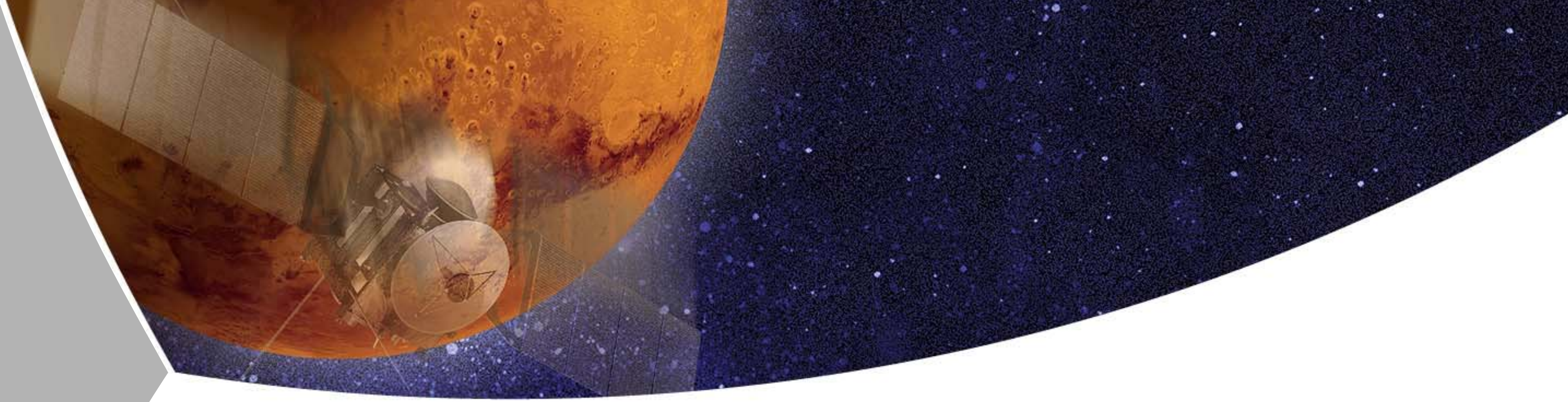


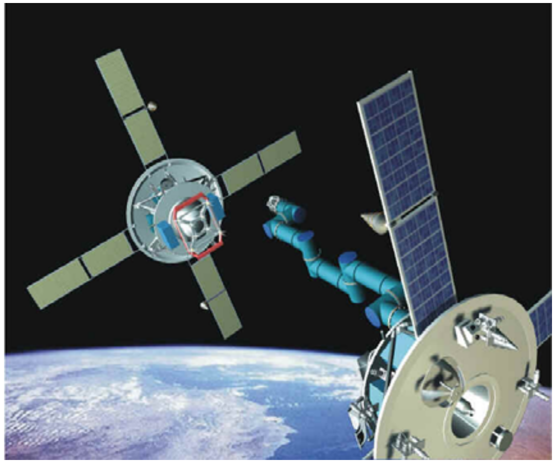
A Dynamic Library for Versatile Modeling of Free-Flying and Mobile Robotic Systems

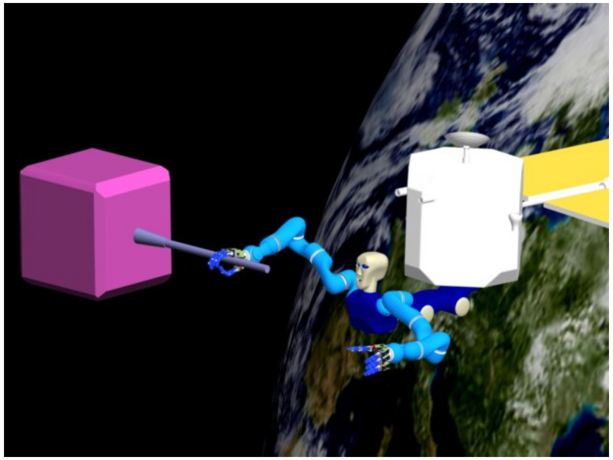
Satoko Abiko, Roberto Lampariello and Gerd Hirzinger

