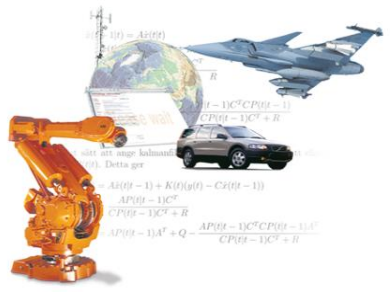
Robot modeling – the dynamics



Lecture 3
Mikael Norrlöf

Background

Kinematics – the geometric behavior of the robot.

Dynamics – full consideration of the forces / torques necessary to produce the motion.

Up till now

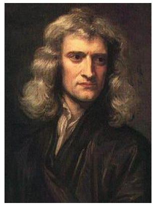
- Lecture 1
  - Introduction
  - Rigid body motion
  - Representation of rotation
  - Homogenous transformation
- Lecture 2
  - Kinematics
    - Position
    - Jacobians
    - DH parameterization

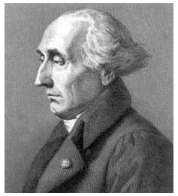
Why are the dynamic models so important?

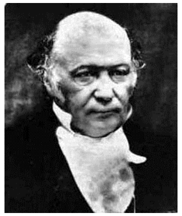
- Important in the manipulator design
  - Virtual prototyping
  - Life time estimation
  - Design optimization
- Fundamental for control design
  - Identification
  - Model based control
- “Must have” in optimal trajectory planning
- …

Systematic ways to derive the dynamic equations

- Analytical mechanics
  - Lagrange's equation
  - Newton – Euler iterative technique
  - Kane's method
  - …
- Graphic / Component modeling
  - Modelica
  - SimMechanics
  - …
- FEM – modeling
- …


Isaac Newton
1643 - 1727

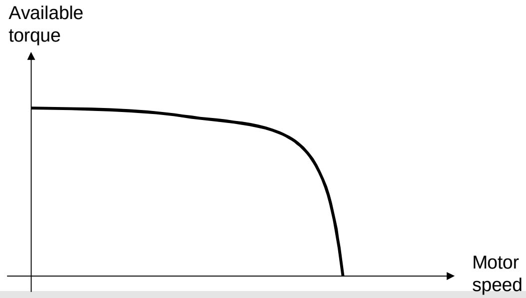

Joseph Louis Lagrange-
Giuseppe Luigi Lagrange
1736 - 1813


Sir William Rowan
Hamilton
1805 - 1865


Leonhard Euler
1707 - 1783

Limitations

- Consider open chain robot structures
  - This constraint can be relaxed …
- Actuator dynamics are neglected (e.g., motors are assumed to be ideal, torque reference in – torque out)
  - Can also be relaxed with an additional modeling effort



Basic assumptions

Consider a system of *n* particles. Newton's second law,

$$F_i = m_i \ddot{r}_i, \quad r_i \in R^3, i = 1,\dots,k$$

The particles are connected. Introduce constraints

$$g_j(r_1,\dots,r_k) = 0 \quad j = 1,\dots,l$$

Holonomic constraint: Algebraic relation between positions.

Non holonomic constraints …

Constraint forces

- The constraints form a smooth surface in R^{3k}
- Constraint forces act to keep the system velocity tangent to this surface – hence they are normal to the surface
  - The constraint forces do not produce any work!
- The system equations can be written as

$$F = \begin{pmatrix} m_1 I & & 0 \\ & \ddots & \\ 0 & & m_k I \end{pmatrix} \begin{pmatrix} \ddot{r}_i \\ \vdots \\ \ddot{r}_k \end{pmatrix} + \sum_{j=1}^{l} \Gamma_j \lambda_j, \qquad g_j(r_1,\dots,r_k) = 0 \quad j = 1,\dots,l$$

where $\Gamma_1, \dots, \Gamma_k \in R^{3k}$ are a basis for the constraint forces and λ_j are scale factors.

Γ_j can be chosen as gradient of the constraints g_j.

