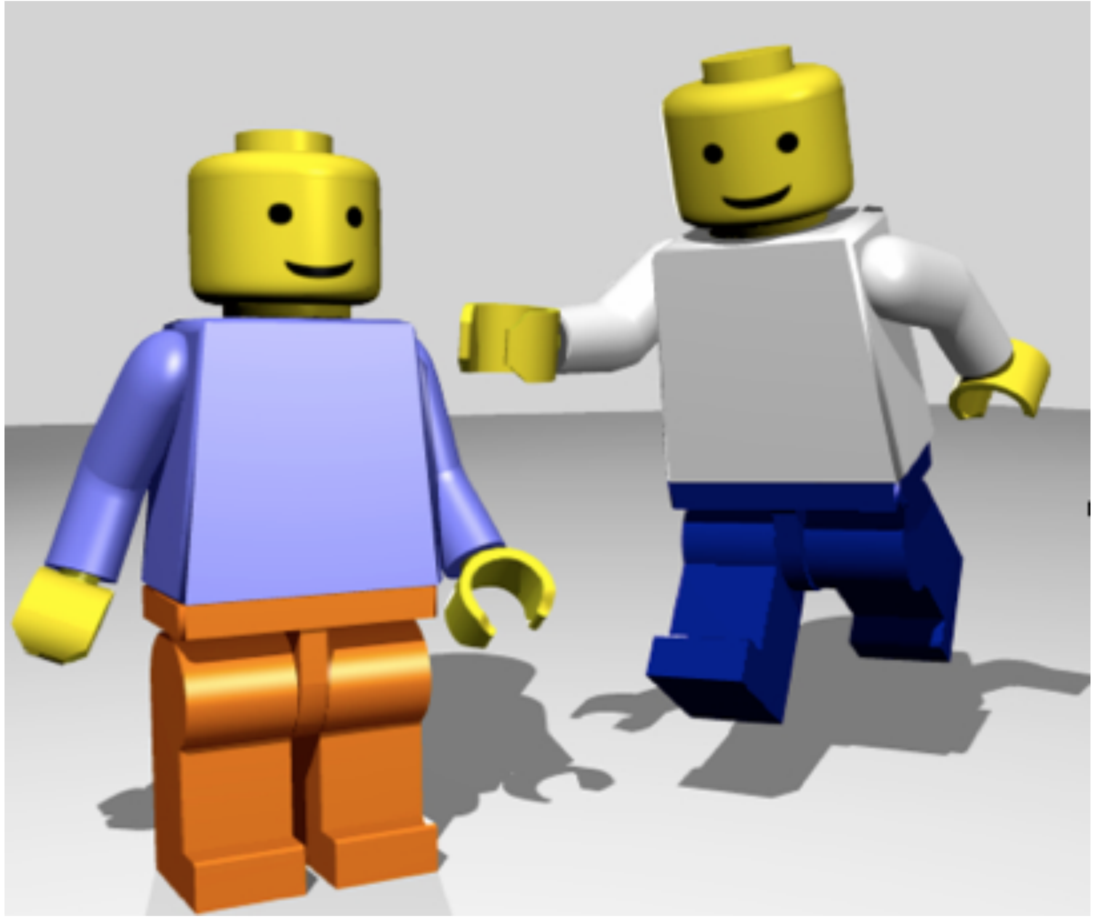
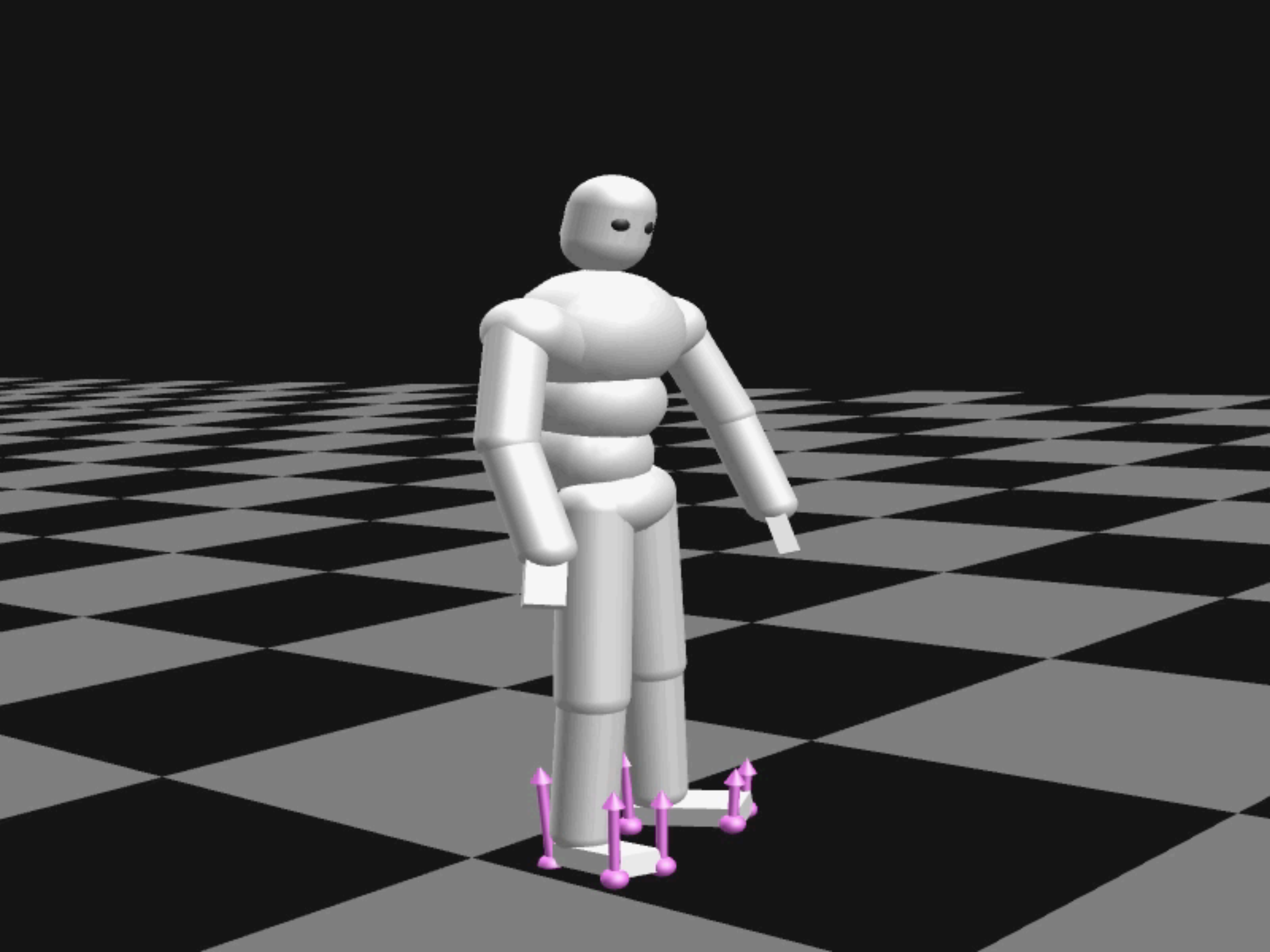
