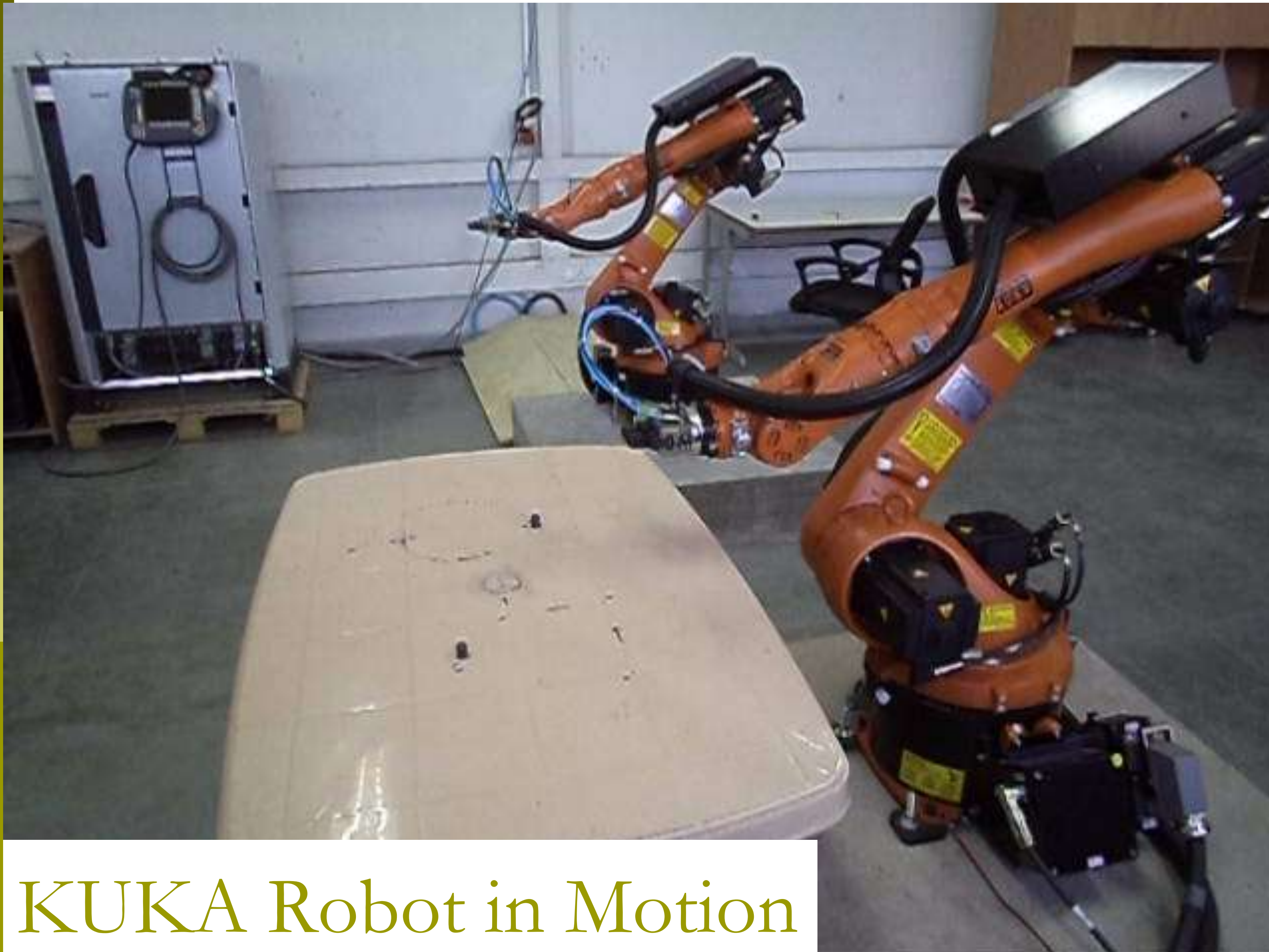
Recursive Dynamics: Application to Robotics, Rural Machines, Ropes, and What Next?

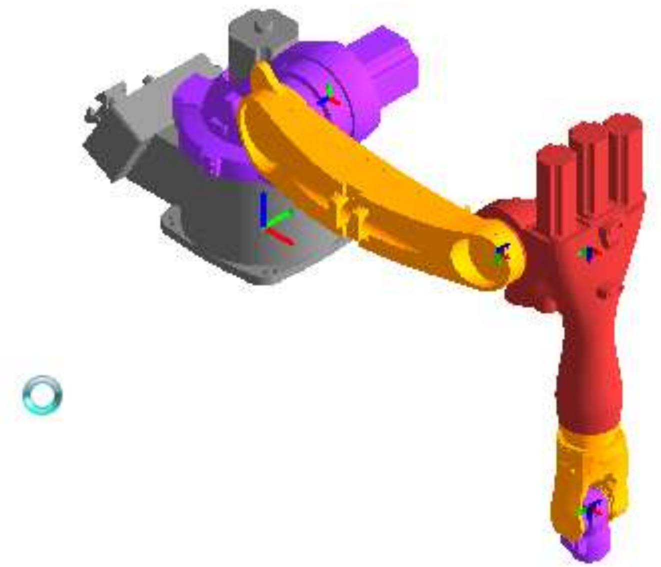
Prof. S.K. Saha
Naren Gupta Chair Professor
Dept. of Mech. Eng., IIT Delhi