Better system representation

- For a system of k particles with l constraints, find $n = 3k - l$ variables $q_1, \ldots, q_n$ and functions $f_1, \ldots, f_k$

$$r_i = f_i(q_1,\ldots,q_n) \qquad g_j(r_1,\ldots,r_k) = 0$$
$$i = 1,\ldots,k \qquad j = 1,\ldots,l$$

  q_i are called *generalized coordinates.*

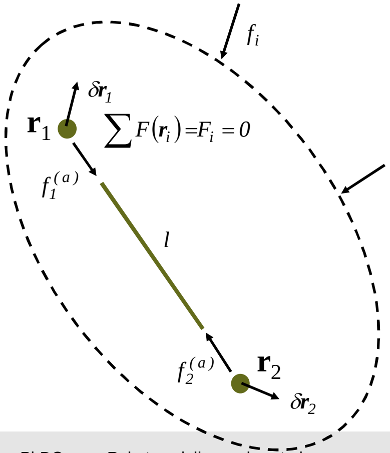
- *Generalized forces* are forces acting along the generalized coordinates.

  Example: Robot manipulator with rotational joints. The generalized forces are **torques** acting around the joints.

- The dynamic equations can be expressed in terms of the new variables.

Virtual work

Principal of virtual work: "The work done by external forces corresponding to any set of virtual displacements is zero."
(Spong etal, p247)



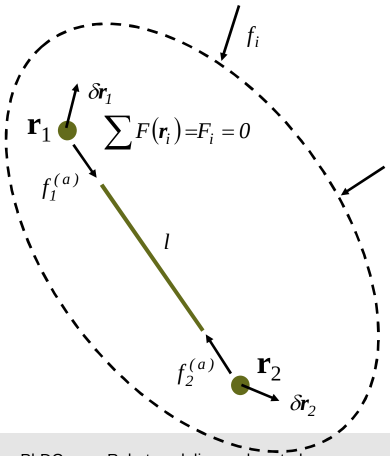
$$\sum_{i=1}^{k} F_i^T \delta r_i = 0$$

$$\sum_{i=1}^{k} \left(f_i^{(a)}\right)^T \delta r_i = 0$$

$$\sum_{i=1}^{k} f_i^T \delta r_i = 0$$

PhDCourse Robot modeling and control
Lecture 3
AUTOMATIC CONTROL
REGLERTEKNIK
LINKÖPINGS UNIVERSITET

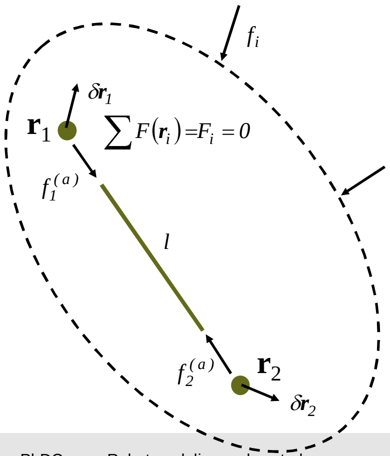
Dynamic case (D'Alembert's principle)



"if one introduces a fictitious additional force $-\dot{p}_i$ on particle i for each i, where p_i is the momentum of the particle, then each particle will be in equilibrium"

PhDCourse Robot modeling and control
Lecture 3
AUTOMATIC CONTROL
REGLERTEKNIK
LINKÖPINGS UNIVERSITET

Dynamic case (D'Alembert's principle)