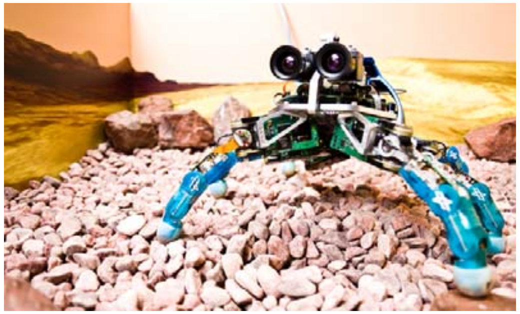
Institute of Robotics and Mechatronics
German Aerospace Center (DLR)
Satoko.Abiko@dlr.de

DLR Deutsches Zentrum für Luft- und Raumfahrt e.V.
in der Helmholtz-Gemeinschaft

Examples of Articulated Multi-Body Systems







Aims of the Research

To develop a versatile dynamic library for multi-body systems in space, in order to achieve the following purposes:

- to support motion planning for robots which require different kinematic structures for different tasks
- to be able to handle different structure changes on-line without switching between different models
- to minimize computational cost – the alternative of locking joints is computationally inefficient
- to develop an efficient dynamics simulation tool for general systems
- to develop a versatile programming environment, e.g. for the efficient development of a control algorithm

Features of the Library

- Articulated Body Algorithm (ABA) O(n) Forward and Inverse Dynamics for Free-Flying Robot
- Generalized Jacobian Matrix / Momentum Conservation Equations
- Computation of Inertia Matrix and Centrifugal Term for Control
- Branching System / Closed Loop System
- Operational Space Formulation for Control of Free-Flying Robot
- On-Line Kinematic Structure Changes
- Generation of Regressor Matrix for Parameter Identification
- Flexible Joints

- Flexible Structures (spacecraft appendages, robot links)
- Contact/Impact Dynamics
- Sloshing Model

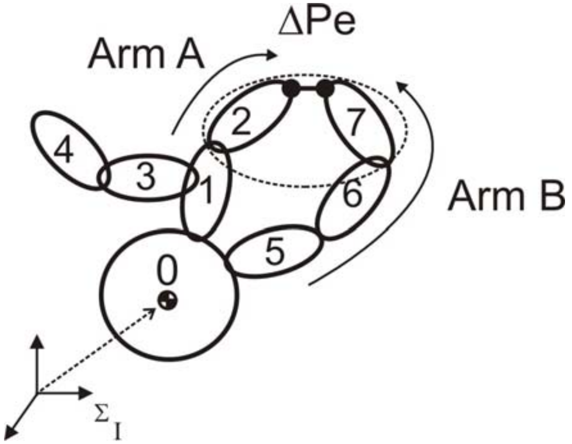
Abiko S. and Hirzinger G. (2008). "Computational Efficient Algorithms for Operational Space Formulation of Branching Arms on Space Robots," *IROS 2008*.

Lampariello R., Abiko S., and Hirzinger G. (2008). "Dynamic Modeling of Structure-Varying Kinematic Chains for Free-Flying Robots," *ICRA 2008*.

Dynamic Modeling

Open Loop System

$$\begin{bmatrix} H_b & H_{bm} \\ H_{bm}^T & H_m \end{bmatrix} \begin{bmatrix} \ddot{x}_b \\ \ddot{\phi} \end{bmatrix} + \begin{bmatrix} c_b \\ c_m \end{bmatrix} = \begin{bmatrix} \mathcal{F}_b \\ \tau \end{bmatrix} + \begin{bmatrix} J_b^T \\ J_m^T \end{bmatrix} \mathcal{F}_e$$



Closed Loop System

m independent kinematic constraints :

$$\begin{bmatrix} J_G & J_D \end{bmatrix} \begin{bmatrix} \dot{\phi}_G \\ \dot{\phi}_D \end{bmatrix} = O$$

