \theta(0)}{T}\left[t - \frac{T}{2\pi}\sin\left(\frac{2\pi}{T}t\right)\right]$$

$$\dot{\theta} = \frac{\theta(T) - \theta(0)}{T}\left[1 - \cos\left(\frac{2\pi}{T}t\right)\right]$$

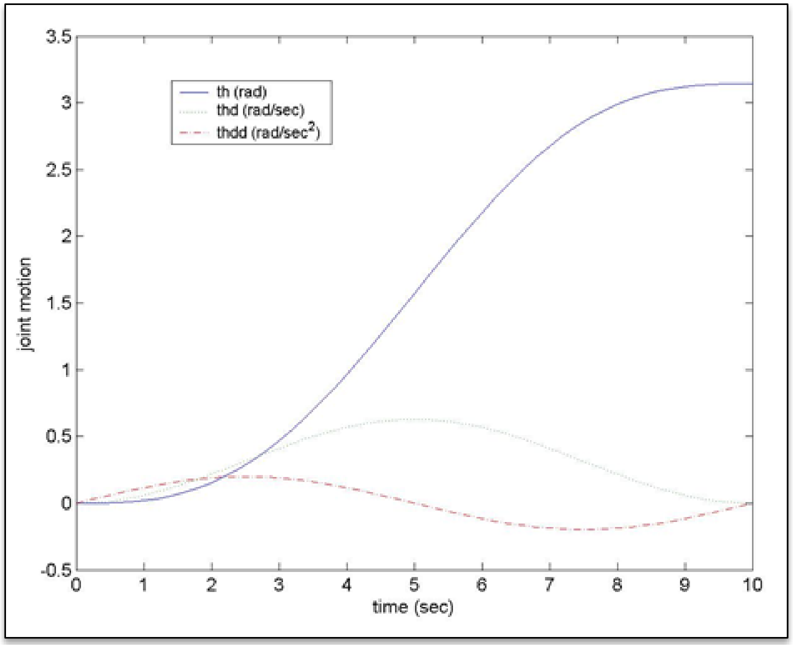
$$\ddot{\theta} = \frac{\theta(T) - \theta(0)}{T}\left[\frac{2\pi}{T}\sin\left(\frac{2\pi}{T}t\right)\right]$$

$$\frac{1}{3}ma^2\ddot{\theta} + \frac{1}{2}mgas\theta = \tau$$

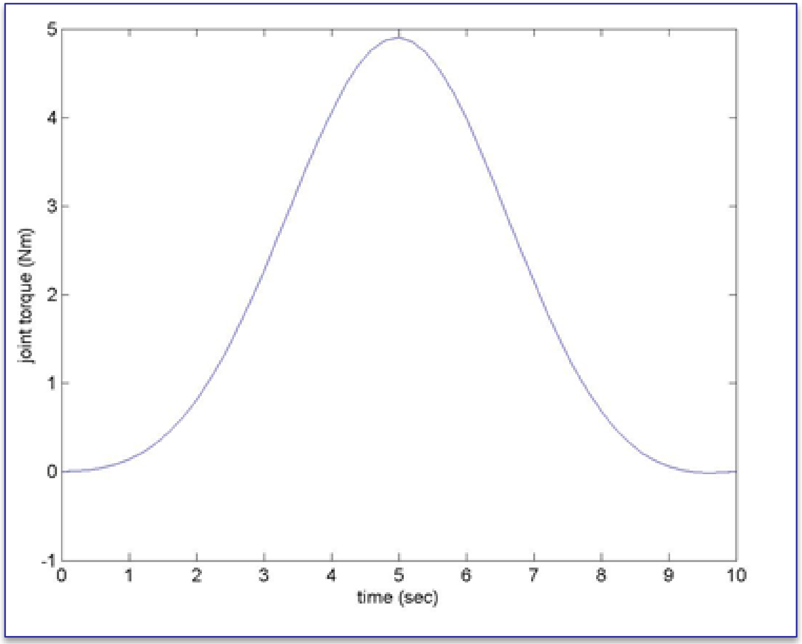
```
% Input for trajectory and link parameters
T = 10; thT = pi; th0 = 0; m = 1; a = 1; g = 9.81;
con = 2*pi/T; delth = thT - th0;
iner = m*a*a/3; grav = m*g*a/2;
for i = 1:51,
    ti (i) = (i-1)*T/50;
    ang = con*ti(i);
% Joint trajectory
th (i) = th0 + (delth/T)*(ti (i) - sin(ang)/con);
thd (i) = delth*(1 - cos(ang))/T;
thdd (i) = delth*con*sin(ang)/T;
% Joint torque
tau (i)= iner*thdd (i) + grav*sin(th(i));
end
plot (ti,th,'-',ti,thd,':',ti,thdd,'-.')
figure
plot (ti, tau)
```