March 15, 2012



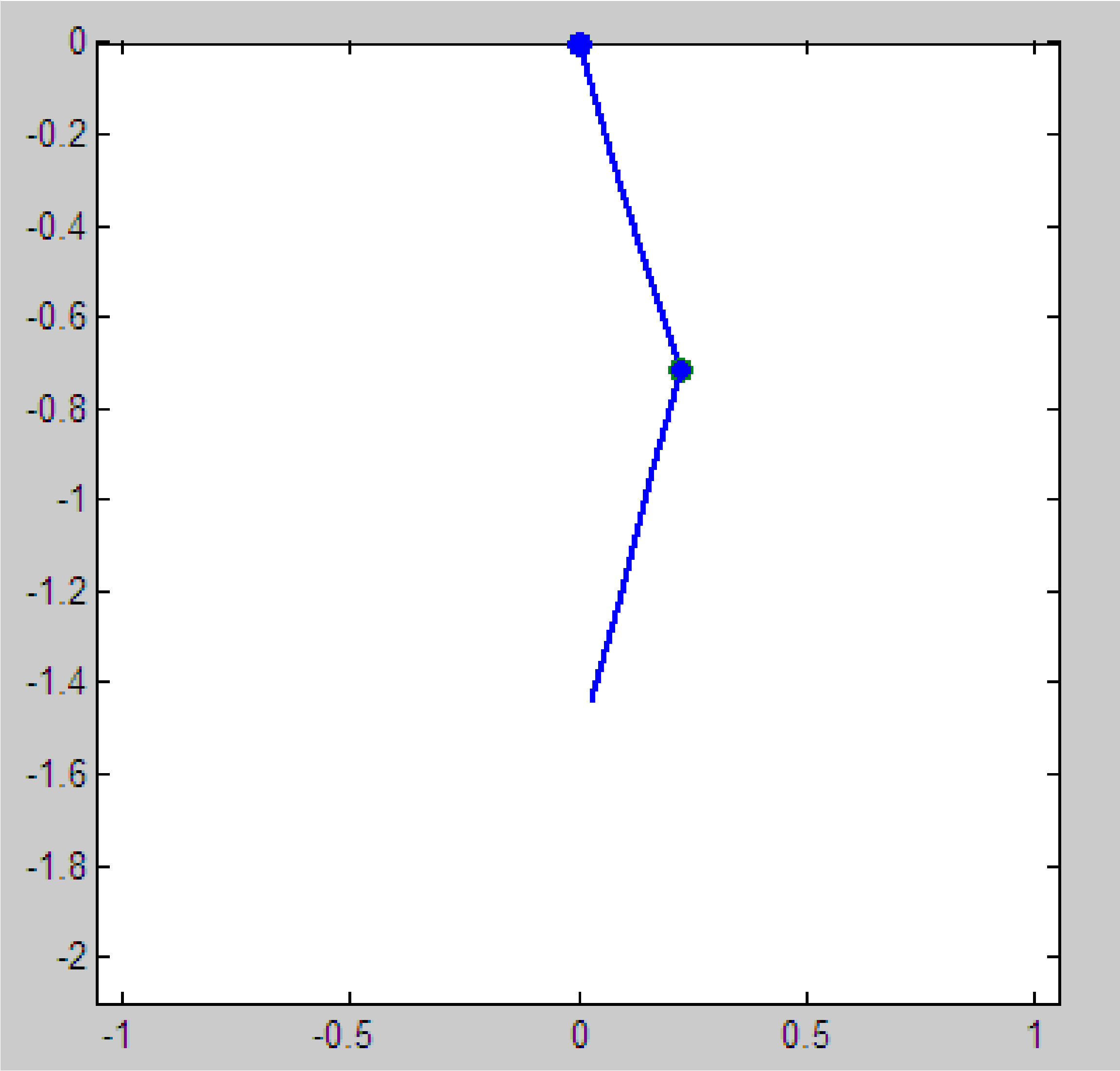
KUKA Robot in Motion

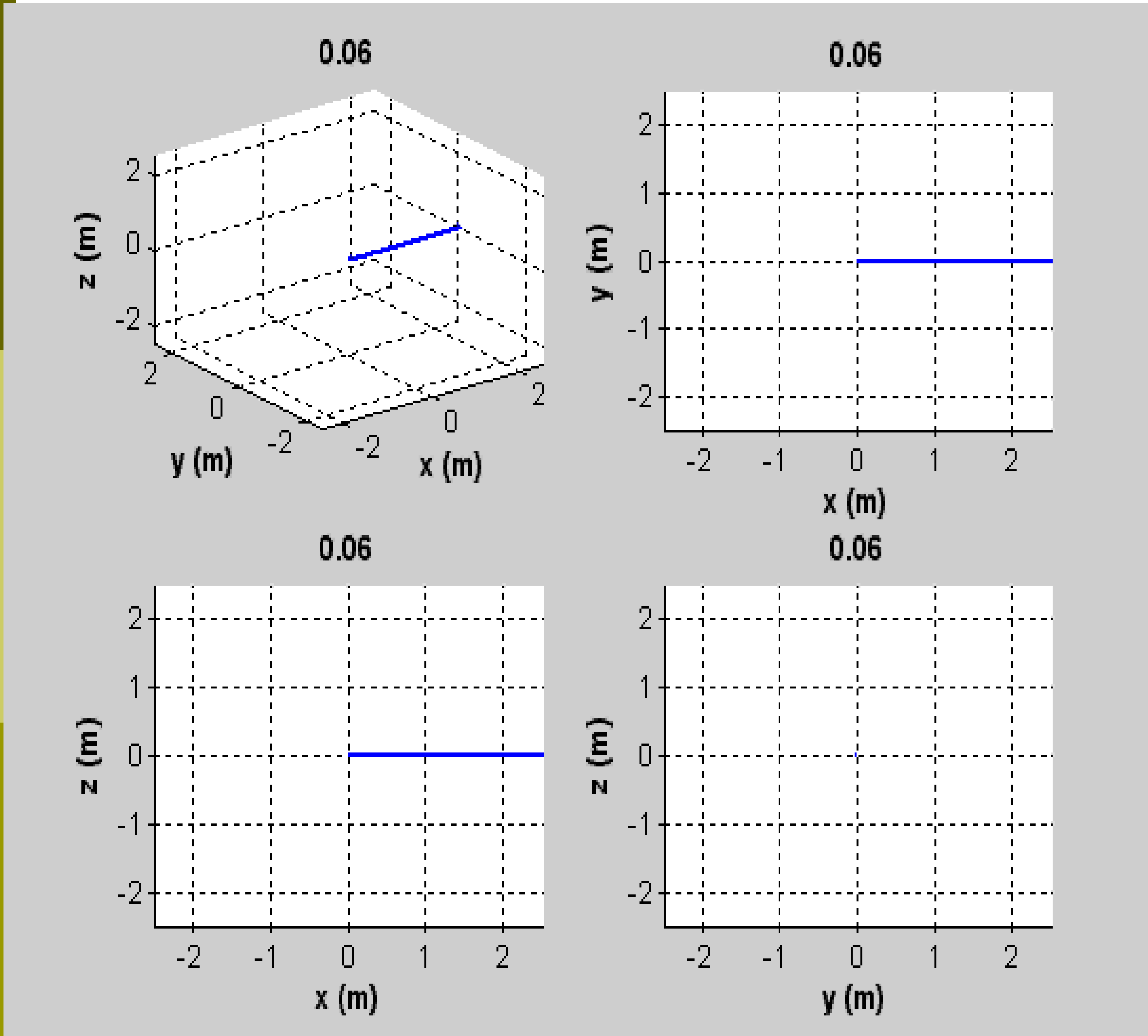
KUKA Robot in RoboAnalyzer

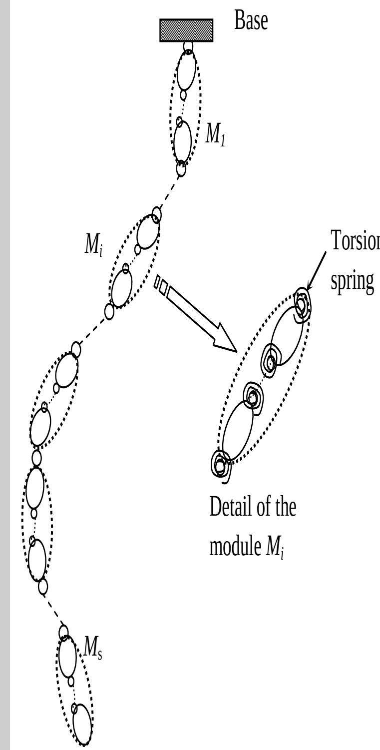


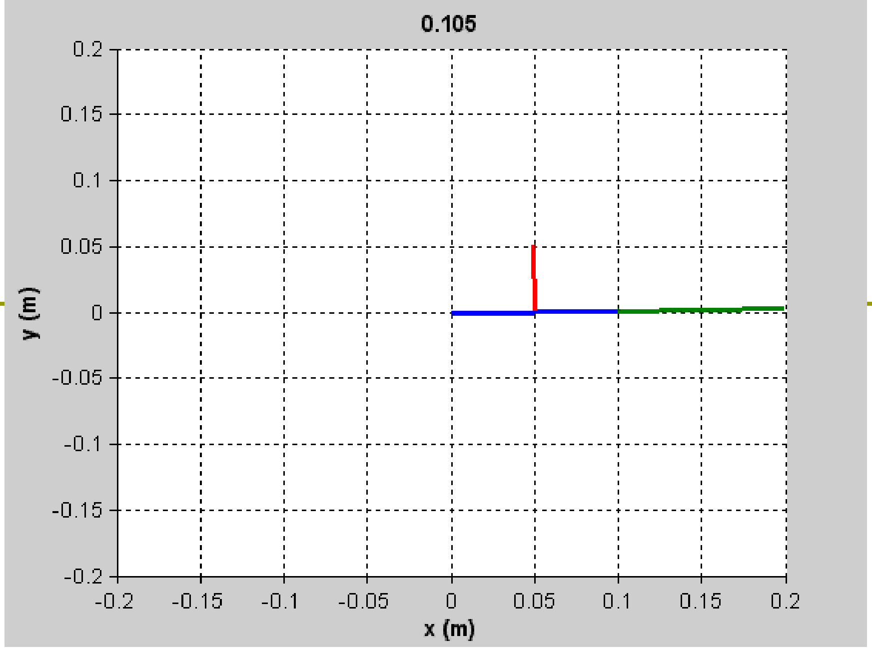
Vibrating Robot Arm

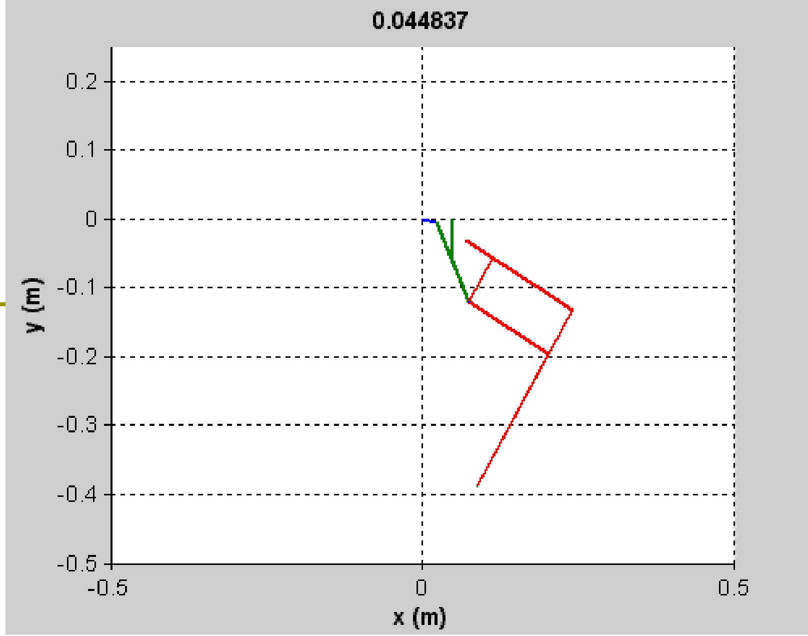
Free-fall Simulation

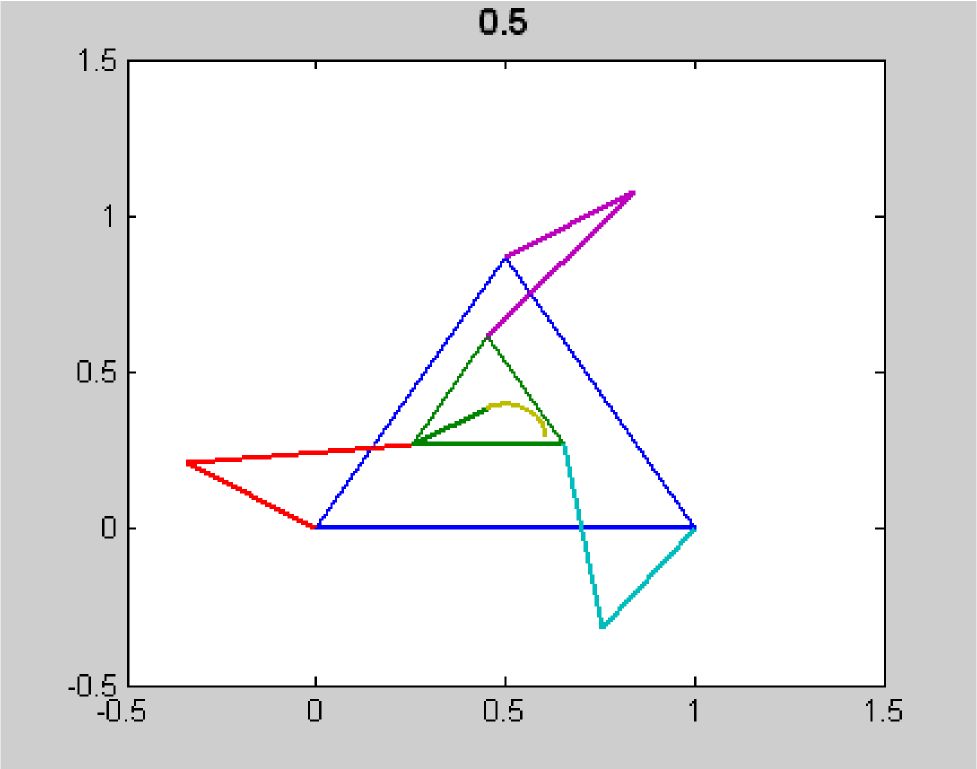


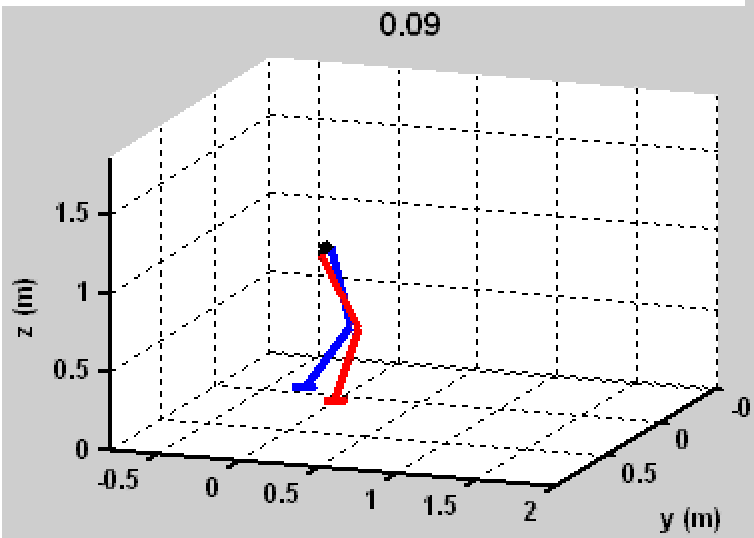

0.06
z (m)
y (m)
x (m)
0.06
y (m)
x (m)
0.06
z (m)
x (m)
0.06
z (m)
y (m)

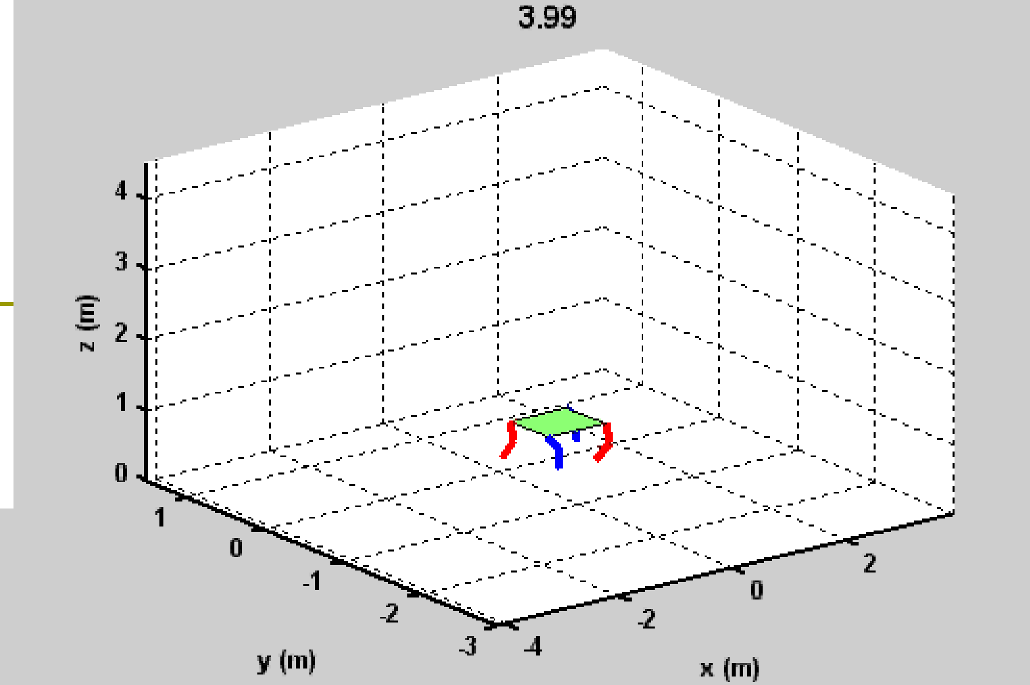

Base
M_1
M_i
Torsion spring
Detail of the module M_i
M_s

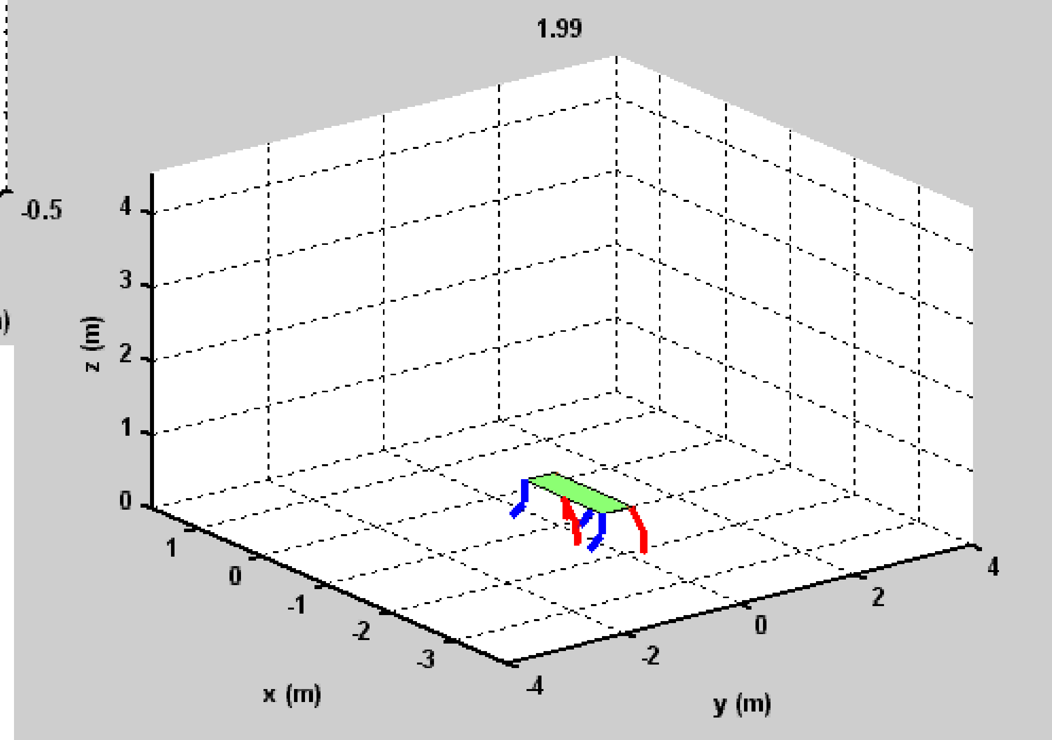

0.105
y (m)
x (m)
0.2
0.15
0.1
0.05
0
-0.05
-0.1
-0.15
-0.2


0.044837
y (m)
x (m)
0.2
0.1
0
-0.1
-0.2
-0.3
-0.4
-0.5
0.5


0.5
1.5
1
0.5
0
-0.5


0.09
z (m)
1.5
1
0.5
0
-0.5
0
0.5
1
1.5
2
-0.5
0
0.5
y (m)


3.99
z (m)
4
3
2
1
0
1
0
-1
-2
-3
y (m)
-4
-2
0
2
x (m)

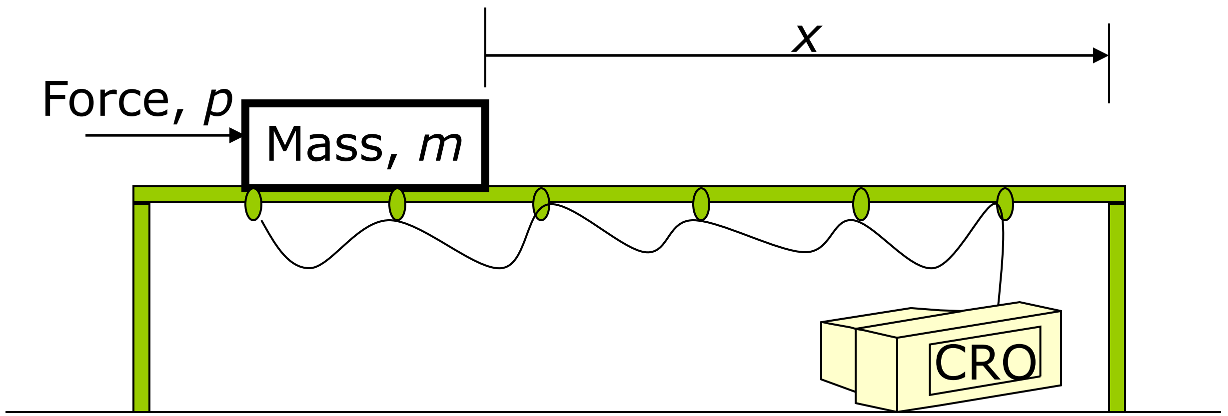

1.99
z (m)
4
3
2
1
0
1
0
-1
-2
-3
x (m)
-4
-2
0
2
4
y (m)

Plan of Presentation

- Purpose
- Serial systems
  - RoboAnalyzer
- Closed-loop system
  - Multibody Dynamics for Rural Applications
- Tree-type system
  - Modeling
  - Simulation
- Conclusions