$\phi_G \in R^{(n-m)}$: active joints

$\phi_D \in R^m$: dependent joints

$$M \begin{bmatrix} \ddot{x}_b \\ \ddot{\phi}_G \end{bmatrix} + C = \begin{bmatrix} \mathcal{F}_b \\ \tau_G \end{bmatrix} + \Pi^T \begin{bmatrix} J_b^T \\ J_m^T \end{bmatrix} \mathcal{F}_e \qquad \Pi = \begin{bmatrix} E^{6\times 6} & O^{6\times(n-m)} \\ O^{(n-m)\times 6} & E^{(n-m)\times(n-m)} \\ O^{m\times 6} & -J_D^{-1} J_G \end{bmatrix}$$

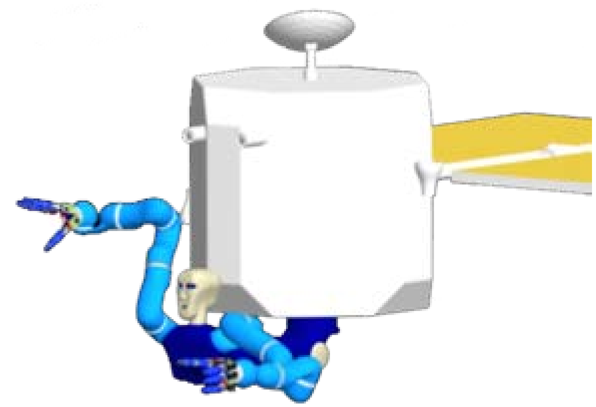
where

$$M = \Pi^T H \Pi \qquad C = \Pi^T \begin{bmatrix} c_b \\ c_m \end{bmatrix} + \Pi^T H \dot{\Pi} \begin{bmatrix} \dot{x}_b \\ \dot{\phi}_G \end{bmatrix}$$

Dynamic Modeling

Joint Space Formulation

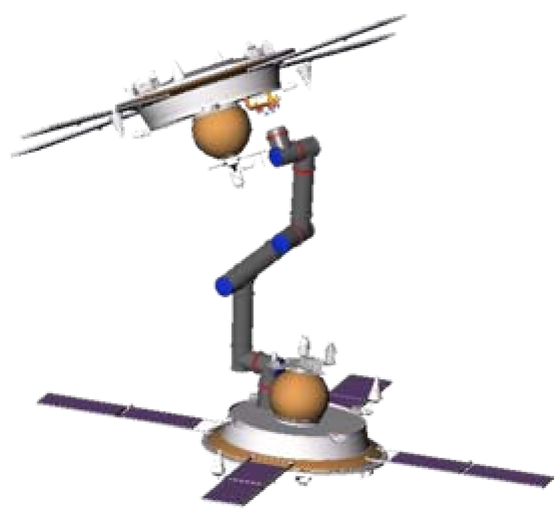
$$\begin{bmatrix} \boldsymbol{H}_b & \boldsymbol{H}_{bm} \\ \boldsymbol{H}_{bm}^T & \boldsymbol{H}_m \end{bmatrix}\begin{bmatrix}\ddot{\boldsymbol{x}}_b\\ \ddot{\boldsymbol{\phi}}\end{bmatrix} + \begin{bmatrix} \boldsymbol{c}_b \\ \boldsymbol{c}_m\end{bmatrix} = \begin{bmatrix}\mathcal{F}_b\\ \boldsymbol{\tau}\end{bmatrix} + \begin{bmatrix} \boldsymbol{J}_b^T \\ \boldsymbol{J}_m^T\end{bmatrix}\mathcal{F}_e$$



Operational Space Formulation

$$\boldsymbol{\Gamma}_e \ddot{\boldsymbol{x}}_e + \boldsymbol{\mu}_e = \mathcal{F}_e$$

$$\begin{bmatrix}\mathcal{F}_b\\ \boldsymbol{\tau}\end{bmatrix} = \boldsymbol{J}_e^T \mathcal{F}_e$$