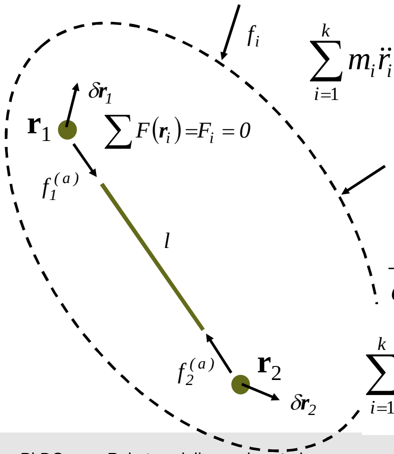
$$\sum_{i=1}^{k} f_i^T \delta r_i - \sum_{i=1}^{k} \dot{p}_i^T \delta r_i = 0$$

$$\sum_{i=1}^{k} f_i^T \delta r_i = \sum_{i=1}^{k} \sum_{j=1}^{n} f_i^T \frac{\partial \mathbf{r}_i}{\partial q_j} \delta q_j = \sum_{j=1}^{n} \psi_j \delta q_j$$

$$\psi_j = \sum_{i=1}^{k} f_i^T \frac{\partial \mathbf{r}_i}{\partial q_j}$$

PhDCourse Robot modeling and control
Lecture 3
AUTOMATIC CONTROL
REGLERTEKNIK
LINKÖPINGS UNIVERSITET

Dynamic case (D'Alembert's principle)



$$\sum_{i=1}^{k} \dot{p}_i^T \delta r_i = \sum_{i=1}^{k}\sum_{j=1}^{n} m_i \ddot{r}_i^T \frac{\partial r_i}{\partial q_j}\delta q_j$$

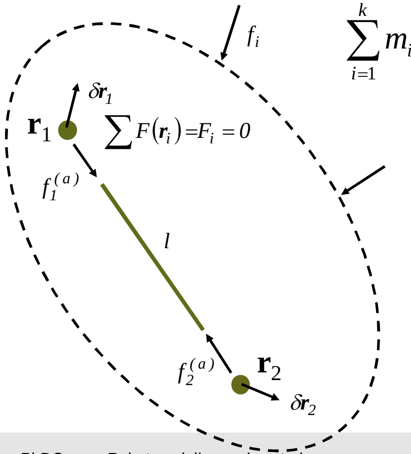
$$\sum_{i=1}^{k} m_i \ddot{r}_i^T \frac{\partial r_i}{\partial q_j} = \sum_{i=1}^{k}\left\{\frac{d}{dt}\left[m_i \dot{r}_i^T \frac{\partial r_i}{\partial q_j}\right] - m_i \dot{r}_i^T \frac{d}{dt}\left[\frac{\partial r_i}{\partial q_j}\right]\right\}$$

$$v_i = \dot{r}_i = \sum_{j=1}^{n} \frac{\partial r_i}{\partial q_j}\dot{q}_j \Rightarrow \frac{\partial v_i}{\partial \dot{q}_j} = \frac{\partial r_i}{\partial q_j}$$

$$\frac{d}{dt}\left[\frac{\partial r_i}{\partial q_j}\right] = \sum_{l=1}^{n} \frac{\partial^2 r_i}{\partial q_j \partial q_l}\dot{q}_l = \frac{\partial}{\partial q_j}\sum_{l=1}^{n}\frac{\partial r_i}{\partial q_l}\dot{q}_l = \frac{\partial v_i}{\partial q_j}$$

$$\sum_{i=1}^{k} m_i \ddot{r}_i^T \frac{\partial r_i}{\partial q_j} = \sum_{i=1}^{k}\left\{\frac{d}{dt}\left[m_i v_i^T \frac{\partial v_i}{\partial \dot{q}_j}\right] - m_i v_i^T \frac{\partial v_i}{\partial q_j}\right\}$$

Dynamic case (D'Alembert's principle)



$$K = \sum_{i=1}^{k} \frac{1}{2} m_i \mathbf{v}_i^T \mathbf{v}_i$$

$$\sum_{i=1}^{k} m_i \ddot{r}_i^T \frac{\partial r_i}{\partial q_j} = \sum_{i=1}^{k}\left\{\frac{d}{dt}\left[m_i v_i^T \frac{\partial v_i}{\partial \dot{q}_j}\right] - m_i v_i^T \frac{\partial v_i}{\partial q_j}\right\}$$

$$= \frac{d}{dt}\frac{\partial K}{\partial \dot{q}_j} - \frac{\partial K}{\partial q_j}$$

$$\sum_{i=1}^{k} \dot{p}_i^T \delta r_i = \sum_{j=1}^{n}\left\{\frac{d}{dt}\frac{\partial K}{\partial \dot{q}_j} - \frac{\partial K}{\partial q_j}\right\}\delta q_j$$

$$\sum_{j=1}^{n}\left\{\frac{d}{dt}\frac{\partial K}{\partial \dot{q}_j} - \frac{\partial K}{\partial q_j} - \psi_j\right\}\delta q_j = 0$$