Modelling and Simulation

Actually

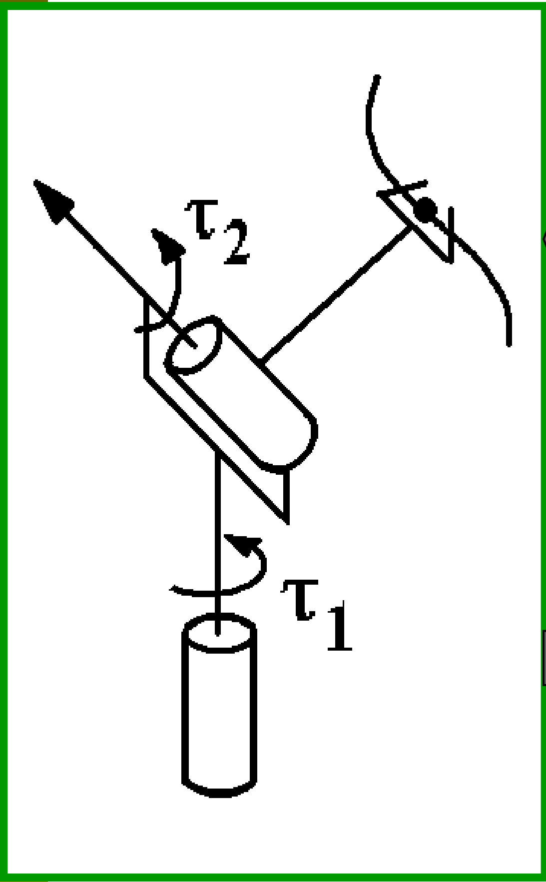


Mathematically

Newton's 2nd law: $p = mf$ → Modelling

Find, $f = p/m$; $v = \int f\, dt$; $x = \int v\, dt$ → Simulation

Inverse vs. Forward Dynamics

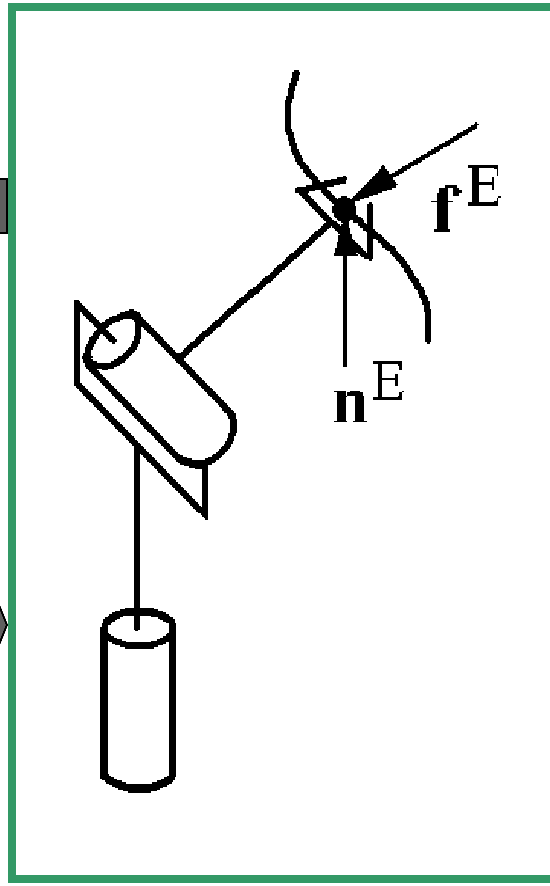


Inverse Dynamics

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

Forward Dynamics

Find end-effector motion for known joint torques/forces



Serial Robots

A Decomposition of the Manipulator Inertia Matrix

Subir Kumar Saha

Abstract—A decomposition of the manipulator inertia matrix is essential, for example, in forward dynamics, where the joint accelerations are solved from the dynamical equations of motion. To do this, unlike a numerical algorithm, an analytical approach is suggested in this paper. The approach is based on the symbolic Gaussian elimination of the inertia matrix that reveal recursive relations among the elements of the resulting matrices. As a result, the decomposition can be done with the complexity of order n, $\mathcal{O}(n)$—n being the degrees of freedom of the manipulator—, as opposed to an $\mathcal{O}(n^3)$ scheme, required in the numerical approach. In turn, $\mathcal{O}(n)$ inverse and forward dynamics algorithms can be developed. As an illustration, an $\mathcal{O}(n)$ forward dynamics algorithm is presented.

Index Terms— Articulated-body inertia, Kalman filtering, reverse Gaussian elimination (RGE), serial manipulator, symbolic decomposition.

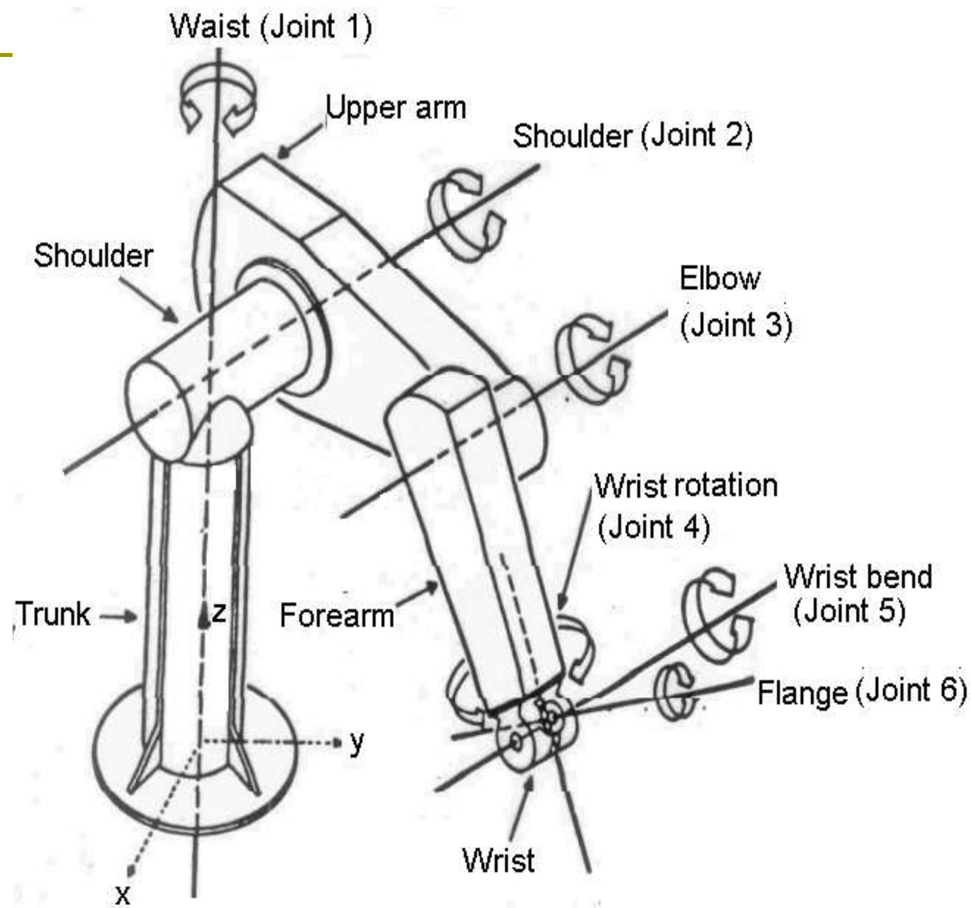
I. INTRODUCTION

The inertia matrix of a robotic manipulator or the generalized inertia matrix (GIM) arises from the robot's dynamic equations of motion. The decomposition of the GIM is required, for example,

Manuscript received February 21, 1995; revised August 30, 1995. This paper was recommended for publication by Associate Editor V. Kumar and Editor S. E. Salcudean upon evaluation of the reviewers' comments.

The author is with Department of Mechanical Engineering, Indian Institute of Technology, Delhi, Hauz Khas, New Delhi 110 016, India.

Publisher Item Identifier S 1042-296X(97)01052-5.



1997 IEEE Trans. on Rob. & Aut.
V. 13, N. 2, Apr., pp. 301-304

PUMA Robot

Methods

- Newton-Euler (NE)

  Euler's: $\mathbf{I}_i\dot{\omega}_i + \omega_i \times \mathbf{I}_i\omega_i = \mathbf{n}_i$

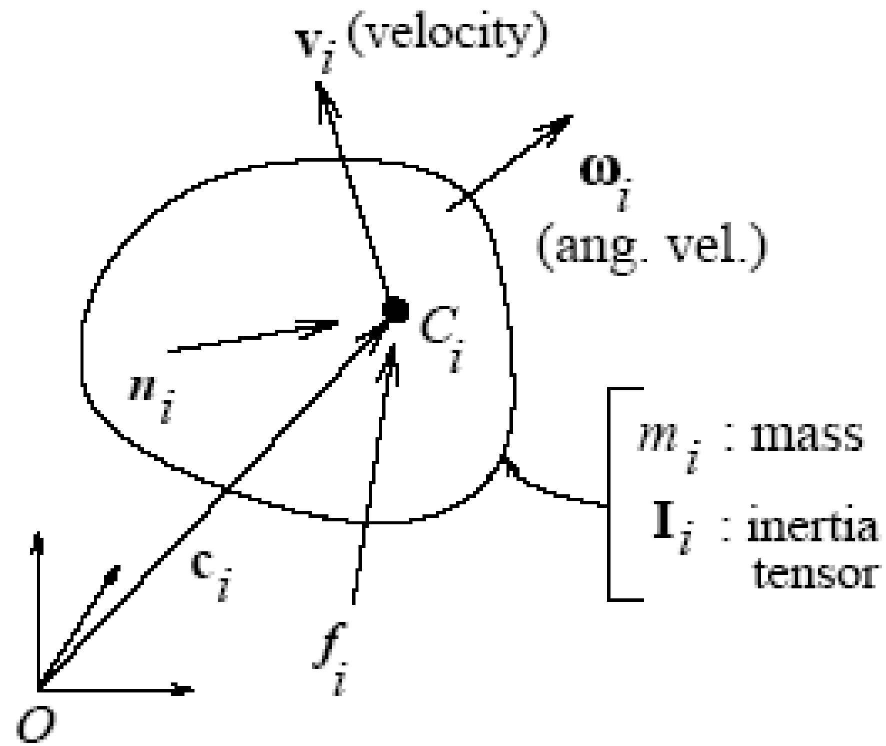
  Newton's: $m_i\dot{\mathbf{v}}_i = \mathbf{f}_i$

  $\Rightarrow \mathbf{M}_i\dot{\mathbf{t}}_i + \dot{\mathbf{M}}_i\mathbf{t}_i = \mathbf{w}_i$

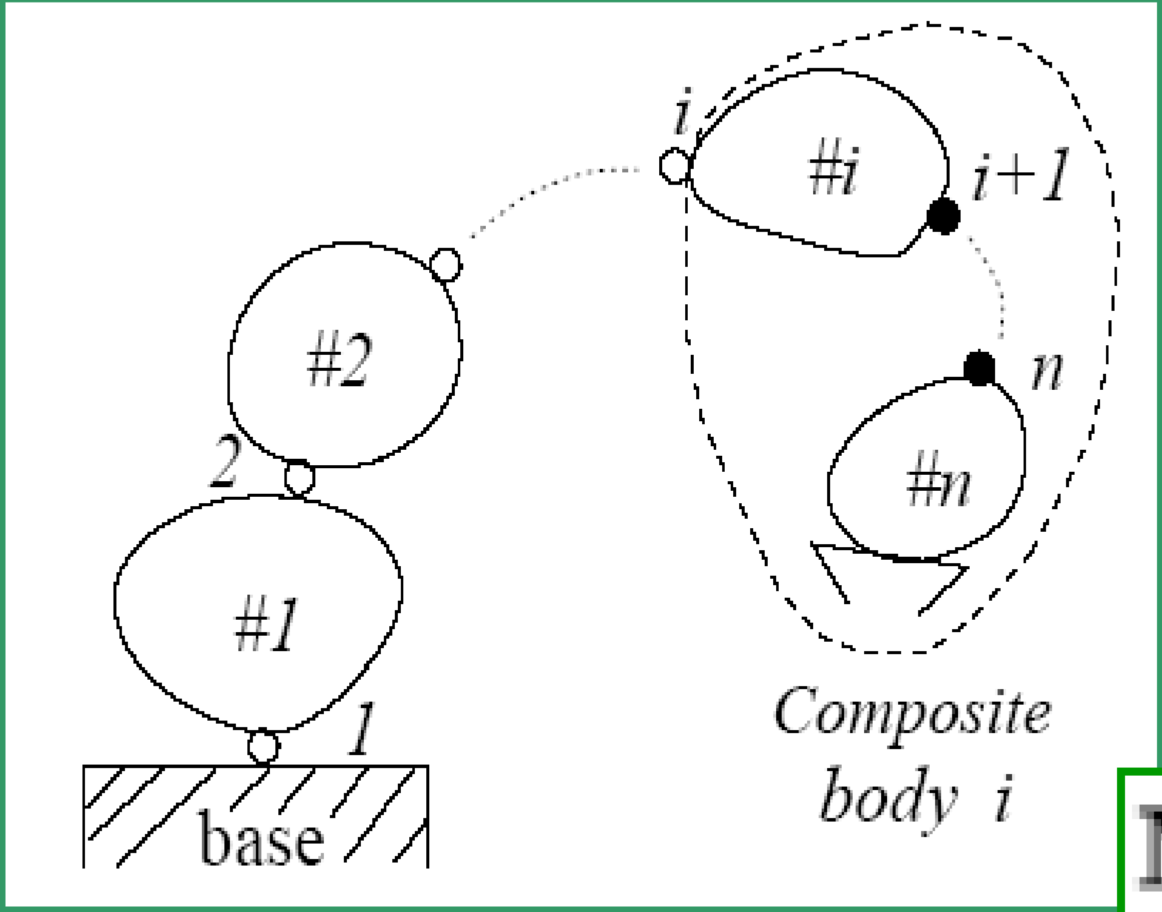
- Euler-Lagrange (EL)

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\boldsymbol{\theta}}}\right) - \frac{\partial L}{\partial \boldsymbol{\theta}} = \boldsymbol{\tau}$$

- Kane's, Hamilton's …
- Orthogonal Complement based, e.g., *Decoupled Natural Orthogonal Complement (DeNOC)*



Uncoupled NE Equations



$$\mathbf{M} \equiv \text{diag}[\mathbf{M}_1, \cdots, \mathbf{M}_n]$$

$$\dot{\mathbf{t}} \equiv [\dot{\mathbf{t}}_1^T, \cdots, \dot{\mathbf{t}}_n^T]^T$$

$$\dot{\mathbf{M}} \equiv \text{diag}[\dot{\mathbf{M}}_1, \cdots, \dot{\mathbf{M}}_n]$$

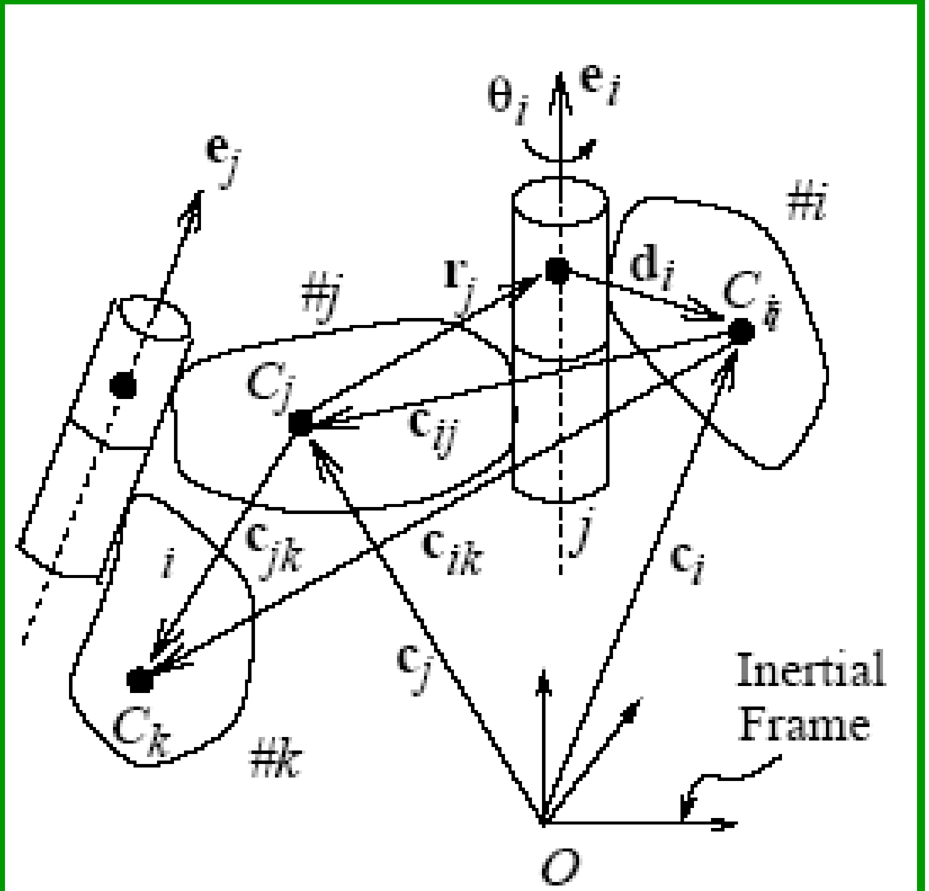
$$\mathbf{t} \equiv [\mathbf{t}_1^T, \cdots, \mathbf{t}_n^T]^T$$

$$\mathbf{w} \equiv [\mathbf{w}_1^T, \cdots, \mathbf{w}_n^T]^T$$

$$\mathbf{M}\dot{\mathbf{t}} + \dot{\mathbf{M}}\mathbf{t} = \mathbf{w}$$

- The 6*n* uncoupled equations of motion

Kinematic Constraints: DeNOC Matrices



$$\boldsymbol{\omega}_i = \boldsymbol{\omega}_j + \dot{\theta}_i \mathbf{e}_i$$

$$\mathbf{v}_i = \mathbf{v}_j + \boldsymbol{\omega}_j \times \mathbf{r}_j + \boldsymbol{\omega}_i \times \mathbf{d}_i$$

$$\mathbf{t}_i = \mathbf{B}_{ij}\mathbf{t}_j + \mathbf{p}_i \dot{\theta}_i$$

$$\mathbf{B}_{ij}\mathbf{B}_{jk} = \mathbf{B}_{ik}$$

$$\mathbf{B}_{ii} = \mathbf{1}, \quad \text{and} \quad \mathbf{B}_{ij}^{-1} = \mathbf{B}_{ji}$$