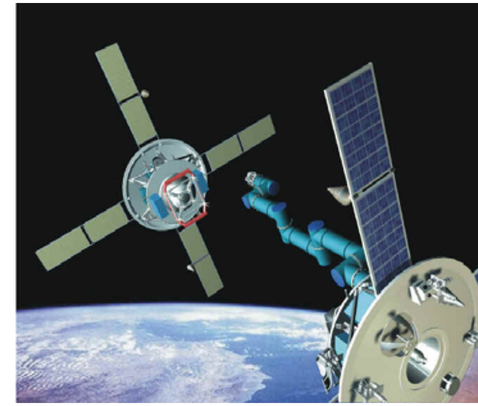


$$\boldsymbol{\Gamma}_e = (\boldsymbol{J}_e \boldsymbol{H}^{-1} \boldsymbol{J}_e^T)^{-1}$$: Recursive calculation is applied

$$\boldsymbol{\mu}_e = \boldsymbol{J}_e^{T+}\begin{bmatrix} \boldsymbol{c}_b \\ \boldsymbol{c}_m\end{bmatrix} - \boldsymbol{\Gamma}_e \frac{d}{dt}\boldsymbol{J}_e \begin{bmatrix}\dot{\boldsymbol{x}}_b\\ \dot{\boldsymbol{\phi}}\end{bmatrix}$$

Structure Changes : Extract Fixed-Base Equations

$$\begin{bmatrix} \boldsymbol{H}_b & \boldsymbol{H}_{bm} \\ \boldsymbol{H}_{bm}^T & \boldsymbol{H}_m \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{x}}_b \\ \ddot{\boldsymbol{\phi}} \end{bmatrix} + \begin{bmatrix} \boldsymbol{c}_b \\ \boldsymbol{c}_m \end{bmatrix} = \begin{bmatrix} \mathcal{F}_b \\ \boldsymbol{\tau} \end{bmatrix} + \begin{bmatrix} \boldsymbol{J}_b^T \\ \boldsymbol{J}_m^T \end{bmatrix} \mathcal{F}_e$$



$$\ddot{\boldsymbol{x}}_b = 0$$
$$\dot{\boldsymbol{x}}_b = 0$$

$$\boldsymbol{H}_m \ddot{\boldsymbol{\phi}} + \boldsymbol{c}_m = \boldsymbol{\tau} + \boldsymbol{J}_m^T \mathcal{F}_e$$

- ABA recursive O(*n*) forward and inverse dynamics

Structure Changes : Closed Loops

• the system has n generalized coordinates in total

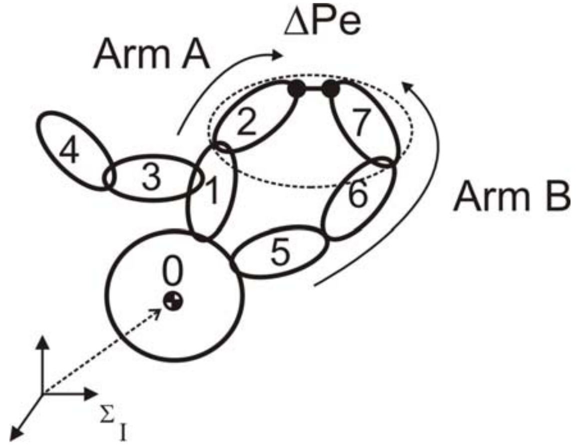
• closed loop index ***"cflag(j) = i"***
e.g. link 2 and link 7 form the closed loop with cflag(7)=2

• partitioning of joints:

- reduce number of independent states, removing the first m backwards from the constraint along the same branch

- compute new m dependent states in function of closed loop constraint

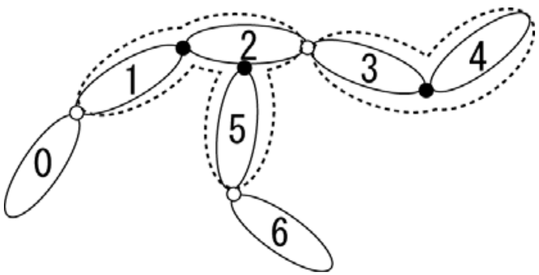
• the equations change structurally to those of the closed loop system



