Robot Dynamics

Topics

- Dynamics

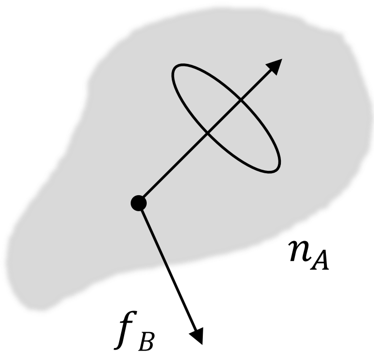
Complementary Reading: J.J. Craig, Introduction to Robotics, Chapter 6.

Recap on Kinematics

- **Forward kinematics**: mapping from joint variables to position and orientation of the end effector
- **Inverse kinematics:** finding joint variables that satisfy a given position and orientation of the end effector
- **Jacobian:** mapping from the joint velocities to the end effector linear and angular velocities

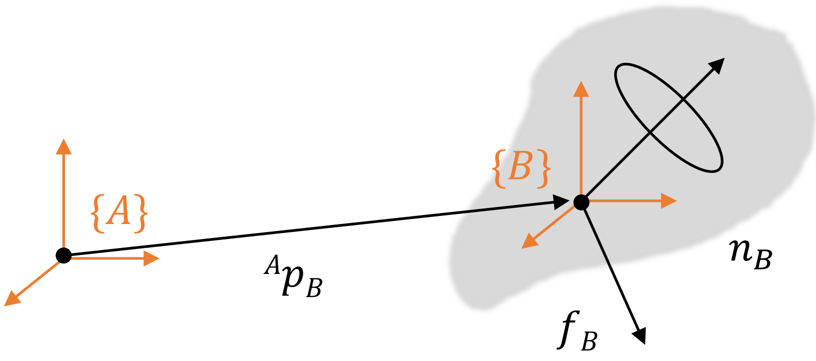
(Newtonian) Dynamics

- All robots are controlled using forces and torques at its joints
  - Often, system constraints are necessarily modeled using Newtonian physics
    - Inertia of a moving robot
    - Friction: slip vs.. grip (e.g. $F_f = \mu N$)
    - Torque of force limits on actuators



- Modeling Dynamics Requires:
  - Center of gravity, $r \in \mathbb{R}^3$
  - Moments of Inertia, $I \in S^{N \times N}$
    - Describes mass distribution’s effect on the inertia of each body
    - Requires geometry of object, density, etc.

Changing Forces and Torques between Coordinate Frames



- Torques are *free vectors.*
- Forces are *fixed vectors* and have an origin.
- Conversion of forces/torques, as seen with kinematic chains, mixes free and fixed vectors, requiring coupling equations:

$$ {}^{A}f_{B} = {}^{A}_{B}R\, {}^{B}f_{B} $$
$$ {}^{A}n_{B} = {}^{A}\hat{p}_{B}\, {}^{A}f_{B} + {}^{A}_{B}R\, {}^{B}n_{B} $$

$$ \begin{bmatrix} {}^{A}f_{B} \\ {}^{A}n_{B} \end{bmatrix} = \begin{bmatrix} {}^{A}_{B}R & 0 \\ {}^{A}\hat{p}_{B}\, {}^{A}_{B}R & {}^{A}_{B}R \end{bmatrix} \begin{bmatrix} {}^{B}f_{B} \\ {}^{B}n_{B} \end{bmatrix} $$

Euler-Lagrange Dynamics

Euler-Lagrange solve dynamics model via conservation of energy.

- The equations of motion for an $q \in \mathbb{R}^N$ system is:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} = u_i \qquad\qquad i = 1 \ldots N$$

- $L \in \mathbb{R}$, the Lagrangian, represents the difference between sum of kinetic and potential energies of your system (*from all sources*)

$$L = K - P$$

- $u_i \in \mathbb{R}$ is the sum of force/torque applied from all external loads along i^{th} axis
- $q_i \in \mathbb{R}$ is the joint position
- <u>Observations</u>
  - The left side contains the conservative terms (internal)
  - The right side contains the non-conservative terms (external)
  - q's are the joint angles, which are defined as generalized coordinates.