$\mathbf{B}_{ij}$: the $6n \times 6n$ twist-propagation matrix

$\mathbf{p}_i$: the $6n$-dimensional joint-rate propagation vector or twist generator

Definition: DeNOC Matrices

$$\mathbf{t} \equiv [\mathbf{t}_1^T, \cdots, \mathbf{t}_n^T]^T \quad \dot{\boldsymbol{\theta}} \equiv [\dot{\theta}_1, \cdots, \dot{\theta}_n]^T$$

$$\mathbf{t} = \mathbf{N}\dot{\boldsymbol{\theta}}, \qquad \text{where} \quad \mathbf{N} \equiv \mathbf{N}_l\mathbf{N}_d$$

$$\mathbf{N}_l \equiv \begin{bmatrix} \mathbf{1} & \mathbf{O} & \cdots & \mathbf{O} \\ \mathbf{B}_{21} & \mathbf{1} & \cdots & \mathbf{O} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{B}_{n1} & \mathbf{B}_{n2} & \cdots & \mathbf{1} \end{bmatrix} \quad \text{and} \quad \mathbf{N}_d \equiv \begin{bmatrix} \mathbf{p}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{p}_2 & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{p}_n \end{bmatrix}$$

- **$\mathbf{N}\equiv\mathbf{N}_l\mathbf{N}_d$:** the $6n \times n$ Decoupled Natural Orthogonal Complement

Coupled Equations

$$\mathbf{N}^T(\mathbf{M}\dot{\mathbf{t}} + \dot{\mathbf{M}}\mathbf{t}) = \mathbf{N}^T(\mathbf{w}^W + \mathbf{w}^C) \quad \Leftarrow \quad \mathbf{t}^T\mathbf{w}^C = \dot{\boldsymbol{\theta}}^T\mathbf{N}^T\mathbf{w}^C = 0,$$

$$\Downarrow$$

$$\mathbf{N}^T(\mathbf{M}\dot{\mathbf{t}} + \dot{\mathbf{M}}\mathbf{t}) = \mathbf{N}^T\mathbf{w}^W$$

$$\Downarrow$$

$$\boxed{\mathbf{I}\ddot{\theta} + \mathbf{C}\dot{\theta} = \boldsymbol{\tau}}$$

$$\mathbf{I} \equiv \mathbf{N}^T\mathbf{M}\mathbf{N} \equiv \mathbf{N}_d^T\tilde{\mathbf{M}}\mathbf{N}_d$$

$$\mathbf{C} \equiv \mathbf{N}^T(\mathbf{M}\dot{\mathbf{N}} + \dot{\mathbf{M}}\mathbf{N}) \equiv \mathbf{N}_d^T\tilde{\mathbf{M}}'\mathbf{N}_d$$

$$\boldsymbol{\tau} \equiv \mathbf{N}^T\mathbf{w}^W \equiv \mathbf{N}_d^T\tilde{\mathbf{w}}^W;$$

- *n* coupled Euler-Lagrange equations
  - *no partial differentiation*

Recursive Expressions

- For the $n \times n$ GIM, each element

$$I_{ij} = I_{ji} = \mathbf{p}_i^T \tilde{\mathbf{M}}_i \mathbf{B}_{ij} \mathbf{p}_j$$

$$\tilde{\mathbf{M}}_i = \mathbf{M}_i + \mathbf{B}_{i+1,i}^T \tilde{\mathbf{M}}_{i+1} \mathbf{B}_{i+1,i} \quad \text{where} \quad \tilde{\mathbf{M}}_n \equiv \mathbf{M}_n$$

<u>Composite body mass matrix</u>

- For the $n \times n$ MCI, each element

$$C_{ij} = \begin{cases} \mathbf{p}_i^T (\mathbf{B}_{ji}^T \tilde{\mathbf{M}}_j \mathbf{W}_j + \mathbf{B}_{j+1,i}^T \tilde{\mathbf{H}}_{j+1,j} + \dot{\tilde{\mathbf{M}}}_j) \mathbf{p}_j & \text{if } i \leq \tilde{\mathbf{M}}_{n+1} = \tilde{\mathbf{H}}_{n+1,n} = \mathbf{O} \\ \mathbf{p}_i^T (\tilde{\mathbf{M}}_i \mathbf{B}_{ij} \mathbf{W}_j + \tilde{\mathbf{H}}_{ij} + \dot{\tilde{\mathbf{M}}}_i) \mathbf{p}_j & \text{otherwise} \end{cases}$$

- For the $n \times n$ generalized forces

$$\tau_i = \mathbf{p}_i^T \tilde{\mathbf{w}}_i^W \qquad \tilde{\mathbf{w}}_i^W = \mathbf{w}_i^W + \mathbf{B}_{i+1,i}^T \tilde{\mathbf{w}}_{i+1}^W$$

Inverse Dynamics Algorithm

Saha (1999): ASME

Forward Recursion

$$\boldsymbol{\nu}_1 = \mathbf{p}_1\dot{\theta}_1;$$
$$\boldsymbol{\nu}_2 = \mathbf{p}_2\dot{\theta}_2 + \boldsymbol{\nu}_1;$$
$$\vdots$$
$$\boldsymbol{\nu}_n = \mathbf{p}_n\dot{\theta}_n + \boldsymbol{\nu}_{n-1};$$

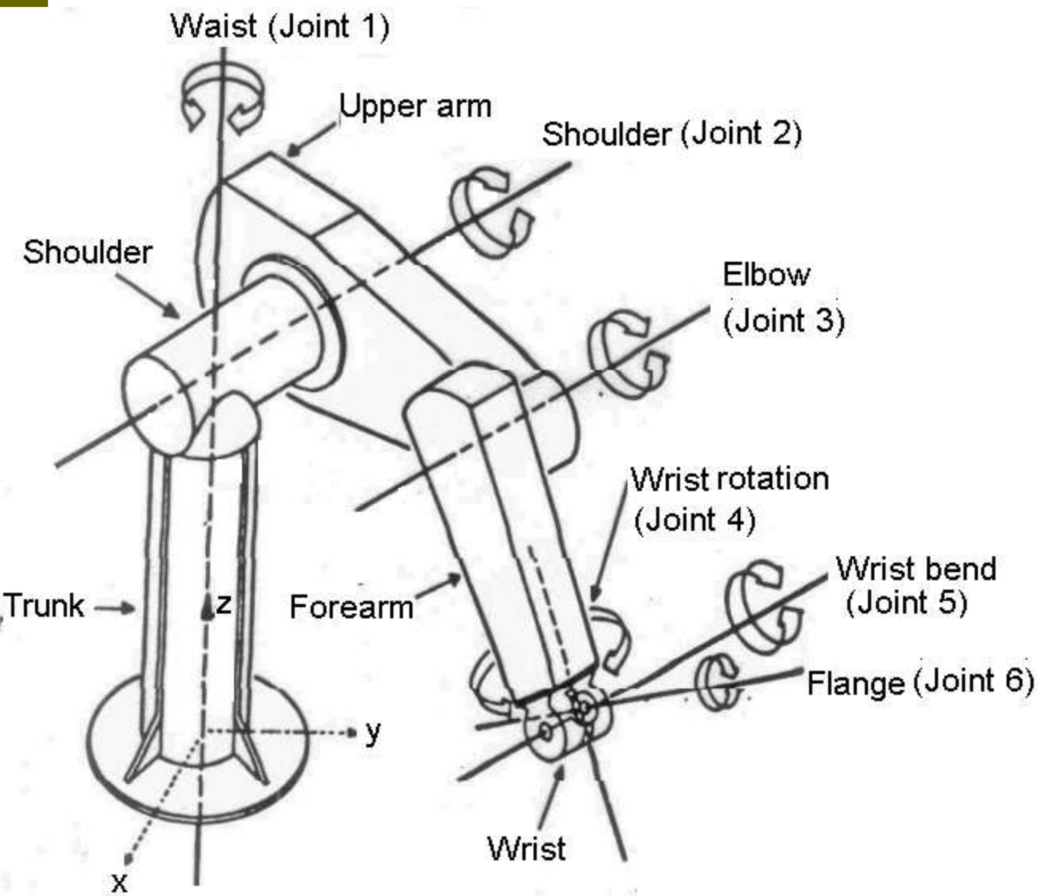
$$\boldsymbol{\xi}_1 = \mathbf{p}_1\ddot{\theta}_1 + \mathbf{W}_1\mathbf{p}_1\dot{\theta}_1$$
$$\boldsymbol{\xi}_2 = \mathbf{p}_2\ddot{\theta}_2 + \mathbf{W}_2\mathbf{p}_2\dot{\theta}_2 + \mathbf{B}_{21}\boldsymbol{\xi}_1 + \dot{\mathbf{B}}_{21}\boldsymbol{\nu}_1$$
$$\vdots$$
$$\boldsymbol{\xi}_n = \mathbf{p}_n\ddot{\theta}_n + \mathbf{W}_n\mathbf{p}_n\dot{\theta}_n + \mathbf{B}_{n,n-1}\boldsymbol{\xi}_{n-1} + \dot{\mathbf{B}}_{n,n-1}\boldsymbol{\nu}_{n-1}$$

Backward Recursion

$$\boldsymbol{\gamma}_n = \mathbf{M}_n\boldsymbol{\xi}_n + \dot{\mathbf{M}}_n\boldsymbol{\nu}_n;$$
$$\boldsymbol{\gamma}_{n-1} = \mathbf{M}_{n-1}\boldsymbol{\xi}_{n-1} + \dot{\mathbf{M}}_{n-1}\boldsymbol{\nu}_{n-1} + \mathbf{B}_{n,n-1}^T\boldsymbol{\gamma}_n;$$
$$\vdots$$
$$\boldsymbol{\gamma}_1 = \mathbf{M}_1\boldsymbol{\xi}_1 + \dot{\mathbf{M}}_1\boldsymbol{\nu}_1 + \mathbf{B}_{21}^T\boldsymbol{\gamma}_2;$$

$$\tau_n = \mathbf{p}_n^T\boldsymbol{\gamma}_n$$
$$\tau_{n-1} = \mathbf{p}_{n-1}^T\boldsymbol{\gamma}_{n-1}$$
$$\vdots$$
$$\tau_1 = \mathbf{p}_1^T\boldsymbol{\gamma}_1$$

Example: PUMA 560

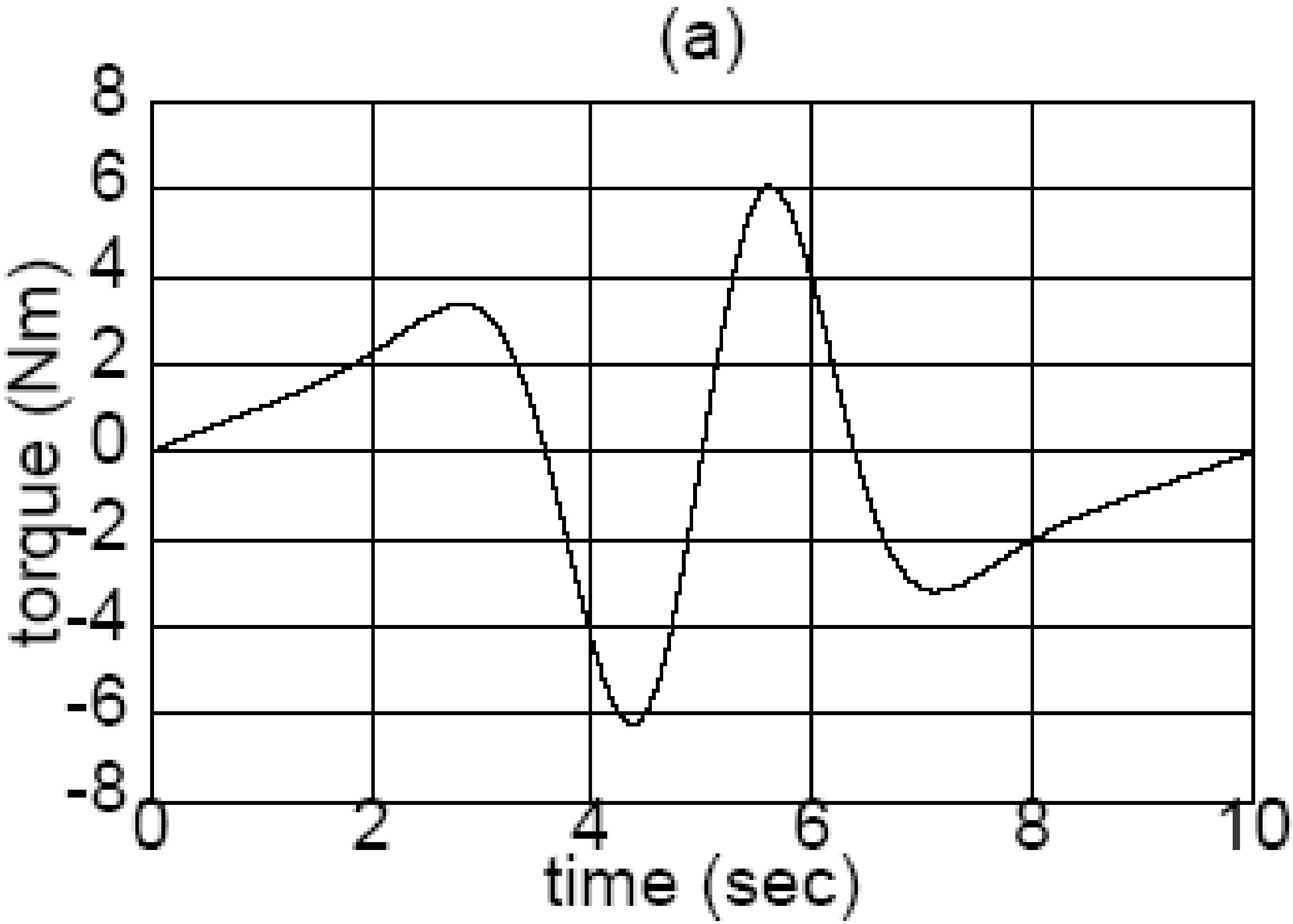


DH Parameters

Link	a (m)	b (m)	α (deg)	θ (deg)
1	0	0	-90	θ_1
2	0.432	0.149	0	θ_2
3	0.02	0	90	θ_3
4	0	0.432	-90	θ_4
5	0	0	90	θ_5
6	0	0.056	0	θ_6