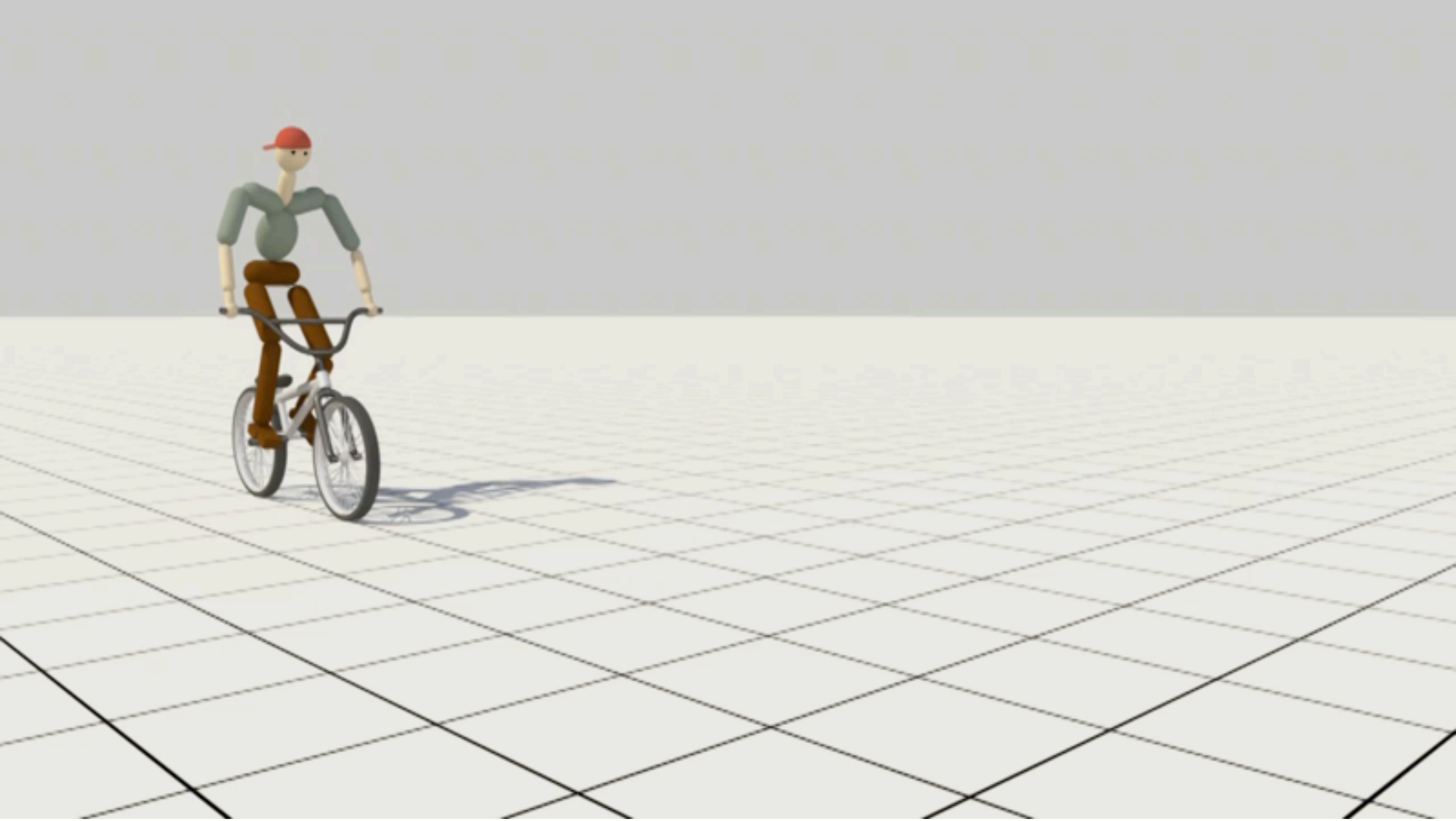
Multibody dynamics