MATLAB program to find joint torque

Inverse Dynamics of One-link Planar Arm Using MATLAB



Joint trajectory (input)



Joint torque (output)

MATLAB

Two-link Planar Arm

```
clear all
% Input for trajectory and link parameters
T = 10; th1T = pi; th10 = 0; th2T = pi/2; th20 = 0;
m1 = 1; a1 = 1; m2 = 1; a1 = 1; a2 = 1; g = 9.81;
con = 2*pi/T; delth1 = th1T - th10; delth2 = th2T - th20; iner21 =
m2*a1*a2;
for i = 1:51,
   ti (i) = (i-1)*T/50;    ang = con*ti(i);
 % Joint trajectory
 th1 (i) = th10 + (delth1/T)*(ti (i) - sin(ang)/con);
th1d (i) = delth1*(1 - cos(ang))/T; th1dd (i) = delth1*con*sin(ang)/T;
th2 (i) = th20 + (delth2/T)*(ti (i) - sin(ang)/con);
th2d (i) = delth2*(1 - cos(ang))/T; th2dd (i) = delth2*con*sin(ang)/T;
thdd = [th1dd(i);th2dd(i)];
%Inertia matrix
sth2 = sin(th2(i)); cth2 = cos(th2(i));
i22 = m2*a2*a2/3;i21 = i22 + iner21*cth2/2;i12 = i21;
i11 = i22 + m1*a1*a1/3 + m2*a1*a1 + iner21*cth2;
im = [i11, i12; i21, i22]
%h-vector
h1 = - (m2*a1*a2*th1d(i) + iner21/2*th2d(i))*th2d(i)*sth2;
h2 = iner21/2*sth2*th1d(i)*th1d(i); hv=[h1;h2]
%gamma-vector
cth1 = cos(th1(i)); cth12 = cos(th1(i) + th2(i));
gam1 = m1*g*a1/2*cth1 + m2*g*(a1*cth1 + a2/2*cth12);
gam2 = m1*g*a2/2*cth12; gv = [gam1;gam2]
% Joint torque
tau=im*thdd + hv + gv; tor1(i) = tau(1); tor2(i)=tau(2);
end
plot(ti,th1,'-',ti,th2,':')
figure
plot (ti, tor1,'-',ti,tor2,':')
```

$$i_{11} = \frac{m_1}{4} a_1^2 + I_{1,zz} + m_2\left(a_1^2 + \frac{a_2^2}{4} + a_1 a_2 c_2\right) + I_{2,zz}$$

$$i_{12} = i_{21} = m_2\left(\frac{a_2^2}{4} + \frac{a_1 a_2 c_2}{2}\right) + I_{2,zz}$$

$$i_{22} = m_2 \frac{a_2^2}{4} + I_{2,zz}$$

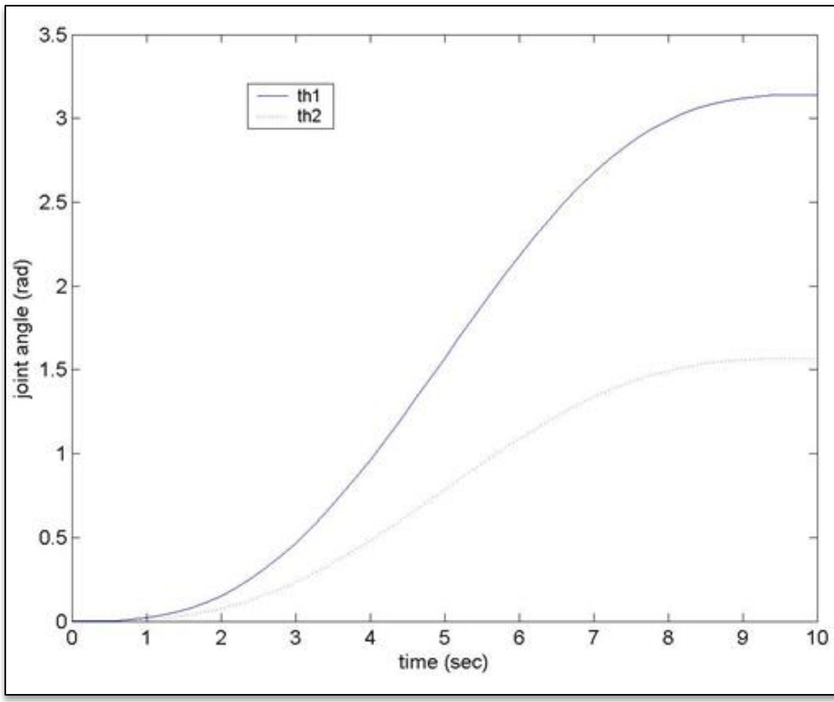
$$h_1 = -m_2 a_1 a_2 s_2 \dot{\theta}_1 \dot{\theta}_2 - m_2 \frac{a_1 a_2}{2} s_2 \dot{\theta}_2^2$$