Result: Torque at Joint 1

$$\theta_i = \frac{1}{2}\left[\frac{2\pi}{T}t - \sin\left(\frac{2\pi}{T}t\right)\right] \quad T = 10.0 \text{ sec}; \ \theta_i(0) = 0, \text{ and } \theta_i(T) = 180^o$$



Comparison

Algorithm	M	A	$n=6$	
Hollerbach (1980)	$412n-277$	$320n-201$	$2195M$	$1719A$
Luh et al. (1980)	$150n-48$	$131n+48$	$852M$	$834A$
Walker and Orin (1982)	$137n-22$	$101n-11$	$800M$	$595A$
Proposed	$\underline{120n-44}$	$\underline{97n-55}$	$\underline{676M}$	$\underline{527A}$
Khalil et al. (1986)	$105n-92$	$94n-86$	$538M$	$478A$
Angeles et al. (1989)	$105n-109$	$90n-105$	$521M$	$435A$
Balafoutis et al. (1988)	$93n-69$	$81n-65$	$489M$	$421A$

Forward Dynamics & Simulation

$$\mathbf{I}\ddot{\theta} + \mathbf{C}\dot{\theta} = \tau$$

$$\mathbf{I} \equiv \mathbf{UDU}^T \qquad \phi \equiv \tau - \mathbf{C}\dot{\theta}$$

$$\mathbf{UDU}^T\ddot{\theta} = \phi$$

$$\hat{\tau} = \mathbf{U}^{-1}\phi$$

$$\tilde{\tau}_i = \hat{\tau}_i / \hat{m}_i$$

$$\ddot{\theta} = \mathbf{U}^{-T}\tilde{\tau}$$

$$\hat{\tau}_i = \phi_i - \mathbf{p}_i^T \eta_{i,i+1} \quad \eta_{i,i+1} \equiv \mathbf{B}_{i+1,i}^T \eta_{i+1} \quad \eta_{i+1} \equiv \psi_{i+1}\hat{\tau}_{i+1} + \eta_{i+1,i+2} \quad \hat{\tau}_n \equiv \phi_n$$

$$\ddot{\theta}_i = \tilde{\tau}_i - \psi_i^T \mu_{i,i-1} \quad \mu_{i-1} \equiv \mathbf{p}_{i-1}\ddot{\theta}_{i-1} + \mu_{i-1,i-2} \quad \mu_{i,i-1} \equiv \mathbf{B}_{i,i-1}\mu_{i-1} \quad \mu_{1,0} = 0$$

Articulated body matrix

$$\hat{\psi}_k \equiv \hat{\mathbf{M}}_k \mathbf{p}_k, \qquad \hat{\psi}_{ik} \equiv \mathbf{B}_{ki}^T \hat{\psi}_k, \qquad \hat{m}_k \equiv \mathbf{p}_k^T \hat{\psi}_k,$$

$$\psi_k \equiv \frac{\hat{\psi}_k}{\hat{m}_k}, \quad \text{and} \quad \psi_{ik} \equiv \frac{\hat{\psi}_{ik}}{\hat{m}_k}$$

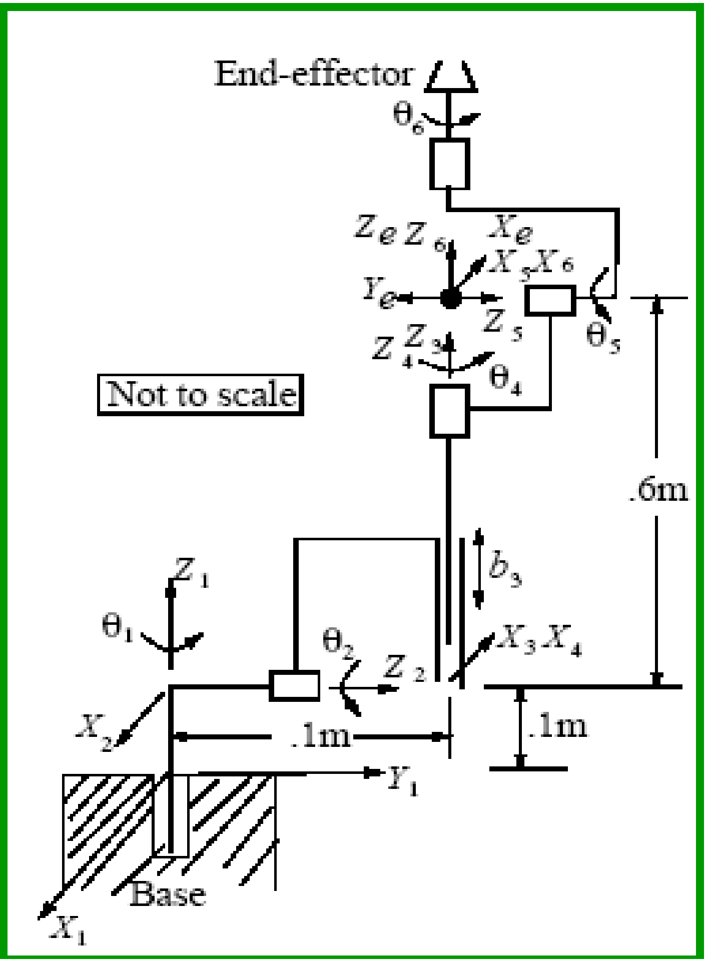
$$\hat{\mathbf{M}}_{ik} = \mathbf{M}_i + \mathbf{B}_{i+1,i}^T \hat{\mathbf{M}}_{i+1,k} \mathbf{B}_{i+1,i}$$

Comparison

Algorithm	M	A
Proposed	$\underline{191n - 284}$	$\underline{187n - 325}$
Featherstone (1983)	$199n - 198$	$174n - 173$
Valasek†	$226n - 343$	$206n - 345$
Brandl et al. †	$250n - 222$	$220n - 198$
Walker and Orin (1982)	$\frac{1}{6}n^3 + 11\frac{1}{2}n^2 + 38\frac{1}{3}n - 47$	$\frac{1}{6}n^3 + 7n^2 + 38\frac{5}{6}n - 46$

$n = 6$	$n = 10$
$\underline{862M\ 797A}$	$\underline{1626M\ 1545A}$
$996M\ 871A$	$1792M\ 1567A$
$1013M\ 891A$	$1917M\ 1715A$
$1278M\ 1122A$	$2278M\ 2002A$
$633M\ 480A$	$1653M\ 1209A$

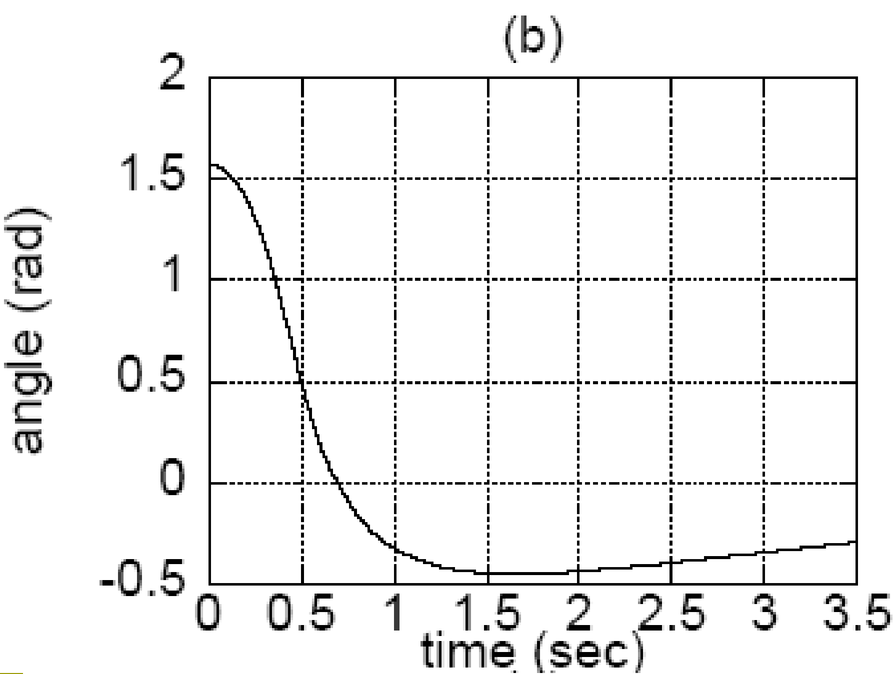
Results: Stanford Robot

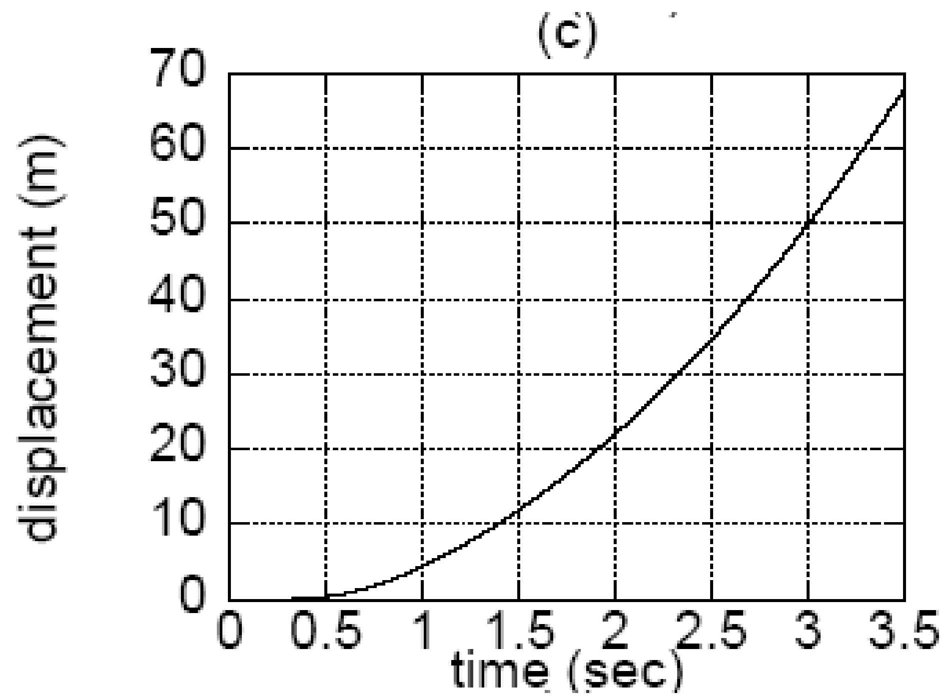


DH and Inertia Parameters

i	a_i (m)	b_i (m)	α_i (deg)	θ_i (deg)	m_i (kg)	r_x (m)	r_y	r_z	I_{xx} (kg-m^2)	I_{yy}	I_{zz}
1	0	.1	-90	θ_1[0]	9	0	-.1	0	.01	.02	.01
2	0	.1	-90	θ_2[90]	6	0	0	0	.05	.06	.01
3	0	b_3[0]	0	0	4	0	0	0	.4	.4	.01
4	0	.6	90	θ_4[0]	1	0	-.1	0	.001	.001	.0005
5	0	0	-90	θ_5[0]	.6	0	0	0	.0005	.0005	.0002
6	0	0	0	θ_6[0]	.5	0	0	0	.003	.001	.002

Free-Fall: Joints 2 and 3

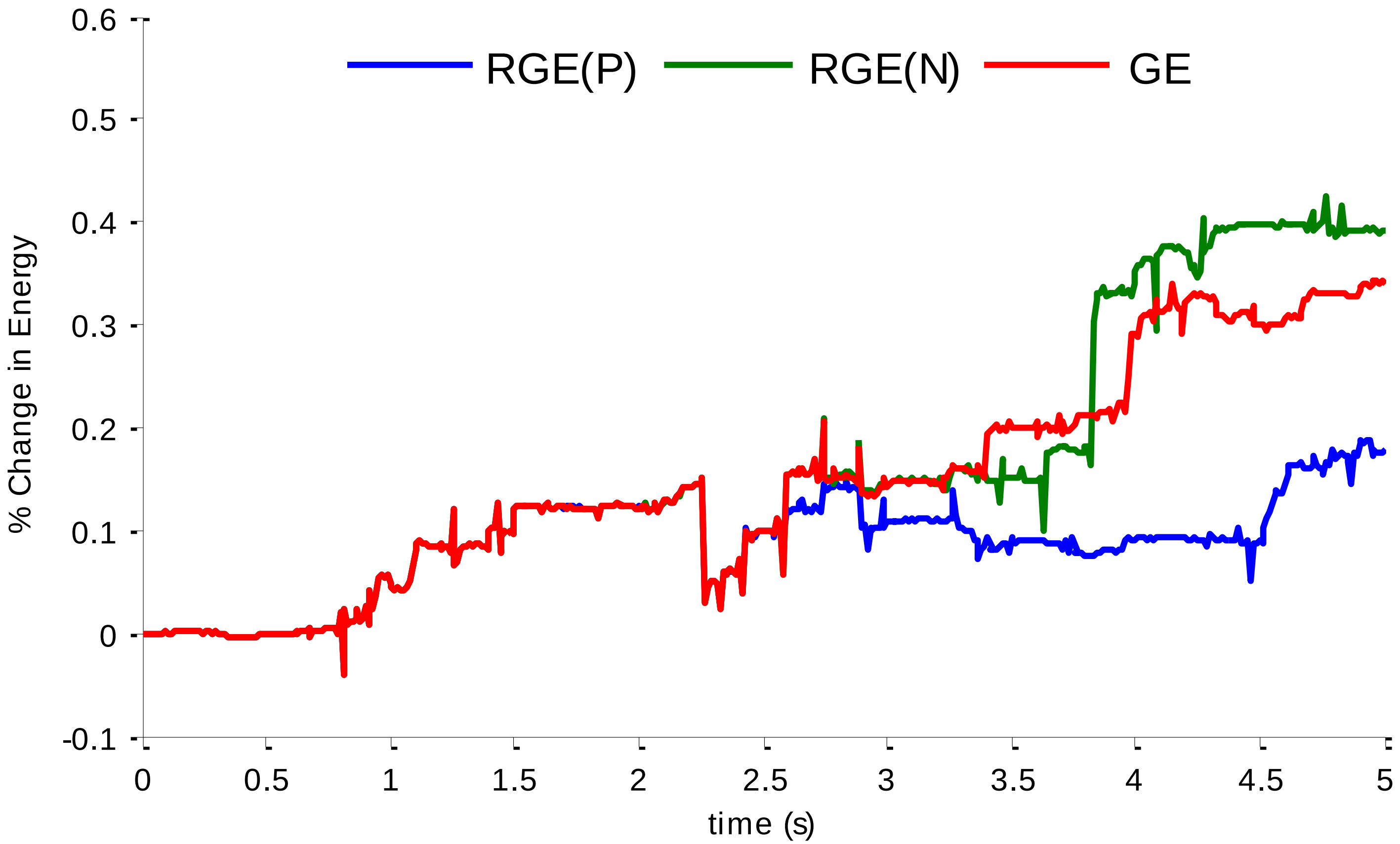




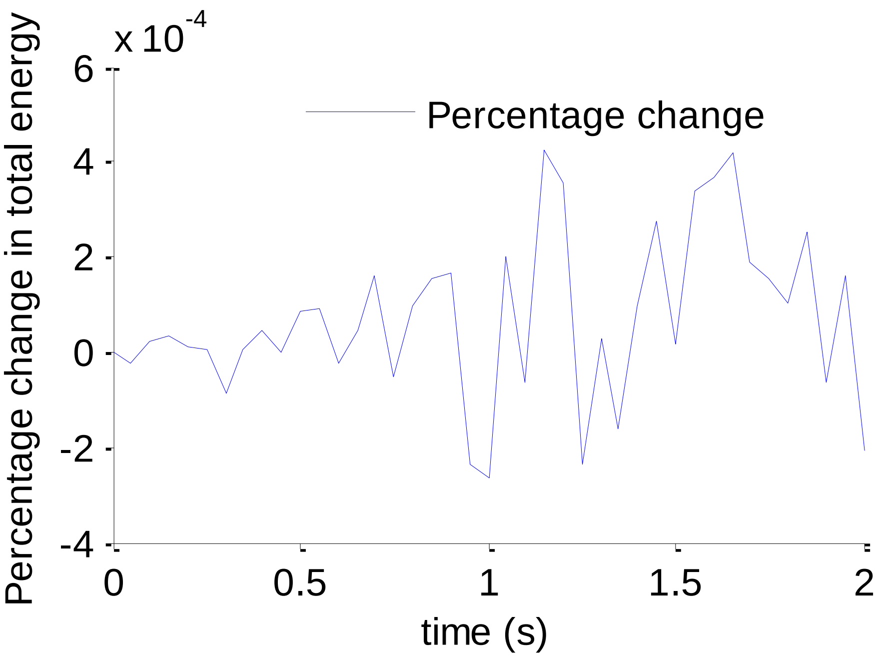
Robot Software: RoboAnalyzer
www.roboanalyzer.com