Kinetic Energy

$$K = \sum_{i=0}^{n} \left(\frac{1}{2} m_i v_i^{\top} v_i + \frac{1}{2} \omega_i^{\top} I_i \omega_i \right)$$

Potential Energy

$$P = \sum_{i=1}^{n} m_i {}^{0}g^{\top} {}^{0}p_{CoM,i}$$

- v is the linear velocity of the center of mass for a link
- ω_i is the angular velocity of the center of mass for a link
- m is the mass of the link
- I is the moments of area for the link
- r_c is the center of mass vector.

Inertia

Inertia is an intrinsic property of a body

- Both **translational** and **rotational** components for rigid bodies
- Mass affects linear momentum
- Rotational inertia affects angular momentum
- In the body frame, it is a constant 3x3 matrix:

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

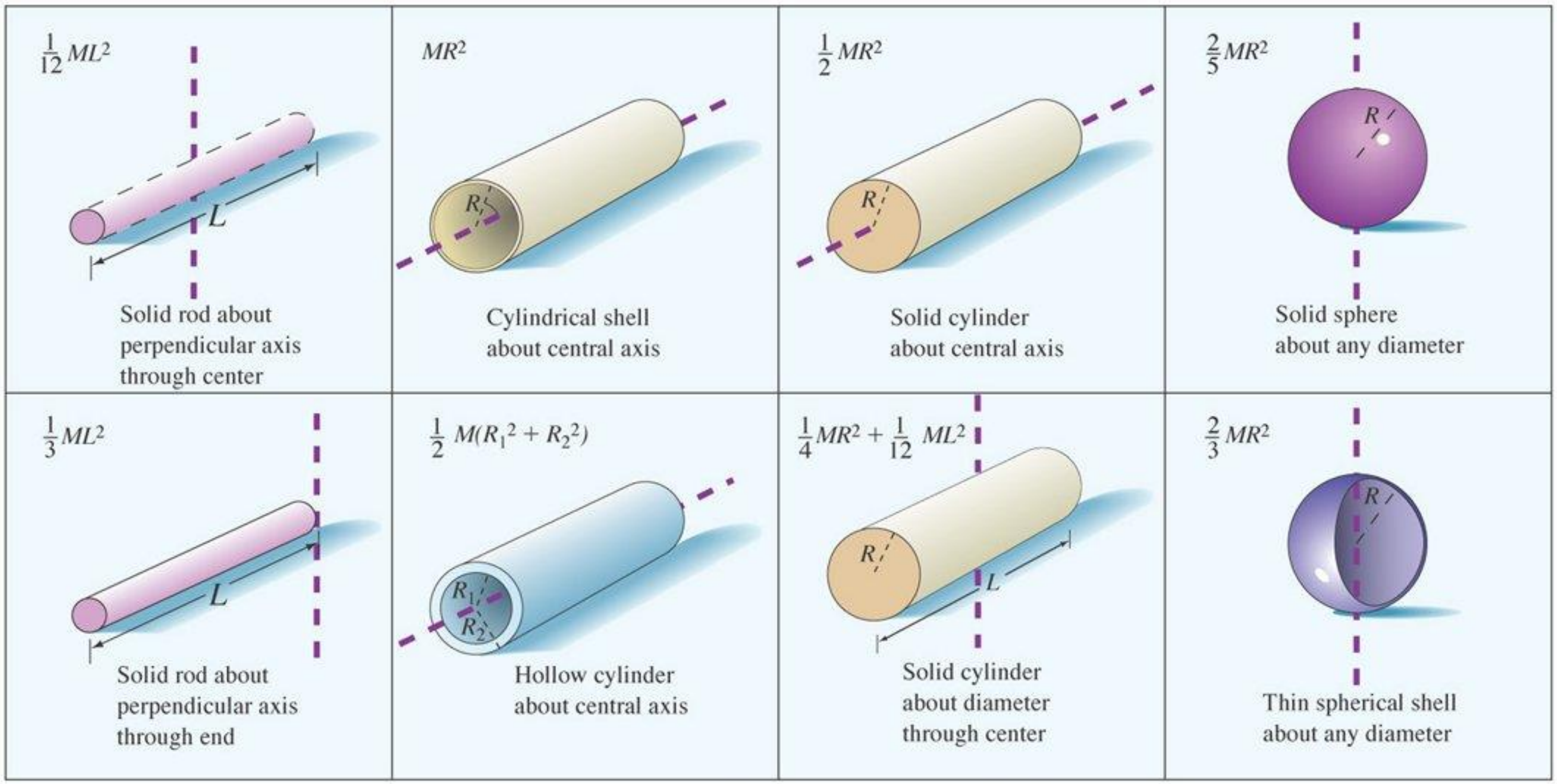
- Elements represent inertial mass distribution of a body in a coordinate frame