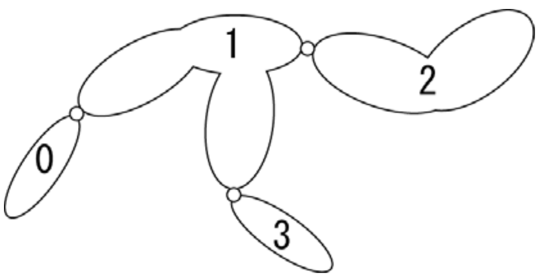
i	Parent(i)	cflag (i)
1	0	-
2	1	-
3	1	-
4	3	-
5	0	-
6	5	-
7	6	2

Structure Changes : New Number of Joints

- calculates new inertial parameters which describe the system for its $n\text{-}p$ DOFs – mass, inertia, D-H parameters for each link
- connectivity is updated via new index ***"Lock(i)"***
- equations of motion remain structurally the same
- dimension of the system is reduced or increased to represent the new number of joints of the system



i	Parent(i)	Lock(i)
1	0	0
2	1	1
3	2	0
4	3	1
5	2	1
6	6	0



i	Parent(i)	Lock(i)
1	0	0
2	1	0
3	2	0

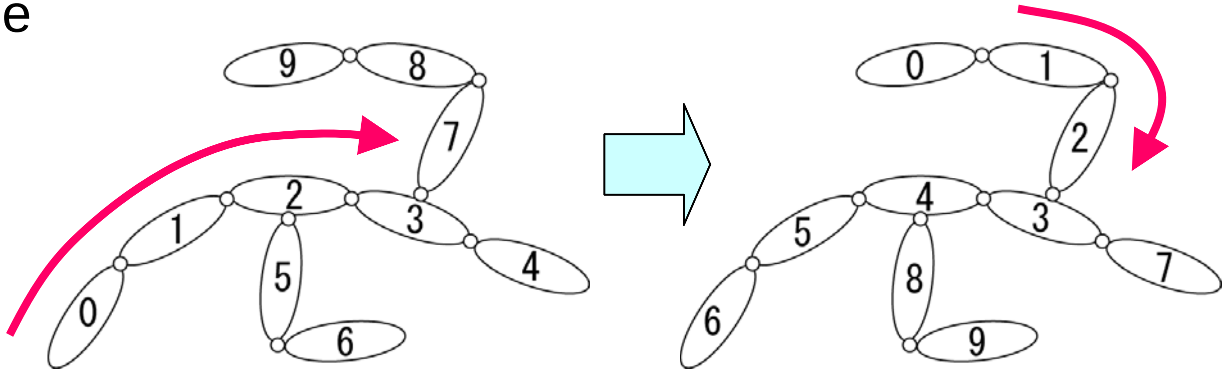
Structure Changes : Invert Chain

For a branched system the following rules apply:

- given a new *"base"*, construct the connectivity of an open chain until an end-effector is reached

- then continue the numbering of the remaining braches, such that the first child belonging to a branch is connected to the new parent which has the same numbering as that which its old parent receives