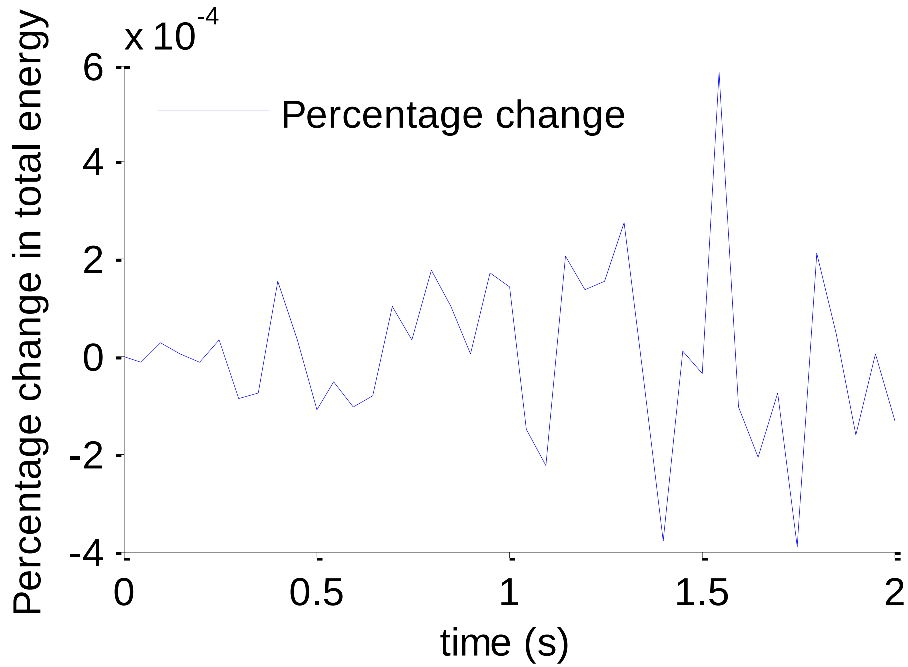
Why UDU^T?

- Accurate (based on Reverse Gaussian Elimination) [Plot for 3-link system]



Efficient

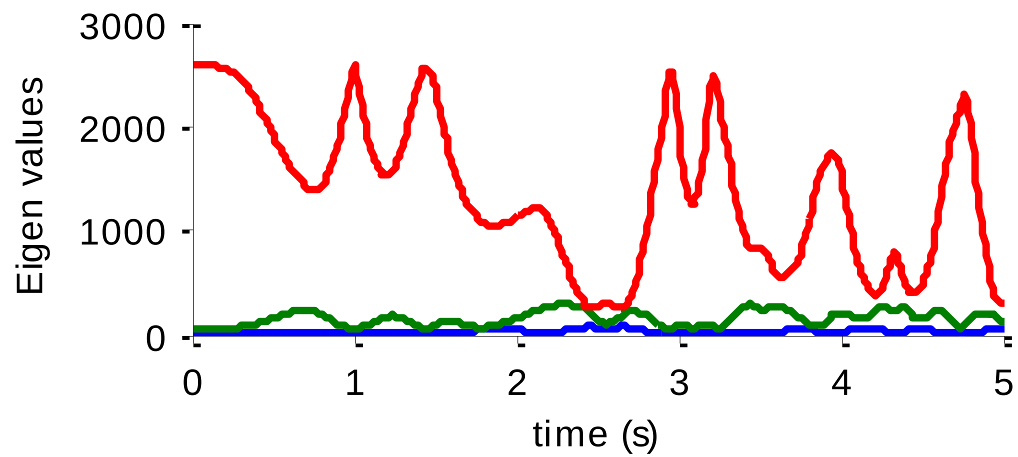




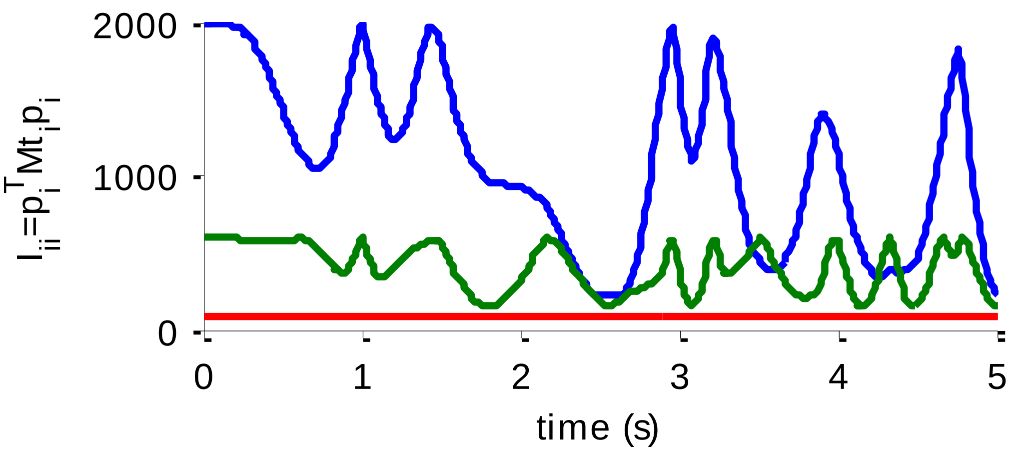
Rel. Tol: 10^{-4}; Abs. Tol: 10^{-6}	$O(n)$	$O(n^3)$
RE-U750	1330 s (22min)	606269 s (7 days)

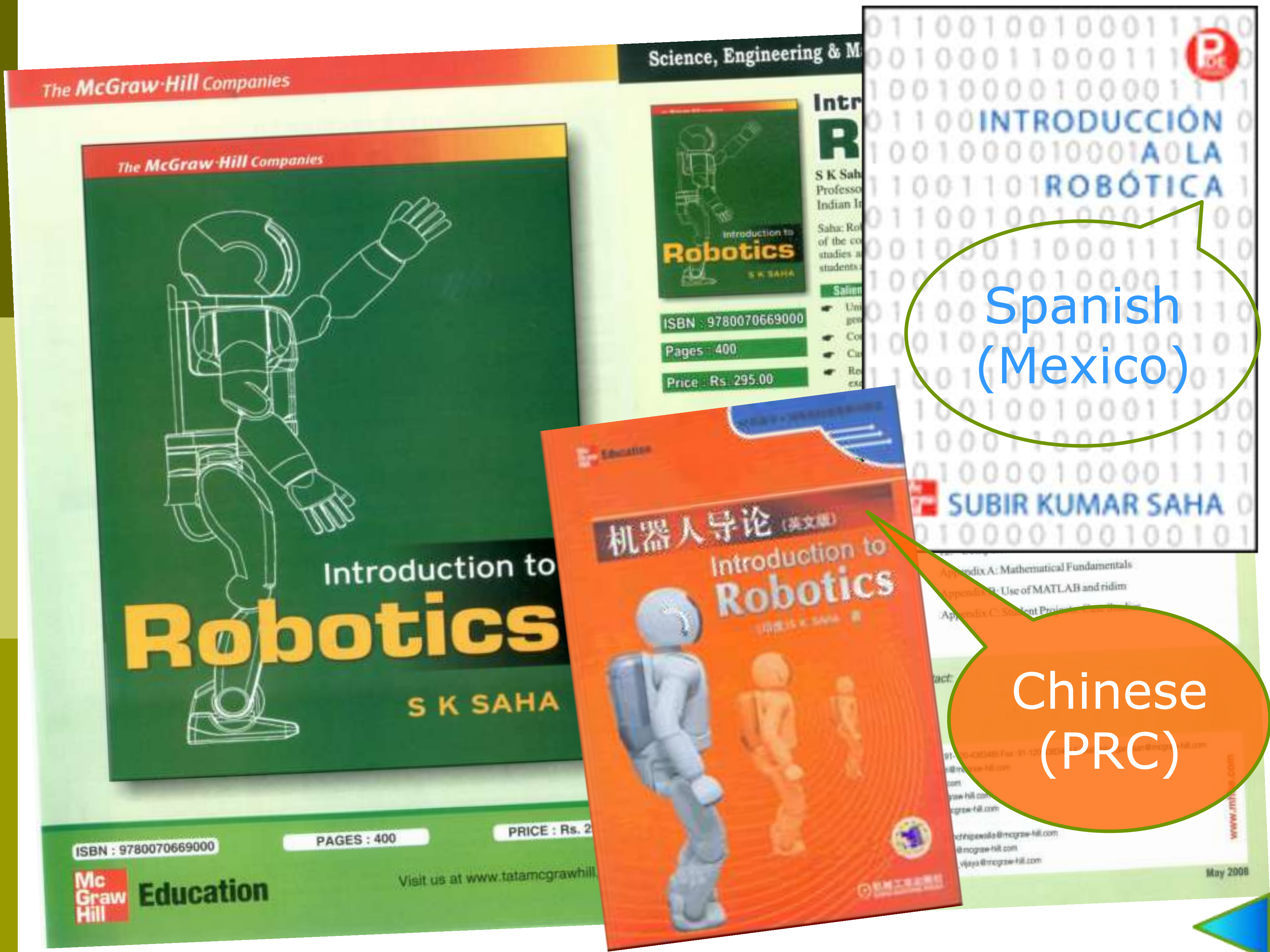
Information on Stability

- Eigenvalues of the GIM

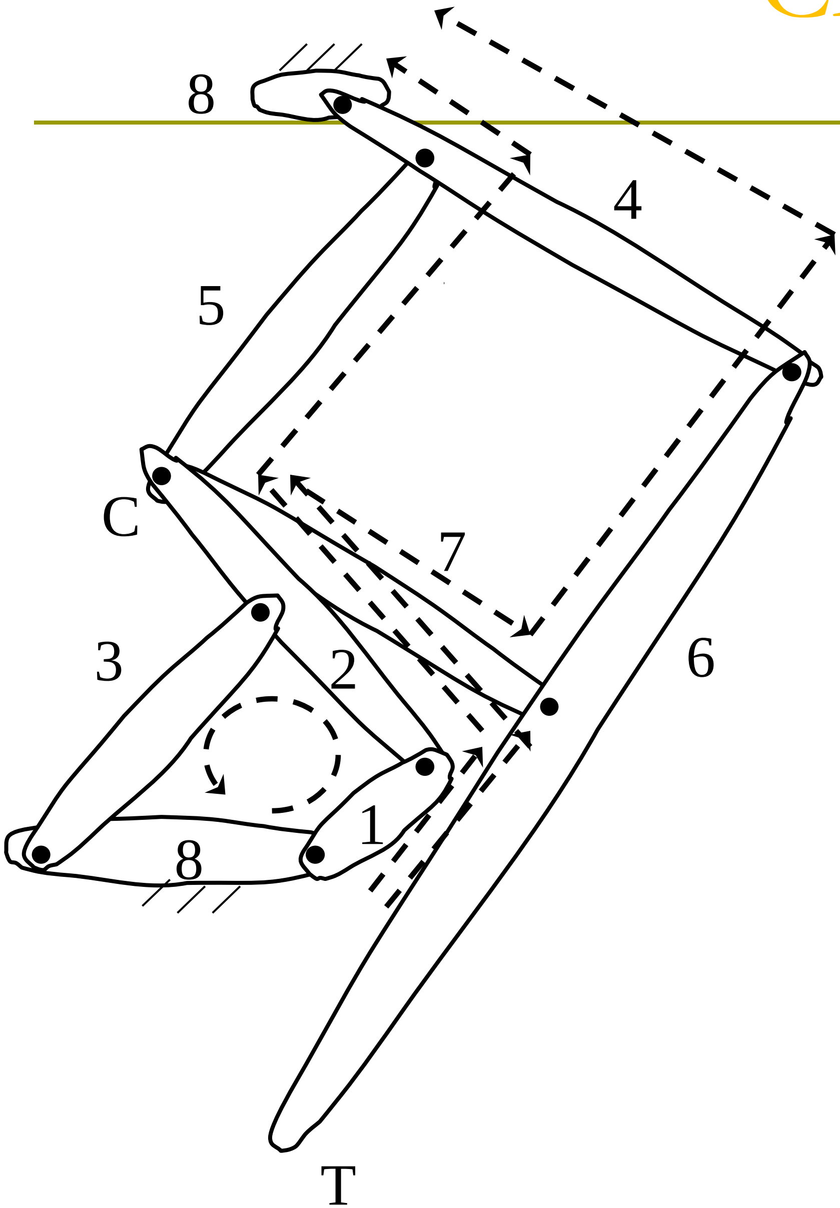


- Diagonal elements of the GIM




The McGraw-Hill Companies
Introduction to
Robotics
S K SAHA
ISBN : 9780070669000
PAGES : 400
Science, Engineering & M
ISBN : 9780070669000
Pages : 400
Price : Rs. 295.00
Mc Graw Hill Education
INTRODUCCIÓN
A LA
ROBÓTICA
SUBIR KUMAR SAHA
Spanish (Mexico)
机器人导论 (英文版)
Introduction to
Robotics
Chinese (PRC)
May 2008