$$h_2 = \frac{m_2}{2} a_1 a_2 s_2 \dot{\theta}_1^2$$

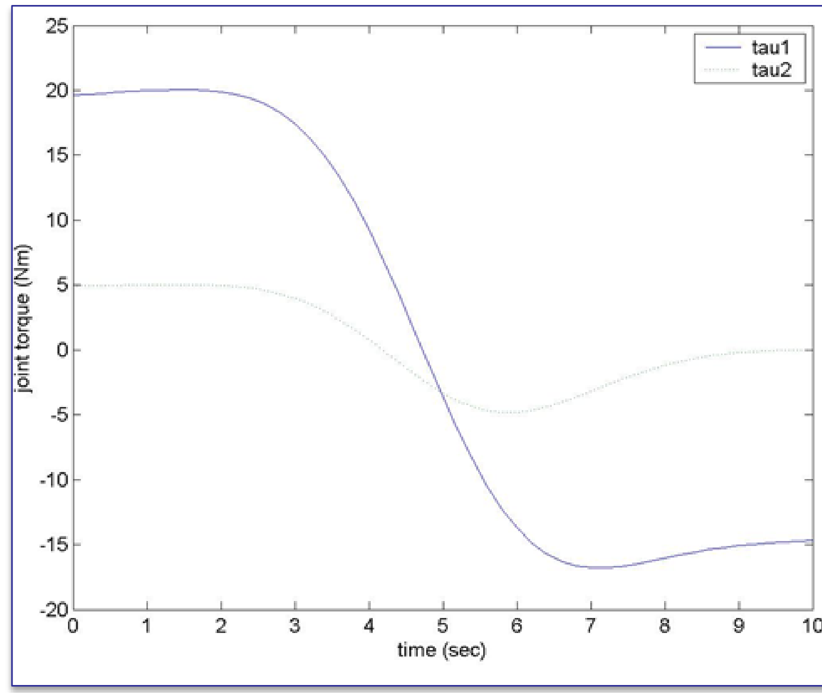
$$\gamma_1 = m_1 g \frac{a_1}{2} c_1 + m_2 g\left(a_1 c_1 + \frac{a_2}{2} c_{12}\right)$$