$$i_{xx} = \iiint_{x,y,z} (y^2 + z^2)\rho(x,y,z)dxdydz$$ (principle moments of inertia)

$$i_{xy} = \iiint_{x,y,z} xy\rho(x,y,z)dxdydz$$ (cross products of inertia)

Solving for Inertia for Multi-body Robot Systems

- Of specific interest is i_{zz} since following D-H conventions the z axis will always be the axis on which a body rotates.
- Common forms of i_{zz}:



- When CoM does not line up with axis of rotation k, use **Parallel Axis Theorem** to shift inertia matrix elements from the body CoM to the desired axis k:

$$I_k = I_{CoM} - m\hat{p}_k\hat{p}_k$$

where $p_k$is the displacement from CoM to the axis k

- Inertia in the $\{C_0\}$ can be described in a different frame's orientation $\{C_i\}$ by a similarity transform:

$$^0I_i = {}^0_iR\,{}^iI_i\,{}^0_iR^\top$$

Why is this useful? Prove that the reference frame for calculating kinetic energy for each link is not important, as long as the reference frame remains the same for term of K, P in $L = K - P$.

Deriving Dynamics Equations of Motion via Euler Lagrange

Begin with Euler Lagrange Equation:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_k}\right) - \frac{\partial L}{\partial q_k} = u_k \qquad\qquad k = 1 \dots N$$

1. Write a $L = K - P$ of the entire system**:**

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

$$= \frac{1}{2}\dot{q}^\top D(q)\dot{q} - P(q)$$

$$= \frac{1}{2}\sum_{j}^{n}\sum_{i}^{n} d_{ij}(q)\dot{q}_i\dot{q}_j - P(q)$$

2a. Substituting L into the first term of the Euler-Lagrange equation:

$$\frac{\partial L}{\partial \dot{q}_k} = \frac{\partial}{\partial \dot{q}_k}\left(\frac{1}{2}\sum_j^n \sum_i^n d_{ij}(q)\dot{q}_i\dot{q}_j - P(q)\right) = \frac{1}{2}\sum_j^n d_{kj}(q)\dot{q}_j + \frac{1}{2}\sum_i^n d_{ik}(q)\dot{q}_i$$

$= \sum_j^n d_{kj}(q)\dot{q}_j$ (due to symmetric nature of D)

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_k}\right) = \frac{d}{dt}\left(\sum_i^n d_{kj}(q)\dot{q}_j\right) = \sum_j^n \sum_i^n \left(\frac{\partial d_{kj}(q)}{\partial q_i}\dot{q}_i\right)\dot{q}_j + \sum_j^n d_{kj}(q)\ddot{q}_j$$

2b. Substituting L into the first term of the Euler-Lagrange equation:

$$\frac{\partial L}{\partial q_k} = \frac{\partial}{\partial q_k}\left(\frac{1}{2}\sum_j^n \sum_i^n d_{ij}(q)\dot{q}_i\dot{q}_j - P(q)\right) = \frac{1}{2}\sum_j^n \sum_i^n \frac{\partial d_{ij}(q)}{\partial q_k}\dot{q}_i\dot{q}_j - \frac{\partial P(q)}{\partial q_k}$$

After substitution

$$\left(\sum_{j}^{n}\sum_{i}^{n}\left(\frac{\partial d_{kj}(q)}{\partial q_i}\dot{q}_i\right)\dot{q}_j+\sum_{j}^{n}d_{kj}(q)\ddot{q}_j\right)-\left(\frac{1}{2}\sum_{j}^{n}\sum_{i}^{n}\frac{\partial d_{ij}(q)}{\partial q_k}\dot{q}_i\dot{q}_j-\frac{\partial P(q)}{\partial q_k}\right)=u_k$$

$$\sum_{j=1}^{n}d_{kj}(q)\ddot{q}_j+\sum_{j=1}^{n}\sum_{i=1}^{n}\left(\frac{\partial d_{kj}(q)}{\partial q_i}-\frac{1}{2}\frac{\partial d_{ij}(q)}{\partial q_k}\right)\dot{q}_i\dot{q}_j+g_k(q)=u_k$$