Closed-loop Systems

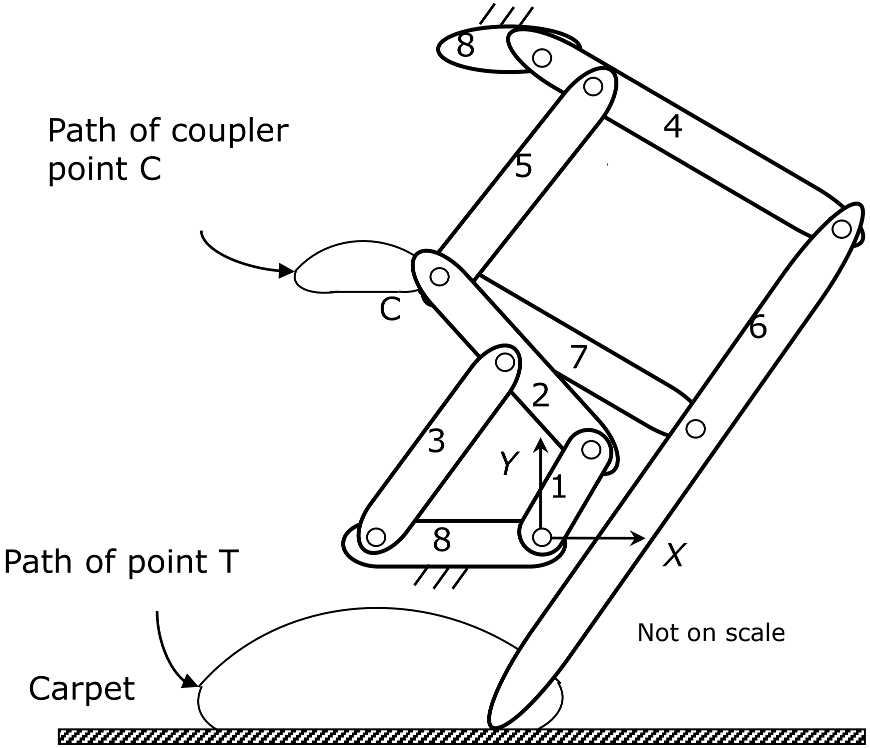


- A Mechanism used for walking legs
- Consisting of two closed-loop linkages
- It has several loops
- Applied also in Carpet Scrapper

Carpet Cleaning: Traditional

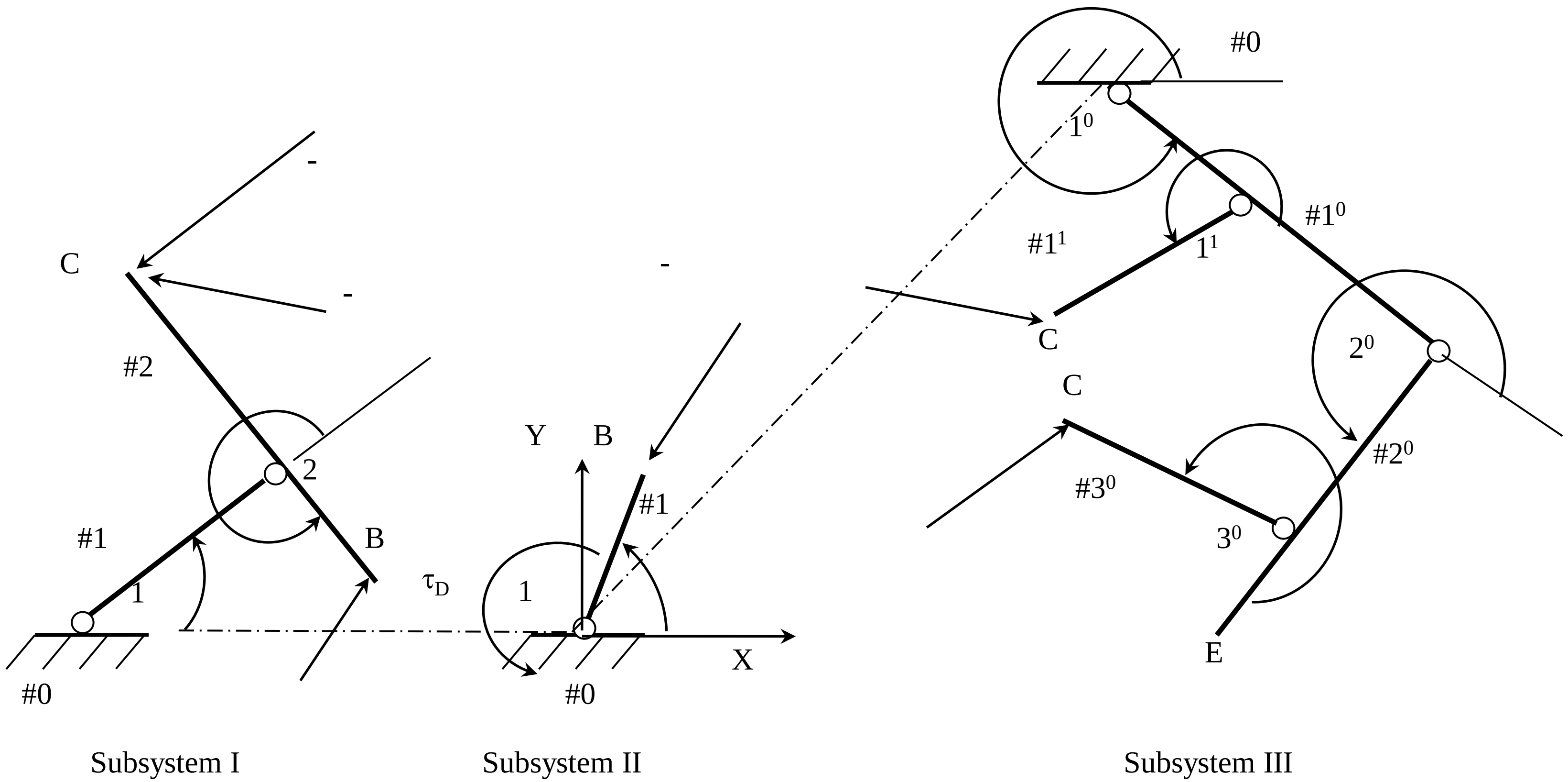
Carpet Scrapping Machine

- Purpose: To reduce human effort
- Straight line generating machine



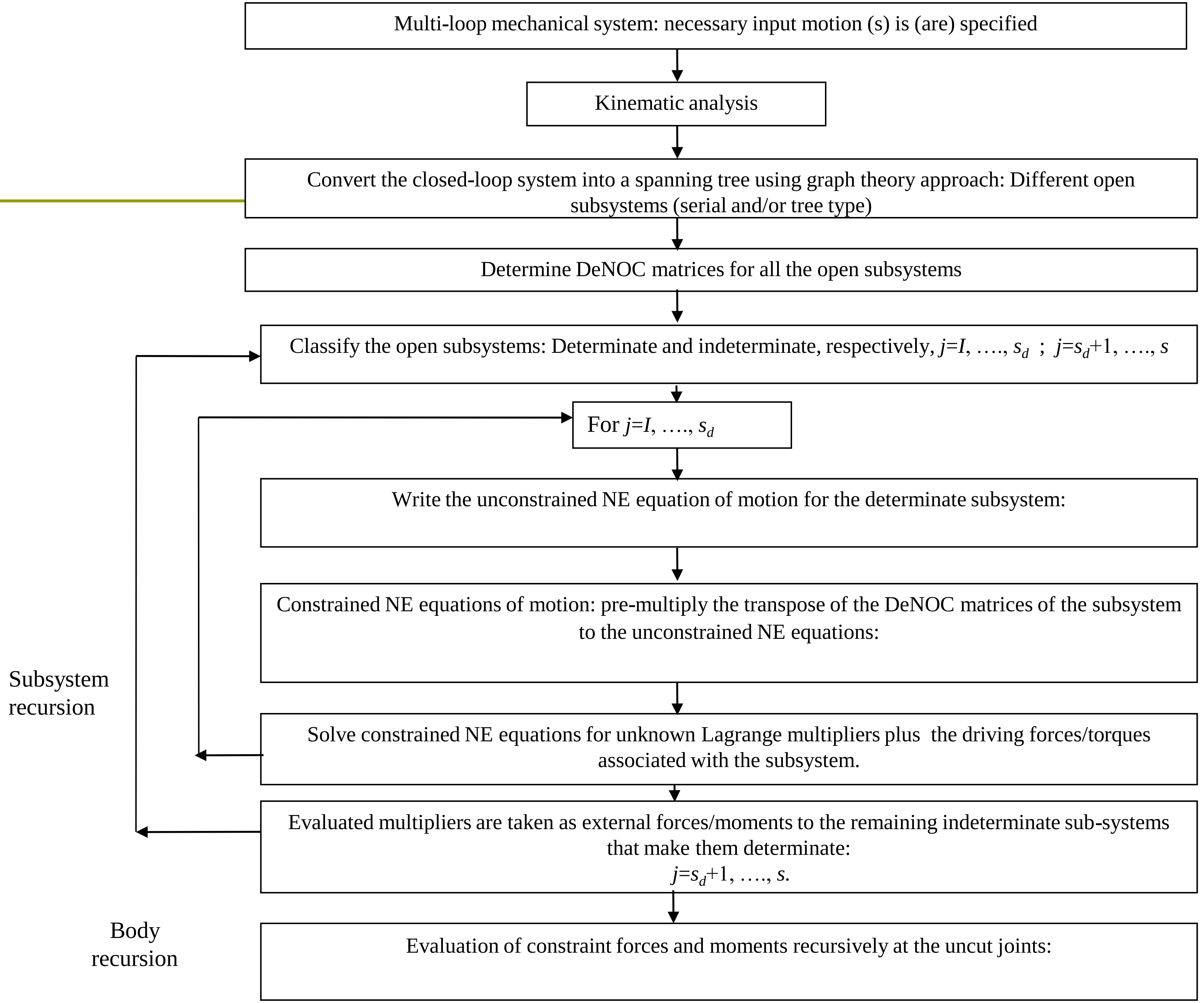
Tree-types: Double Recursion

ASME J. of Mech. Des., Dec. 2007



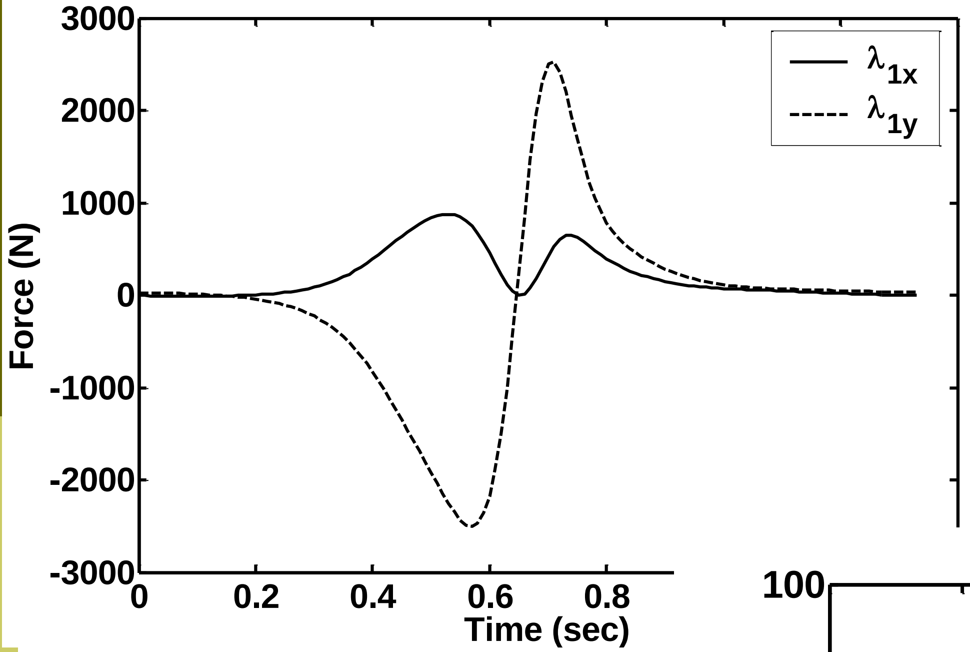
- Unknowns: 6+3 & Eqs. 2+1
- Unsolvable independently (Indeterminate subsystems)

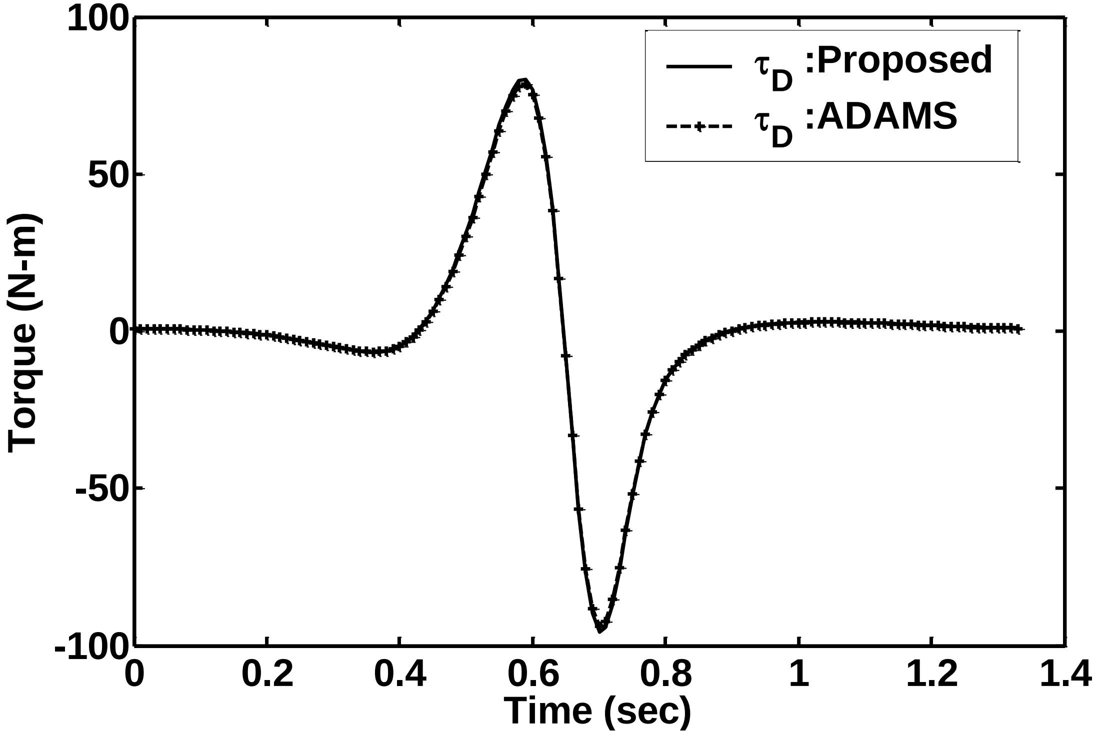
- Unknowns: 4 & Eqs.: 4
- Solvable independently (determinate subsystem)



Multi-loop mechanical system: necessary input motion (s) is (are) specified
Kinematic analysis
Convert the closed-loop system into a spanning tree using graph theory approach: Different open subsystems (serial and/or tree type)
Determine DeNOC matrices for all the open subsystems
Classify the open subsystems: Determinate and indeterminate, respectively, $j=I, \ldots, s_d$; $j=s_d+1, \ldots, s$
For $j=I, \ldots, s_d$
Write the unconstrained NE equation of motion for the determinate subsystem:
Constrained NE equations of motion: pre-multiply the transpose of the DeNOC matrices of the subsystem to the unconstrained NE equations:
Solve constrained NE equations for unknown Lagrange multipliers plus the driving forces/torques associated with the subsystem.
Evaluated multipliers are taken as external forces/moments to the remaining indeterminate sub-systems that make them determinate:
$j=s_d+1, \ldots, s$.
Evaluation of constraint forces and moments recursively at the uncut joints:
Subsystem recursion
Body recursion

Results





ADAMS Animation

Comparison

Methods	Theoretical order of computations	CPU time in sec
Traditional (matrix size: 21×21)	$O(21^3/3)=O(3087)$	0.219
System approach (matrix size: 7×7)	$O(7^3/3)+O(7\times2)=O(128.3)$	0.156 (30.59)
Subsystem approach (matrix sizes: 4×4, 3×3, 1×1)	$O(4^3/3+3^3/3+1)+O(7\times2)=O(45.3)$	0.156 (30.59)

Multibody Dynamics for Rural Applications (MuDRA)

- Areas of Research: **Mu**ltibody **D**ynamics
  - Robotics
  - Mechatronics
  - Design
- Developed Mechanisms: **R**ural **A**pplications
  - For carpet processing
  - For villages ADPM

MuDRA Concept

- Floated as B. Tech/M. Tech Projects
  - Not interested
- Apparent Reasons
  - What is the research content ?
  - Not fashionable
- Other Reasons
  - Limited literature
  - Difficult

Springer
the language of science

springer.com

Dynamics and Balancing of Multibody Systems
Series: Lecture Notes in Applied and Computational Mechanics , Vol. 37

Chaudhary, Himanshu, **Saha**, Subir Kumar
2009, XIV, 176 p. 59 illus., Hardcover
ISBN: 978-3-540-78178-3
Online version available

Online orders shipping within 2-3 days.