Lagrangian and Lagrange's equation

- The Lagrangian is defined as

$$L(q,\dot{q}) = K(q,\dot{q}) - P(q)$$

Kinetic energy ↗ ↖ Potential energy

- Lagrange's equation

$$\frac{d}{dt}\frac{\partial L(q,\dot{q})}{\partial \dot{q}_i} - \frac{\partial L(q,\dot{q})}{\partial q_i} = \tau_i, \quad i = 1,\ldots,n$$

- Newton's law in generalized coordinates

$$\frac{d}{dt}\frac{\partial L(q,\dot{q})}{\partial \dot{q}} = \frac{\partial L(q,\dot{q})}{\partial q} + \tau$$

$\frac{d}{dt}$ (momentum) = applied force

Hamilton's principle

Hamilton's principle states that the true evolution of a system described by *m generalized coordinates* between two specified states and at two specified times t_1 and t_2 is an extremum of the action functional

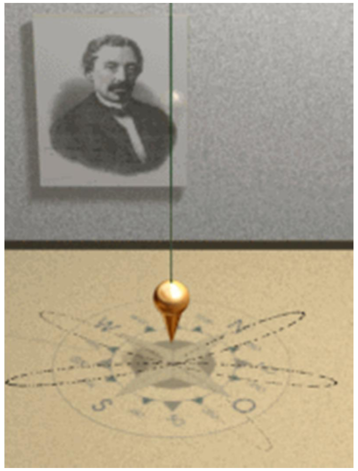
$$S(q) = \int_{t_1}^{t_2} L(q,\dot{q})dt, \quad \frac{\partial S(q)}{\partial q} = 0$$

Trajectory $q(t)$ is a stationary point of S. Assume ε is a perturbation (0 at t_1 and t_2)

$$\partial S = \int_{t_1}^{t_2} L(q+\varepsilon, \dot{q}+\dot{\varepsilon}) - L(q,\dot{q})dt = \int_{t_1}^{t_2} \varepsilon\frac{\partial L}{\partial q} + \dot{\varepsilon}\frac{\partial L}{\partial \dot{q}}dt$$

$$= \underbrace{\left[\varepsilon\frac{\partial L}{\partial \dot{q}}\right]_{t_1}^{t_2}}_{=0} + \int_{t_1}^{t_2}\varepsilon\left(\frac{\partial L}{\partial q} - \frac{d}{dt}\frac{\partial L}{\partial \dot{q}}\right)dt$$

Example



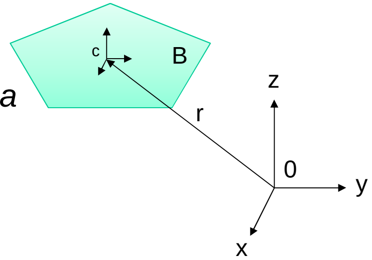
Kinetic and potential energy

- Kinetic energy for body B

$$K = \frac{1}{2} m \dot{r}^T \dot{r} + \frac{1}{2} \omega^T I \omega$$

m is the mass and I is the *inertia tensor*.