- other links in the branch follow standard connectivity rule



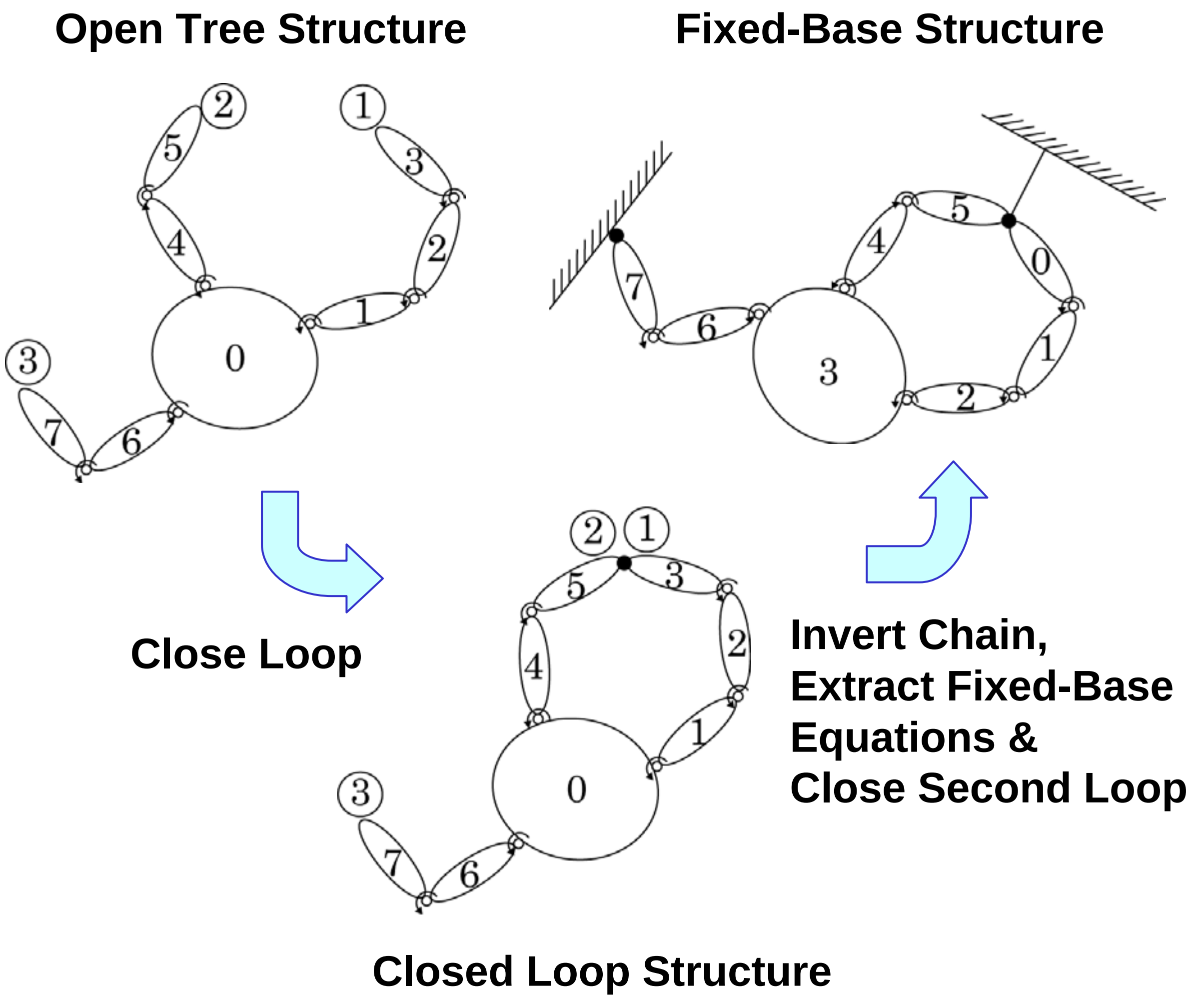
i	Parent(i)	EE(i)
0	-	-
1	0	-
2	1	-
3	2	-
4	3	1
5	2	-
6	5	2
7	3	-
8	7	-
9	8	3

i new
6
5
4
5
7
8
9
2
1
0

i	Parent(i)	EE(i)
0	-	-
1	0	-
2	1	-
3	2	-
4	3	-
5	4	-
6	5	1
7	3	2
8	4	-
9	8	3

Example :
Closed Loop Grappling to Fixed Inertial Point

- first loop is closed with ***"cflag(5)=3"***
- system is inverted with ***Invert chain*** function
- fixed-base equations are obtained with ***Extract fixed-base equations*** function
- second loop is closed with ***"cflag(7)=0"*** and appropriate choice of ΔP_e



Conclusions

- A versatile dynamic library for general multi-body systems, especially for space robots is developed
- The dynamics of structure-varying kinematic chains with complete redefinition of system connectivity is computed
- Locked joints are omitted from the computation
- Structure variations are handled on-line, such that there is no need to prepare every possible kinematic chain in advance
- Several examples are given which include typical motion planning tasks and structure changes of free-flying robots. These include open branched chains, closed loops, in the free and in the grappled states
- The library provides computationally efficient methods to easily handle simulation, motion planning, identification and control tasks