$$\gamma_2 = m_1 g \frac{a_2}{2} c_{12}$$

Inverse Dynamics of Two-link Planar Arm Using MATLAB

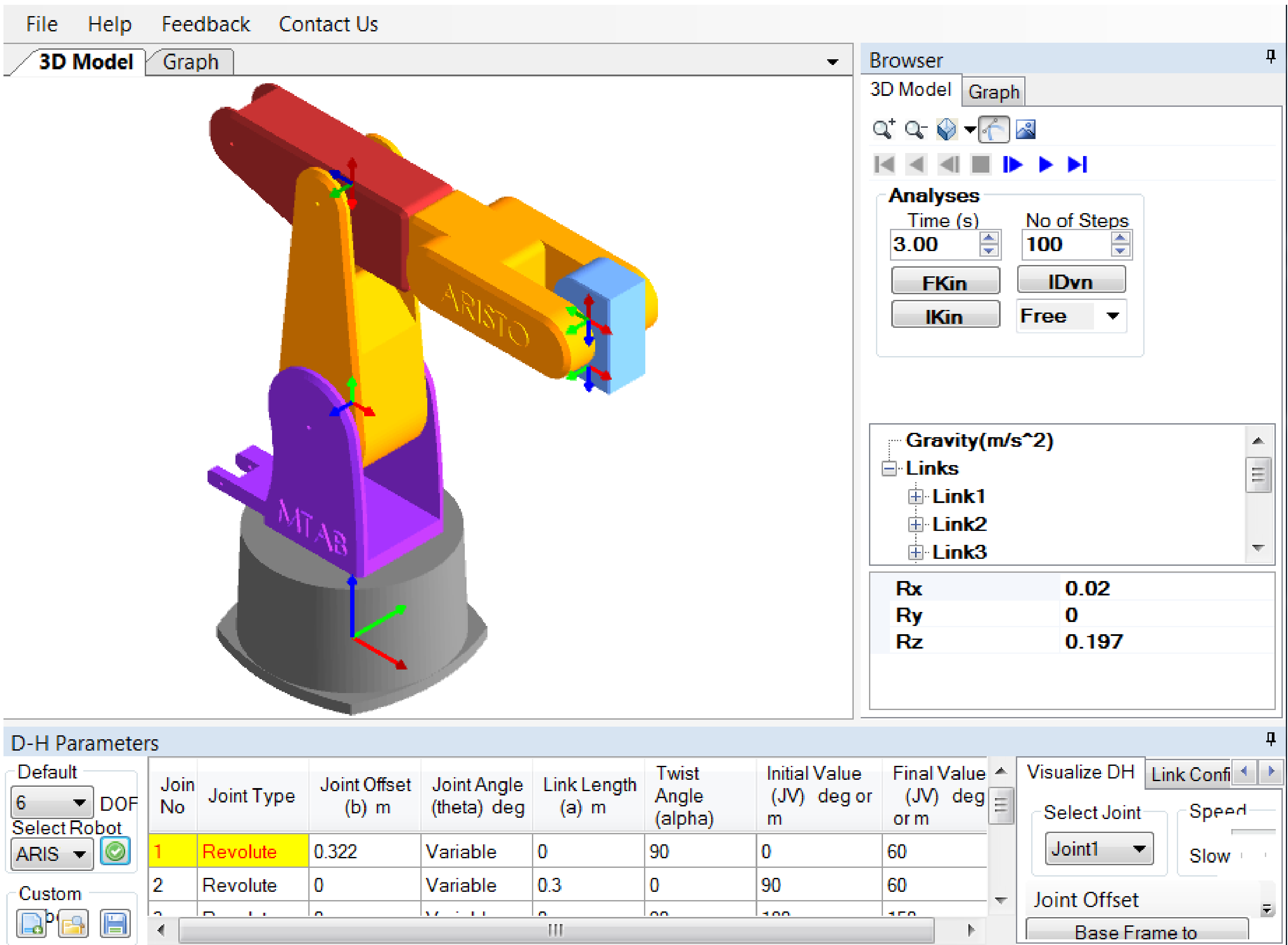


Joint trajectories (input)



Joint torques (output)

Inverse Dynamics of 6-DOF Aristo Robot



Forward Dynamics

$$\boxed{\ddot{\boldsymbol{\theta}} = \mathbf{I}^{-1}(\boldsymbol{\tau} - \mathbf{h} - \boldsymbol{\gamma})}$$

$$\mathbf{y}(t) \equiv [\mathbf{y}_1^T(t), \mathbf{y}_2^T(t)]^T \qquad \mathbf{y}_1(t) = \boldsymbol{\theta}; \mathbf{y}_2(t) = \dot{\boldsymbol{\theta}}$$

$$\dot{\mathbf{y}}(t) \equiv \begin{bmatrix} \dot{\mathbf{y}}_1(t) \\ \dot{\mathbf{y}}_2(t) \end{bmatrix} = \begin{bmatrix} \mathbf{y}_2(t) \\ \mathbf{I}^{-1}(\boldsymbol{\tau} - \mathbf{h} - \boldsymbol{\gamma}) \end{bmatrix}$$

- MATLAB software has in-built routines like ode45 and others to perform the numerical integration

Simulation of One-link Arm using MATLAB

$$\ddot{\theta} = \frac{2}{ma^2}(\tau - \frac{1}{2} mga\sin\theta)$$

Hence, the state-space form is given by

$$\dot{y}_1 = y_2$$

$$\dot{y}_2 = \frac{2}{ma^2}(\tau - \frac{1}{2} mga\sin\theta)$$

```
%For one-link arm
function ydot =ch8fdyn1(t,y);
m = 1; a = 1; g = 9.81; tau=0;
iner = m*a*a/3; grav = m*g*a/2;
ydot=[y(2);(tau-grav*sin(y(1)))/iner];
```