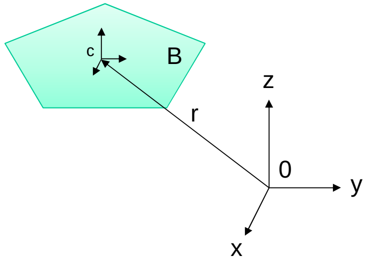
- Inertia tensor
  - 3x3 matrix
  - Symmetric
  - Positive definite
  - Constant in a body fixed coordinate system



Inertia tensor

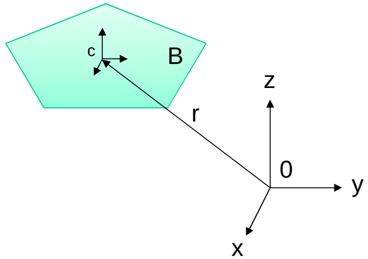
- Rotational part of *K*
  - ω expressed in inertial frame
  - With I_b the inertia tensor in body fixed coordinate system

$$\omega^T I \omega = \omega^T R I_b R^T \omega$$



Inertia tensor

- Computation of the inertia tensor. $\rho(x,y,z)$ is the mass density as a function of position.



$$I = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zx} & I_{zz} \end{bmatrix}$$

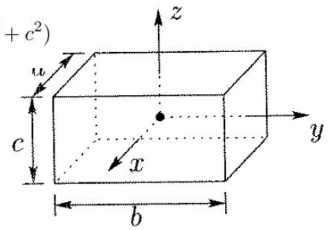
$$\begin{aligned} I_{xx} &= \int\int\int (y^2+z^2)\rho(x,y,z)dx\ dy\ dz & I_{xy} = I_{yx} &= -\int\int\int xy\rho(x,y,z)dx\ dy\ dz \\ I_{yy} &= \int\int\int (x^2+z^2)\rho(x,y,z)dx\ dy\ dz & I_{xz} = I_{zx} &= -\int\int\int xz\rho(x,y,z)dx\ dy\ dz \\ I_{zz} &= \int\int\int (x^2+y^2)\rho(x,y,z)dx\ dy\ dz & I_{yz} = I_{zy} &= -\int\int\int yz\rho(x,y,z)dx\ dy\ dz \end{aligned}$$

Example: Uniform rectangular solid

Body frame attached to center of gravity.

$$I_{xx} = \int_{-c/2}^{c/2}\int_{-b/2}^{b/2}\int_{-a/2}^{a/2}(y^2+z^2)\rho(x,y,z)dx\ dy\ dz = \rho\frac{abc}{12}(b^2+c^2)$$

$$I_{yy} = \rho\frac{abc}{12}(a^2+c^2) \quad ; \quad I_{zz} = \rho\frac{abc}{12}(a^2+b^2)$$



Notice that

$\rho abc = m$

Inertia of two bodies expressed in the same coordinate frame can be added (subtracted)

Kinetic energy for an n-link manipulator

From lecture 2 we know

$$v_i = J_{v_i}(q)\dot{q}, \qquad \omega_i = J_{\omega_i}(q)\dot{q}$$

where v_i and ω_i can be for any point on the manipulator (depends on the Jacobian).

The kinetic energy can now be expressed as

$$\begin{aligned} K &= \frac{1}{2}\dot{q}^T\sum_{i=1}^{n}\left[m_i J_{v_i}(q)^T J_{v_i}(q) + J_{\omega_i}(q)^T R_i(q) I_i R_i(q)^T J_{\omega_i}(q)\right]\dot{q} \\ &= \frac{1}{2}\dot{q}^T D(q)\dot{q} \end{aligned}$$

where $D(q)$ is the *inertia matrix*.

Properties: Symmetric and positive definite.

Potential energy for an n-link manipulator

In rigid dynamics, gravity is the only source of potential energy.

$$P = \sum_{i=1}^{n} P_i = \sum_{i=1}^{n} m_i g^T r_{ci}$$

If the robot contains elastic components the energy stored in the elasticities has to be included in the potential energy.