3. Let $\frac{\partial P(q)}{\partial q_k}=g_k(q)$ and $c_{kij}(q)=\frac{\partial d_{kj}(q)}{\partial q_i}-\frac{1}{2}\frac{\partial d_{ij}(q)}{\partial q_k}$, then

$$\boxed{\begin{gathered}\sum_{j=1}^{n}d_{kj}(q)\ddot{q}_j+\sum_{j=1}^{n}\sum_{i=1}^{n}c_{kij}(q)\dot{q}_i\dot{q}_j+g_k(q)=u_k\\ D(q)\ddot{q}+C(q,\dot{q})+G(q)=U\end{gathered}}$$

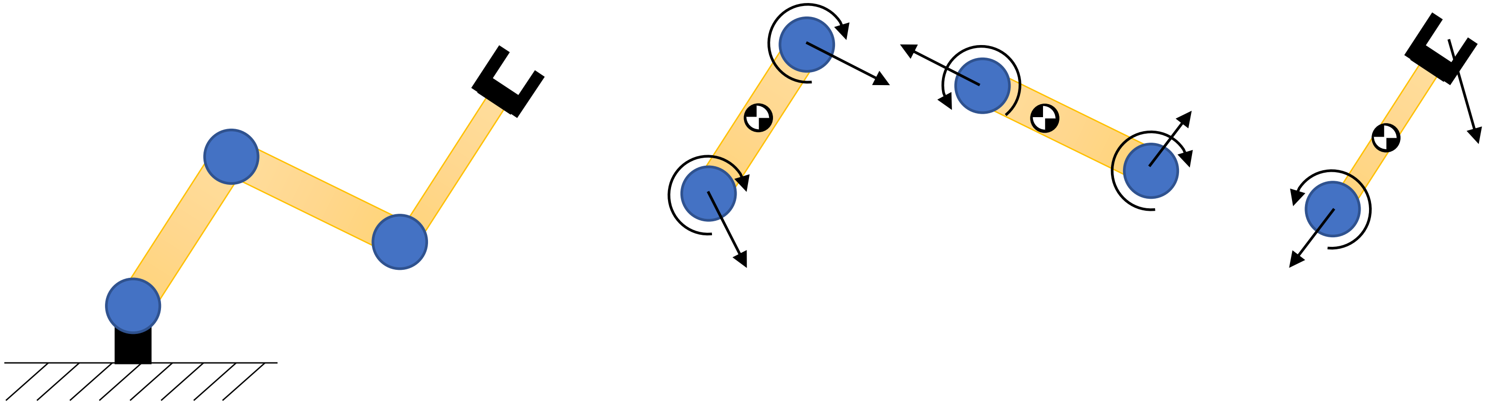
(Robot Manipulator Dynamics Equations)

- $D(q)\in\mathbb{R}^{N\times N}$: inertia mass (sometimes $M(q)$ is used)
- $G(q)\in\mathbb{R}^{N}$: gravity term
- $C(q,\dot{q})\in\mathbb{R}^{N}$: Coriolis and Centrifugal terms
- $U\in\mathbb{R}^{N}$: All dissipative and external forces/torques

Newton – Euler formulation of Dynamics

Newton-Euler equations of motion based on rigid bodies acting on one another

- Uses coordinate frames and vector analysis
- Considers
  - internal constraint forces between linkages
  - mass and center of gravity of the arm, and its cascading effects of proximal joints if non-static



Complicated to analyze, but possible.

Observations:

- Euler-Lagrange *much* easier to solve but does not give you internal forces
  - Sometimes necessary to minimize in control
  - Sometimes needed for calculating material strain
  - etc.
- Newton Euler can also be rearranged into the form

$$D(q)\ddot{q} + C(q,\dot{q}) + G(q) = U$$

Solving Dynamical Equations

- Given U find $q, \dot{q}, \ddot{q}$ and thus $x, \dot{x}, \ddot{x}$?
- Robots are dynamically modeled as a set of nonlinear PDEs
- Solving either for U or Q requires solving the PDE equations
  - No analytical solution! Need numerical methods.