Applications

- Human and animal motion
- Robotics control
- Hair
- Plants
- Molecular motion

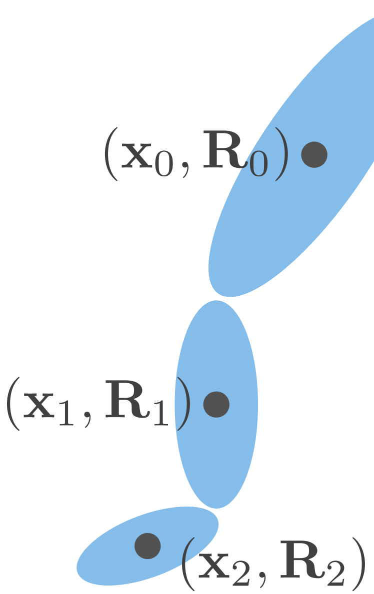




- Generalized coordinates
- Virtual work and generalized forces
- Lagrangian dynamics for mass points
- Lagrangian dynamics for a rigid body
- Lagrangian dynamics for a multibody system
- Forward and inverse dynamics

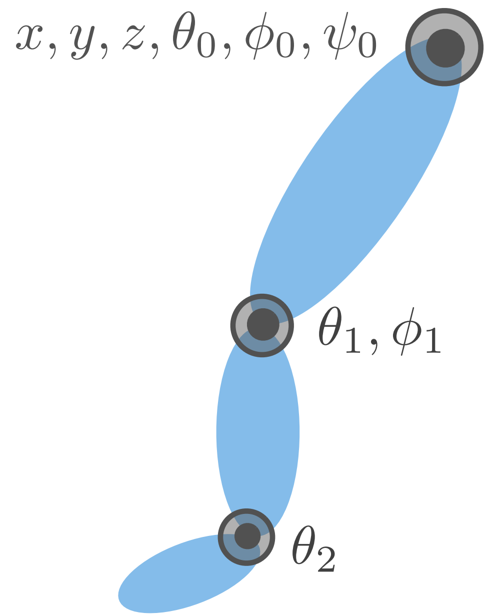
Representations

Maximal coordinates



state variables: 18

Generalized coordinates



state variables: 9

Assuming there are m links and n DOFs in the articulated body, how many constraints do we need to keep links connected correctly in maximal coordinates?