Dynamic equations

Lagrangian $L = K - P = \frac{1}{2}\sum_{i,j} d_{ij}(q)\dot{q}_i\dot{q}_j - P(q)$

Recall: Lagrange's equation

$$\frac{d}{dt}\frac{\partial L(q,\dot{q})}{\partial \dot{q}_k} - \frac{\partial L(q,\dot{q})}{\partial q_k} = \tau_k, \quad k = 1,\ldots,n$$

In terms of L above this gives

$$\frac{\partial L}{\partial \dot{q}_k} = \sum_j d_{kj}\dot{q}_j \Rightarrow \frac{d}{dt}\frac{\partial L}{\partial \dot{q}_k} = \sum_j d_{kj}\ddot{q}_j + \sum_{i,j}\frac{\partial d_{kj}}{\partial q_i}\dot{q}_i\dot{q}_j$$

$$\frac{\partial L}{\partial q_k} = \frac{1}{2}\sum_{i,j}\frac{\partial d_{ij}}{\partial q_k}\dot{q}_i\dot{q}_j - \frac{\partial P}{\partial q_k}$$

Dynamic equations, cont'd

$$\sum_j d_{kj}\ddot{q}_j + \sum_{i,j}\left\{\frac{\partial d_{kj}}{\partial q_i} - \frac{1}{2}\frac{\partial d_{ij}}{\partial q_k}\right\}\dot{q}_i\dot{q}_j + \frac{\partial P}{\partial q_k} = \tau_k$$

$$\Rightarrow \sum_j d_{kj}\ddot{q}_j + \sum_{i,j}\underbrace{\frac{1}{2}\left\{\frac{\partial d_{kj}}{\partial q_i} + \frac{\partial d_{ki}}{\partial q_j} - \frac{\partial d_{ij}}{\partial q_k}\right\}}_{c_{ijk}}\dot{q}_i\dot{q}_j + \frac{\partial P}{\partial q_k} = \tau_k$$

$$c_{ijk} = c_{jik}$$

Dynamic equations

$$\sum_j d_{kj}\ddot{q}_j + \sum_{i,j} c_{ijk}\dot{q}_i\dot{q}_j + g_k = \tau_k$$

Christoffel symbols (→ c_{ijk}) gravity (→ g_k)

Dynamic equations, cont'd

In matrix form

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \tau$$

State space representation $\dot{x} = f(x,u), \ x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} q \\ \dot{q} \end{pmatrix}$

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} x_2 \\ D^{-1}(x_1)\big(-C(x_1,x_2)x_2 - g(x_1) + u\big) \end{pmatrix}$$

Extensions. Friction and joint flexibilities

$$D(q_a)\ddot{q}_a + C(q_a,\dot{q}_a)\dot{q}_a + g(q_a) + f\dot{q}_a = k(q_a - q_m) + d(q_a - q_m)$$

$$M\ddot{q}_m + f_m\dot{q}_m = -k(q_a - q_m) - d(q_a - q_m) + \tau$$

Some properties

The following matrix is skew symmetric

$$N(q,\dot{q}) = \dot{D}(q) - 2C(q,\dot{q})$$

The system is passive.

The inertia matrix is bounded (constant lower/upper bound when only revolute joints)

$$\lambda_m I_{nxn} \le D(q) \le \lambda_M I_{nxn} < \infty$$

Newton-Euler

- Different approach to find the dynamic equations
- A more local approach
  - Each link is modeled separately
  - Links interconnected, leads to a forward – backward iteration scheme

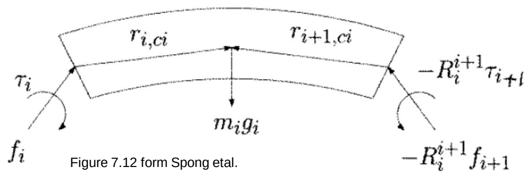


Figure 7.12 form Spong etal.

$$\tau_i - R_{i+1}^i\tau_{i+1} + f_i \times r_{i,ci} - (R_{i+1}^i f_{i+1}) \times r_{i+1,ci} = \alpha_i + \omega_i \times (I_i\omega_i)$$

Balance equations: $f_i - R_{i+1}^i f_{i+1} + m_i g_i = m_i a_{c,i}$

Newton-Euler, cont'd

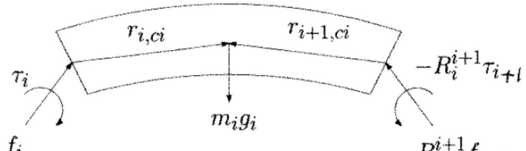


Figure 7.12 form Spong etal.