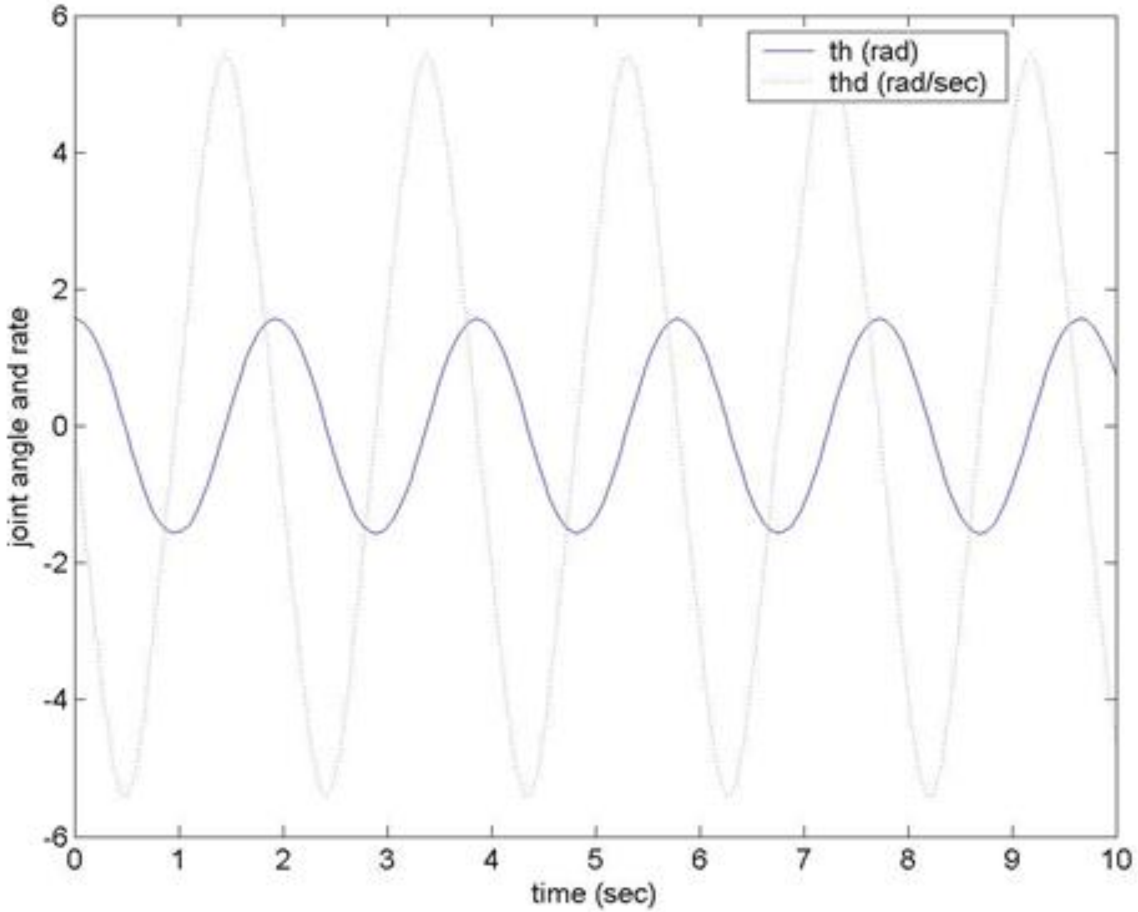
Program for state-space form

```
%For one link arm
tspan=[0 10]; y0=[pi/2; 0];
[t,y]=ode45('ch8fdyn1',tspan,y0);
plot(t,y)
```

Program for numerical integration

MATLAB

Simulation Results of One-link Arm



Simulation results for one-link arm

MATLAB

Simulation of Two-link Arm

```
%For two-link manipulator
function ydot =ch8fdyn2(t,y);
m1 = 3; m2 = 1; a1 = 2; a2 = 1; g = 9.81; iner21 = m2*a1*a2;
tau1 = 0; tau2 = 0;
th1=y(1); th2 =y(2); th1d=y(3); th2d=y(4);

%Inertia matrix
sth2 = sin(th2); cth2 = cos(th2);
i22 = m2*a2*a2/3;
i21 = i22 + iner21*cth2/2; i12 = i21;
i11 = i22 + m1*a1*a1/3 + m2*a1*a1 + iner21*cth2;
im = [i11, i12; i21, i22];

%h-vector
h1 = - (m2*a1*a2*th1d + iner21/2*th2d)*th2d*sth2;
h2 = iner21/2*sth2*th1d*th1d;
hv=[h1;h2];

%gamma-vector
cth1 = cos(th1); cth12 = cos(th1 + th2);
gam1 = m1*g*a1/2*cth1 + m2*g*(a1*cth1 + a2/2*cth12);
gam2 = m1*g*a2/2*cth12;
gv = [gam1;gam2];

% RHS
tau=[tau1;tau2];
phi=tau-hv-gv; thdd=im\phi;
ydot=[y(3);y(4);thdd(1);thdd(2)];
```