Maximal coordinates

- Direct extension of well understood rigid body dynamics; easy to understand and implement
- Operate in Cartesian space; hard to
    - evaluate joint angles and velocities
    - enforce joint limits
    - apply internal joint torques
- Inaccuracy in numeric integration can cause body parts to drift apart

Generalized coordinates

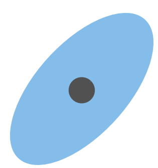
- Joint space is more intuitive when dealing with complex multi-body structures
- Fewer DOFs and fewer constraints
- Hard to derive the equation of motion

Generalized coordinates

- Generalized coordinates are independent and completely determine the location and orientation of each body

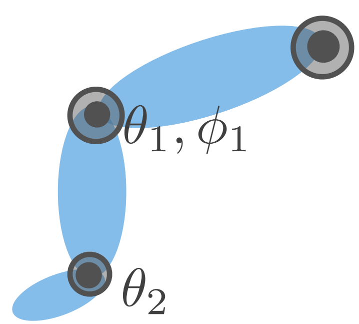
one particle: x, y, z

one rigid body:



$x, y, z, \theta, \phi, \psi$

articulated bodies:



$x, y, z, \theta_0, \phi_0, \psi_0$

θ_1, ϕ_1

θ_2

Peaucellier mechanism

- The purpose of this mechanism is to generate a straight-line motion
- This mechanism has seven bodies and yet the number of degrees of freedom is one

- Generalized coordinates
- Virtual work and generalized forces
- Lagrangian dynamics for mass points
- Lagrangian dynamics for a rigid body
- Lagrangian dynamics for a multibody system
- Forward and inverse dynamics

Virtual work

Represent a point $\mathbf{r}_i$ on the articulated body system by a set of generalized coordinates:

$$\mathbf{r}_i = \mathbf{r}_i(q_1, q_2, \ldots, q_n)$$

The virtual displacement of $\mathbf{r}_i$ can be written in terms of generalized coordinates

$$\delta \mathbf{r}_i = \frac{\partial \mathbf{r}_i}{\partial q_1}\delta q_1 + \frac{\partial \mathbf{r}_i}{\partial q_2}\delta q_2 + \ldots + \frac{\partial \mathbf{r}_i}{\partial q_n}\delta q_n$$

The virtual work of force $\mathbf{F}_i$ acting on $\mathbf{r}_i$ is

$$\mathbf{F}_i \delta \mathbf{r}_i = \mathbf{F}_i \sum_j \frac{\partial \mathbf{r}_i}{\partial q_j}\delta q_j$$