Solve from $i = 0$ to n

$$\omega_i = (R_i^{i-1})^T \omega_{i-1} + b_i \dot{q}_i$$

$$b_i = (R_i^0)^T z_{i-1}$$

$$\alpha_i = (R_{i-1}^i)^T \alpha_{i-1} + b_i \ddot{q}_i + \omega_i \times b_i \dot{q}_i$$

$$a_{e,i} = (R_i^{i-1})^T a_{e,i-1} + \dot{\omega}_i \times r_{i,i+1} + \omega_i \times (\omega_i \times r_{i,i+1})$$

$$a_{c,i} = (R_i^{i-1})^T a_{e,i-1} + \dot{\omega}_i \times r_{i,ci} + \omega_i \times (\omega_i \times r_{i,ci})$$

Newton-Euler, cont'd

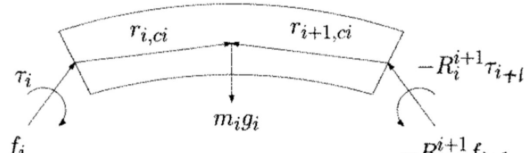


Figure 7.12 form Spong etal.

Balance equations:

$$f_i - R_{i+1}^i f_{i+1} + m_i g_i = m_i a_{c,i}$$

$$\tau_i - R_{i+1}^i \tau_{i+1} + f_i \times r_{i,ci} - (R_{i+1}^i f_{i+1}) \times r_{i+1,ci}$$

$$= \alpha_i + \omega_i \times (I_i \omega_i)$$

- Find f_i and τ_i by solving the equations from $f_{n+1} = 0$ and $\tau_{n+1} = 0$.

$$f_i = R_{i+1}^i f_{i+1} + m_i a_{c,i} - m_i g_i$$

$$\tau_i = R_{i+1}^i \tau_{i+1} - f_i \times r_{i,ci} + (R_{i+1}^i f_{i+1}) \times r_{i+1,ci} + \alpha_i + \omega_i \times (I_i \omega_i)$$

Newton-Euler vs Langrangian

- Solves the same problem.
- Lagrangian technique gives the dynamic equations "directly".
- Newton-Euler gives all torques / forces, not just the generalized torques. Will be very important later …
- …

Home assignment, part I

- From a blue-print of a robot (IRB1600-8/1.45) find a
  - Kinematic model
  - Assume the robot to consist of hollow uniform rectangular beams made out of metal (steel, aluminum, iron, …)
  - Compute the inertia matrix for each link
  - Derive a dynamic model using for example Lagrange's equation
- The dynamic model can be restricted to 3-DOF while the kinematics shall be derived for a full 6-DOF manipulator.
- Inverse kinematic should be implemented (can be numerical)
- Include gear-box in the model (gear ratio [-100 100 100 -60 -60 40]:1), motor inertia can be assumed to be 50 – 100 % link inertia when transformed to the arm-side, i.e., after the gear-box)
- Motor torque max, [6 10 5 0.6 0.6 0.5] Nm

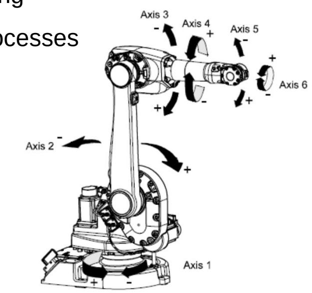
DEAD-LINE: JAN 24

The robot in the assignment

A small/mid size robot

Main applications

- Arc Welding
- Machine tending
- Material handling
- Continuous processes



Kinematic data (pdf-files on the homepage)