Program for state-space form

$$y_1 = \theta_1; y_2 = \theta_2$$

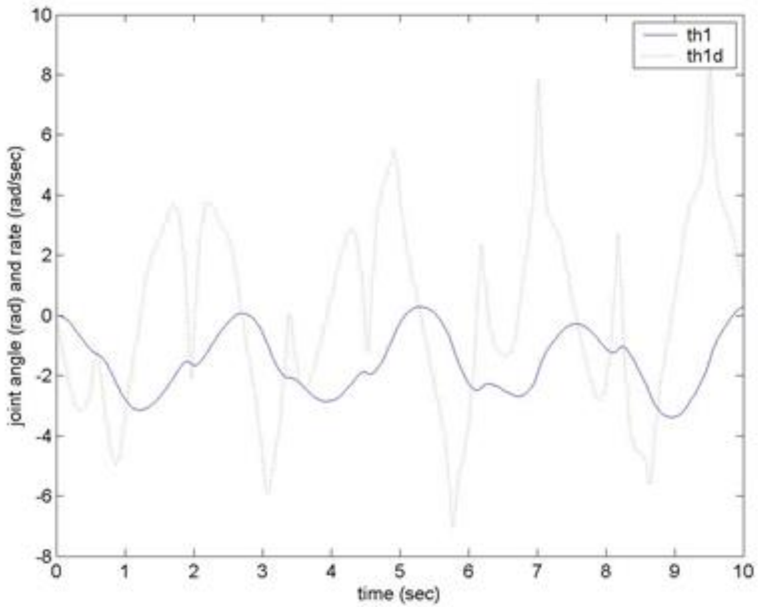
$$y_3 = \dot{y}_1; y_4 = \dot{y}_2$$

$$\begin{bmatrix} \dot{y}_3 \\ \dot{y}_4 \end{bmatrix} = \mathbf{I}^{-1}(\boldsymbol{\tau} - \mathbf{h} - \boldsymbol{\gamma})$$

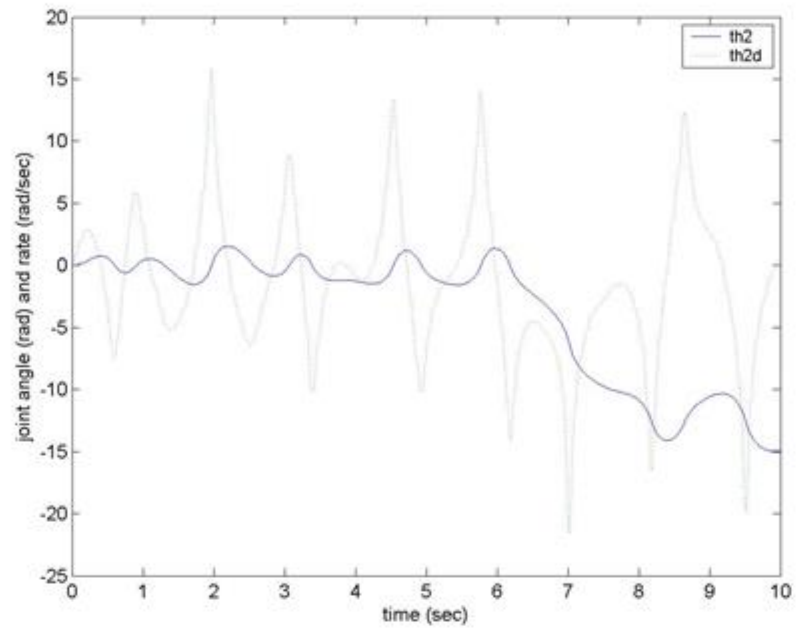
```
%For two-link manipulator
tspan=[0 10]; y0=[0;0;0;0];
[t,y]=ode45('ch8fdyn2',tspan,y0);
```