99,95 €

Another Book, €100

This book has evolved from the passionate desire of the authors in using the modern concepts of multibody dynamics for the design improvement of the machineries used in the rural sectors of India and The World. In this connection, the first author took up his doctoral research in 2003 whose findings have resulted in this book. It is expected that such developments

Every drop counts

PETER FRYKMAN, A STUDENT OF STANFORD UNIVERSITY, HAS INVENTED A MANUFACTURING TECHNOLOGY TO HELP MAKE AFFORDABLE DRIP IRRIGATION FOR SMALL-PLOT FARMERS IN DEVELOPING COUNTRIES. MALINI SEN REPORTS

A third of the world's population suffers from water scarcity. Without access to affordable water efficient irrigation, small-plot farmers are unable to grow crops during much of the year.

As part of a Stanford University course in Entrepreneurial Design for Extreme Affordability, Peter Frykman travelled to Ethiopia, where he witnessed first-hand the hardships caused by the worst drought Ethiopia had experienced in 20 years. The farmers he met had no means to grow crops with their scarce water resources. Locally available drip irrigation products were too expensive and seldom worked properly.

Recognising the need for less costly and more effective ways for small-plot farmers to use their meagre water supplies efficiently, Frykman returned to Stanford and invented a manufacturing technology, and launched his company, Driptech, which makes affordable drip irrigation for small-plot farmers in developing countries.

Describing the technology behind Driptech, Frykman says there are two important parts to the technology. "The first part is the attention we paid to small-scale farmers and how they work that allowed us to design the system specifically to meet their needs. We learned the importance of making our systems simple in order to reduce installation and maintenance costs. We learned that highly uniform water application is essential, even with low pressure, so we designed Driptech tubing with the very precise, clean, consistent holes that make this possible. We learned that small plots come in many configurations, and made our system modular by designing parts that continue to function well even if it scales up or down," he elaborates.

Stanford graduate Peter Frykman explains his invention to farmers in India

CASE STUDY

The second part is that Driptech is able to manufacture this product anywhere in the world using existing plastic bag machinery. "We developed small, inexpensive machines that reliably produce tubing to the high level of quality that we require. This 'distributed manufacturing' model allows us to customise products to meet local needs, minimise transportation costs, and create jobs in nearby communities," adds Frykman.

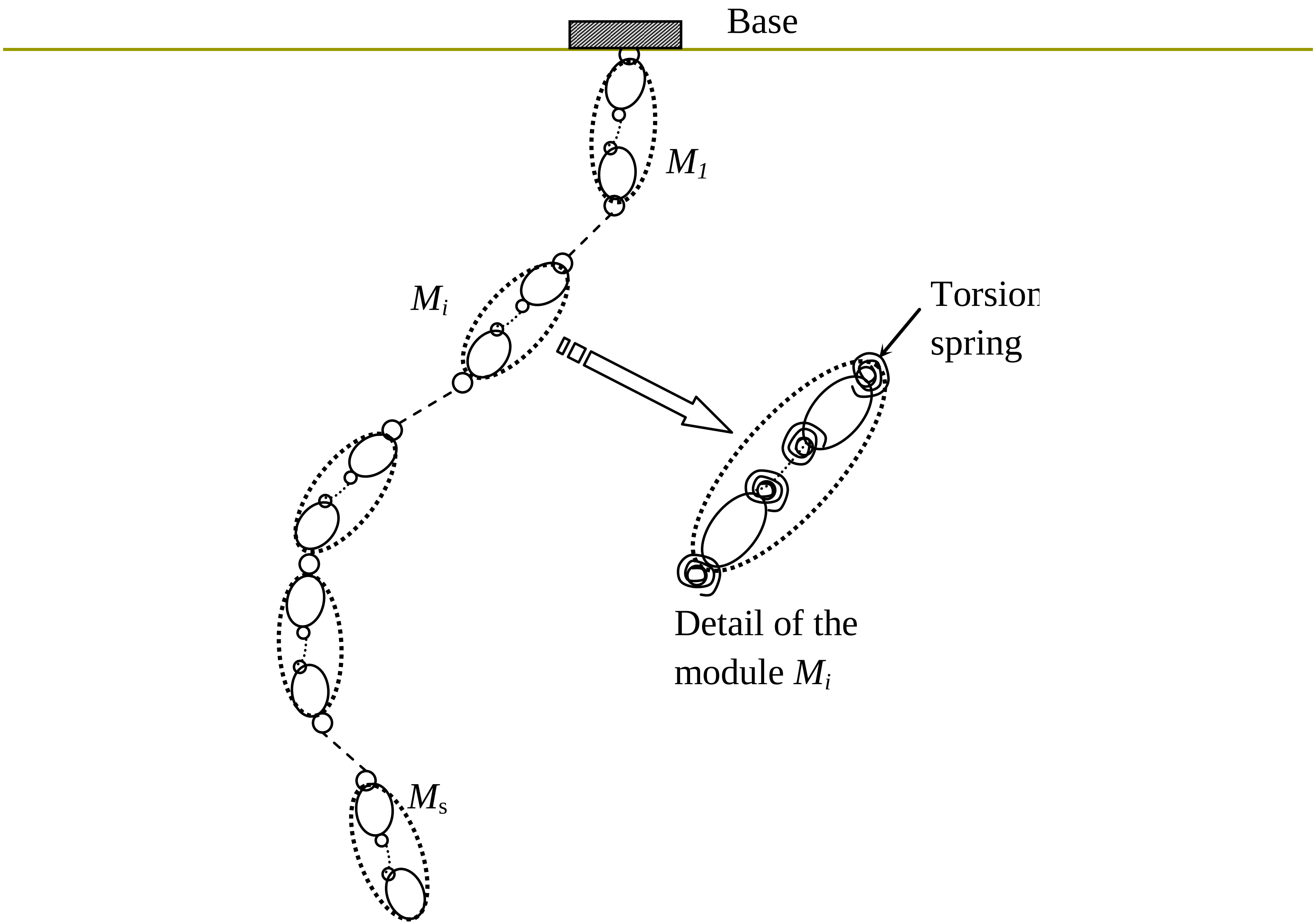
Compared to flood irrigation, Driptech irrigation increases crop yields by 20% to 90%, improves product quality, saves water by 30% to 70%, and reduces the required labour, energy costs for pumping, and fertiliser. Driptech systems operate on very low water pressures as well.

As a graduate student in mechanical engineering at Stanford, Frykman attended the Summer Institute for Entrepreneurship, and he found the programme "exhilarating" and "highly beneficial."

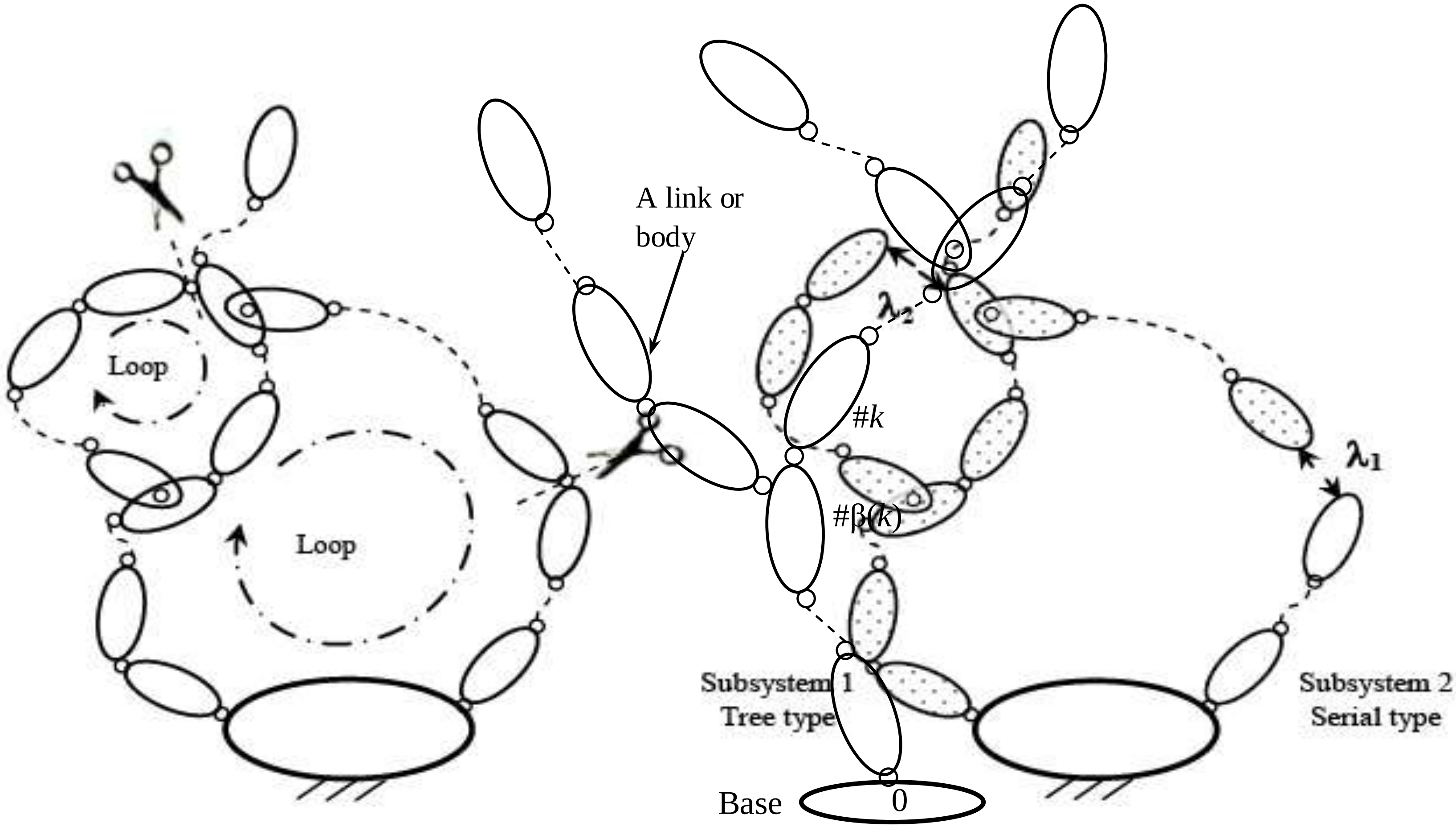
"I applied for the programme because I needed help developing my underlying passion for entrepreneurship. I relied on lessons that I had learned during the programme to launch my company Driptech. Even without a formal business background I had the confidence to launch my own venture and lead it."

Currently Frykman and his team are working with farmers in Maharashtra and Karnataka and last year, Driptech launched commercially in India by partnering with one of the largest manufacturing and retail conglomerates in the country. Driptech's systems sell directly to farmers through retail and local distribution channels.

Ropes (Courtesy: Dr. Suril V. Shah)



Multibody Systems: ReDySim



Biological Systems: Study of Proteins

- Proteins (Nanoparticles: 1–100 nm) are part of every cell, tissue, and organ in our bodies.
- Body proteins are constantly being broken down and replaced.
- The protein in the foods we eat is digested into amino acids that are later used to replace these proteins in our bodies.

Protein

- Proteins are made up of amino acids.
- Think of amino acids as the building blocks.
- There are 20 different amino acids that join together to make all types of protein.
- Some of these amino acids can't be made by our bodies, so these are known as *essential* amin o acids.
- It's *essential* that our diet provide these.

"Analysis and Design of Protein Based Nanodevices: Challenges and Opportunities in Mechanical Design" by Gregory S. Chirikjian

[ASME J. Mech. Design, July 2005, Vol. 127]

- Each of these amino acid building blocks (monomers) is referred to as a residue.

- Tens to hundreds of these residues amino acid monomers connect together in a serial manner to create a long chain, known as a polypeptide chain.

Protein vs. Serial Robot

- From a kinematics point of view, these polypeptide molecules can be considered to be a chain of miniature rigid bodies connected by revolute hinge joints.

- A protein in its denatured state is a serial linkage with *N* +1 solid links connected by *N revolute joint values for N could* be as great as several hundred.

Protein Folding

- **Protein folding** is the process by which a protein structure assumes its functional shape or conformation.
- The folding occurs under the effect of nuclear forces among protein atoms as well as between protein atoms and the solvent’s atoms.
- Can be studied through dynamic simulation

Result

- Being able to accurately predict the three-dimensional structure of a protein based on the known sequences of amino acids in its chain is key to fully understanding a protein’s biological functions and thus to manipulating or controlling these functions as a part of disease treatments.

Activities with Chair Professor Fund

- Presenting new research at ACMD 2010 in Japan
- Research collaboration with IIT Madras on Rope Modeling
- Supporting several Students/Faculty to visit IIT Delhi and vice-verse
- Will be attending ICMD 2012 in Germany

Acknowledgements

- **MR. NAREN GUPTA**
- Dr. Suril V. Shah
- Mr. Rajeev Lochan
- Mr. Amit Jain
- Ms. Jyoti Bahuguna
- Mr. D. Jaitly, and others
- Dr. Sandipan Bandyopadhyah, IITM
- Dr. Madhav Krishnan, IIITH
- Mr. Vijay, IIITH

Conclusions

- DeNOC for serial-chain systems
- RoboAnalyzer
- Closed-loop systems and optimization
- MuDRA
- Rope
- New application to biological systems

THANK YOU

for

Your Attention

http://web.iitd.ac.in/~saha
saha@mech.iitd.ac.in