Program for numerical integration

Simulation of Two-link Arm using MATLAB



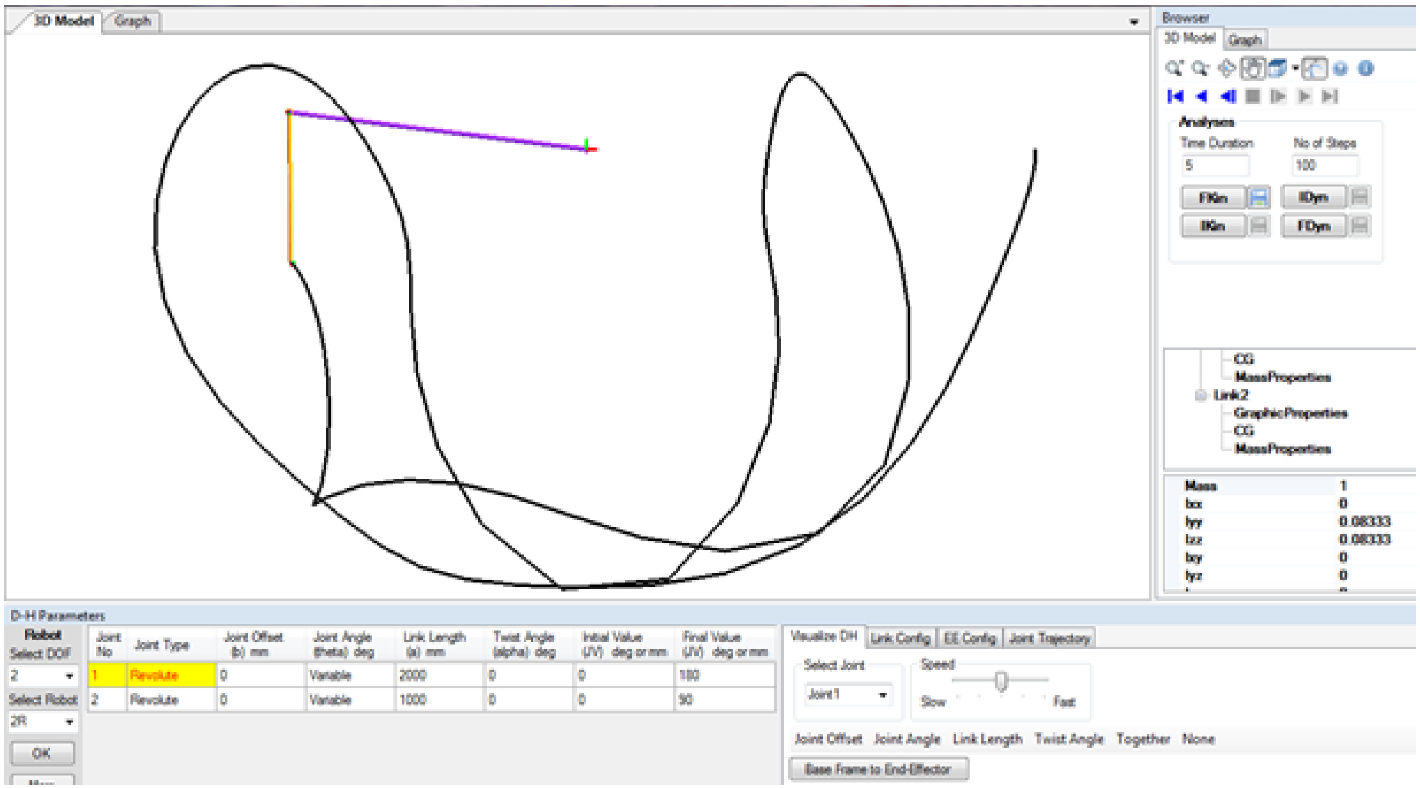
(a) Joint 1



(a) Joint 2

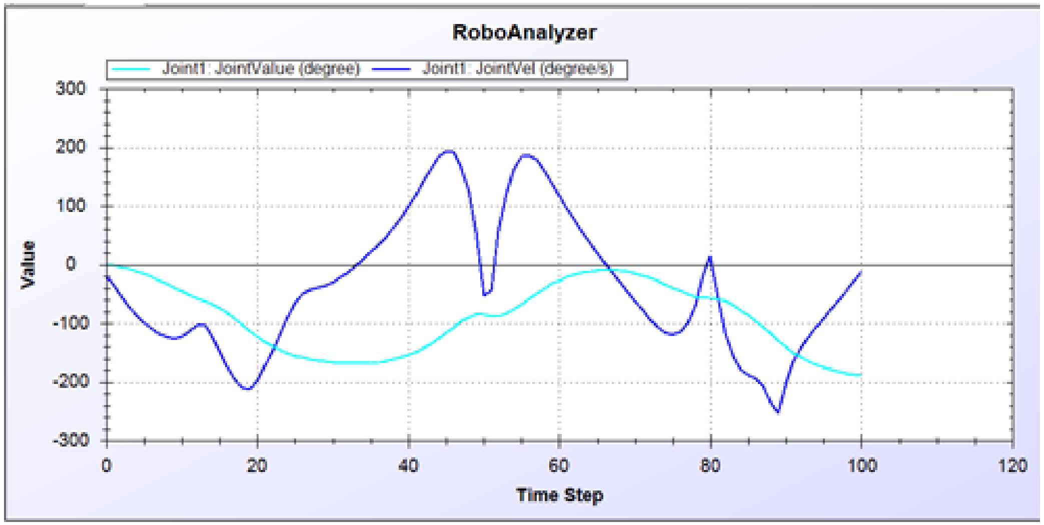
Simulation results for the two-link arm under gravity

Simulation of Two-link Arm using RoboAnalyzer

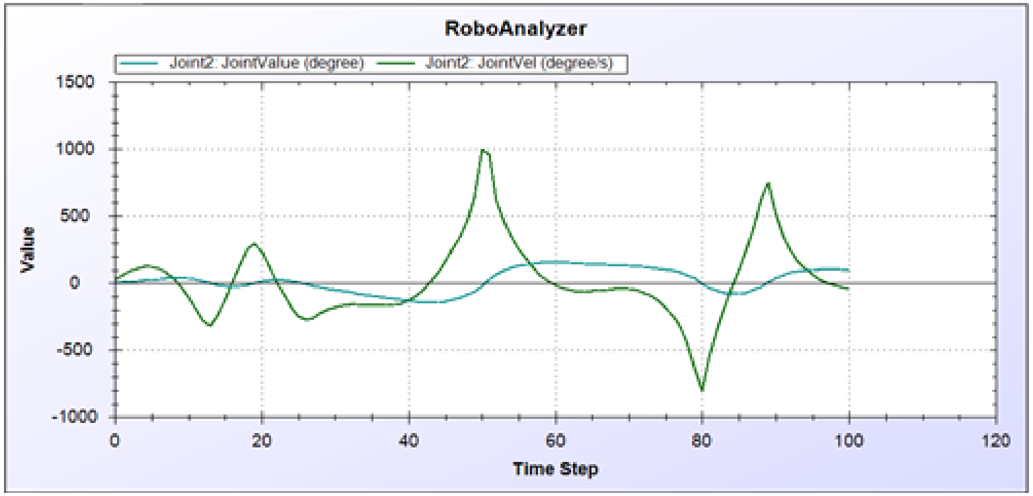


Screenshot of the animation

Simulation of Two-link Arm using RoboAnalyzer

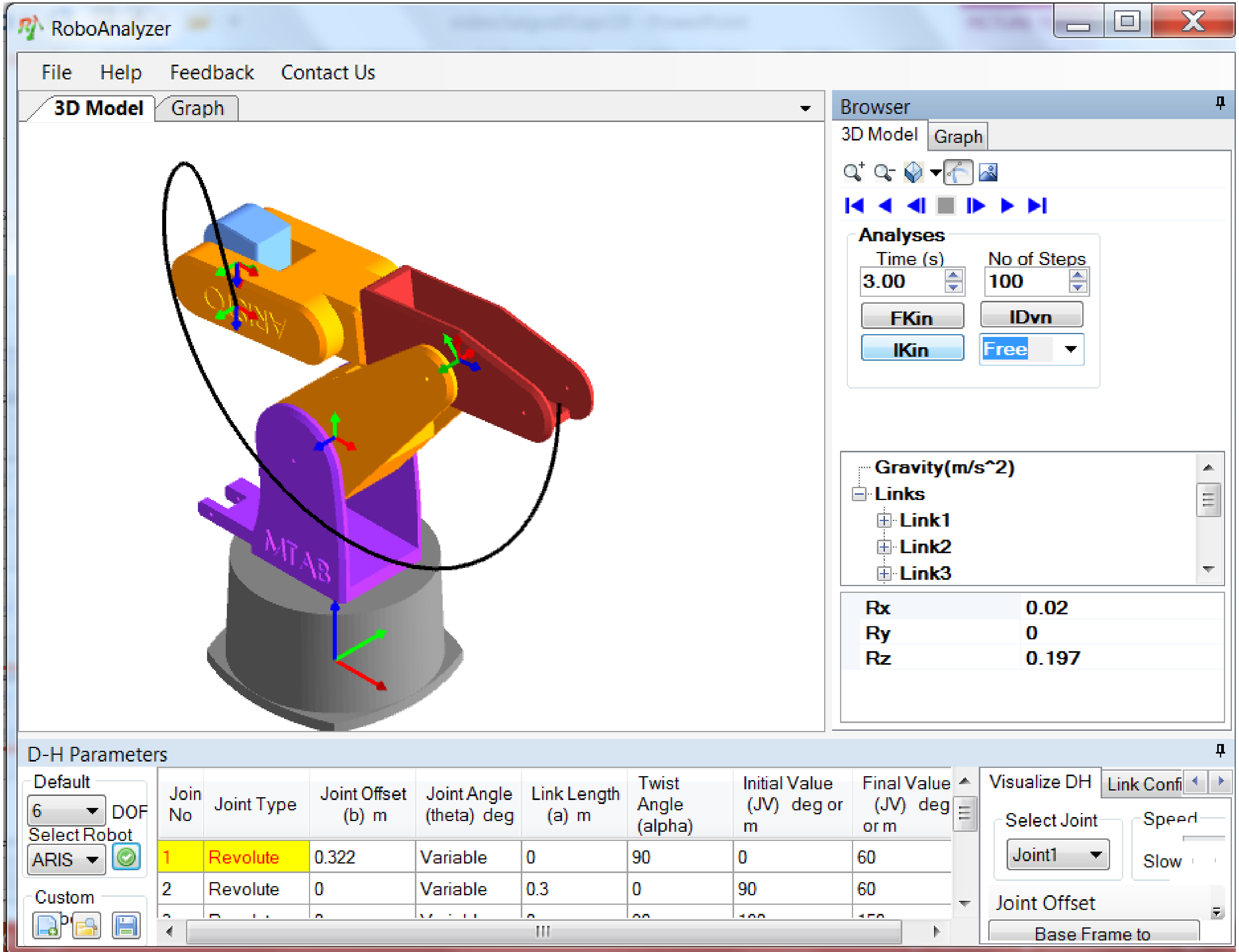


Variations of angles and rates for joint 1



Variations of angles and rates for joint 2

Simulation of Aristo Robot



Summary

- Euler-Lagrange formulation is presented
- Concept of Generalized Coordinates was introduced.
- Inverse dynamics and forward dynamics were defined
- Results for planar one-link and two-link robot arms and ARISTO Robot were obtained using MATLAB and RoboAnalyzer

Thank you

For any doubts, contact

saha@mech.iitd.ac.in
http://sksaha.com