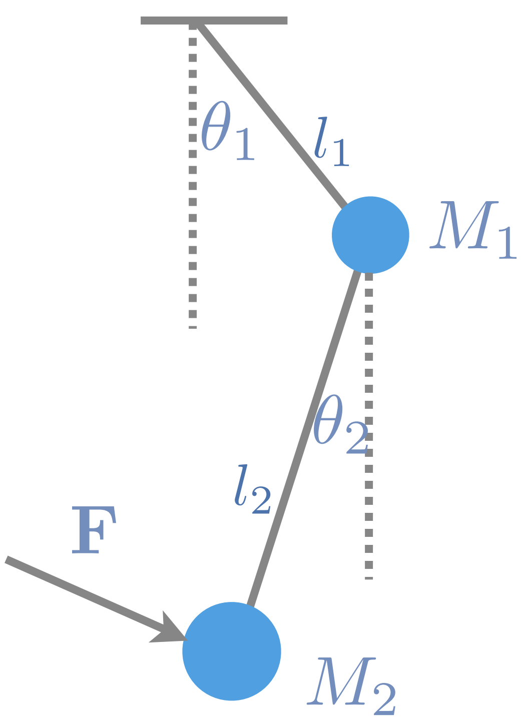
Generalized forces

Define generalized force associated with coordinate q_j

$$Q_j = \mathbf{F}_i \cdot \frac{\partial \mathbf{r}_i}{\partial q_j}$$

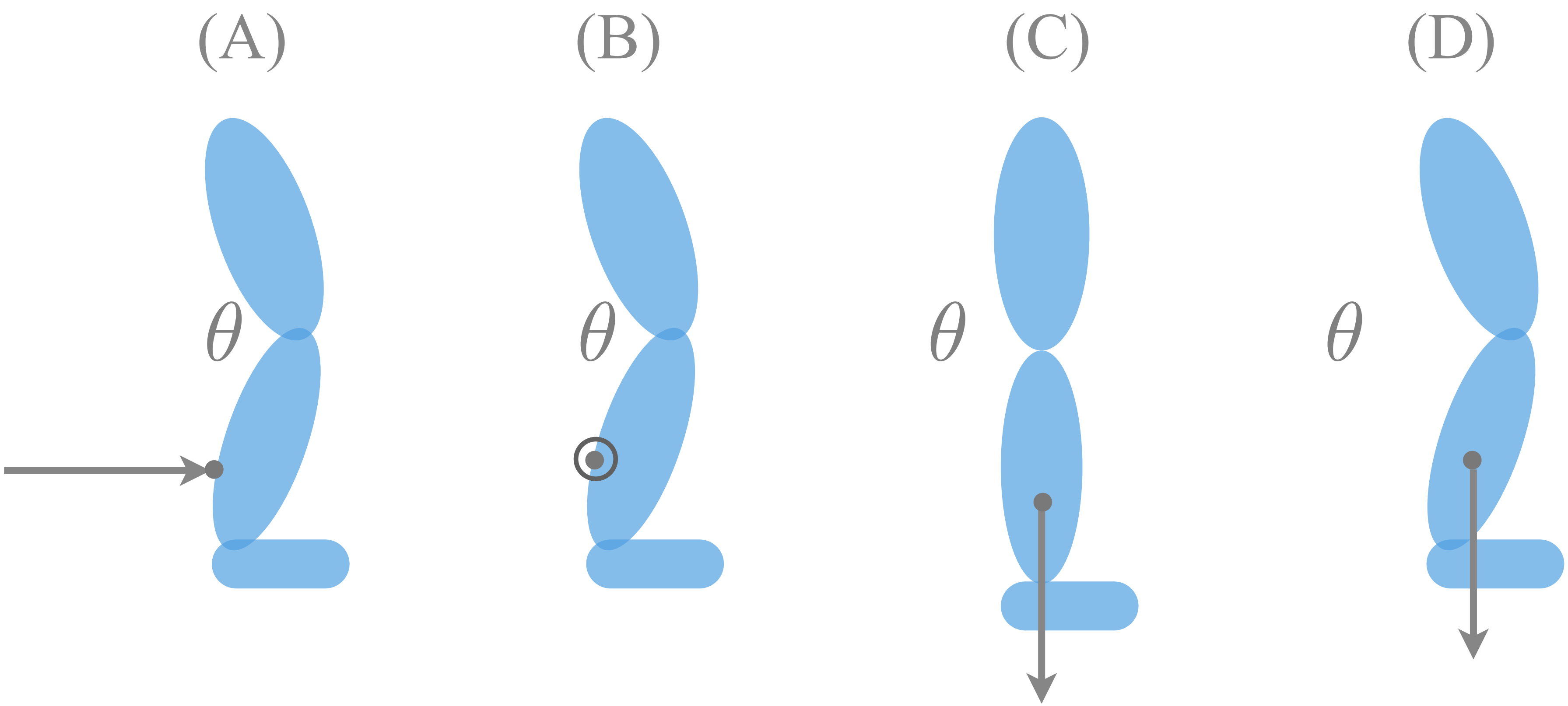
$$\text{virtual work} = \sum_j Q_j \delta q_j$$

Example:



Quiz

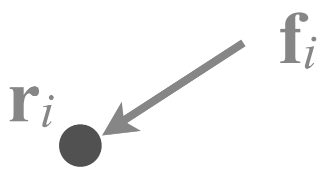
Consider a hinge joint theta. Which one has zero generalized force in theta?



- Generalized coordinates
- Virtual work and generalized forces
- Lagrangian dynamics for mass points
- Lagrangian dynamics for a rigid body
- Lagrangian dynamics for a multibody system
- Forward and inverse dynamics

D'Alembert's principle

- Consider one particle in generalized coordinates under some applied force



$$\mathbf{r}_i = \mathbf{r}_i(q_1, q_2, \ldots, q_{n_j}, t)$$

- Applied force and inertia force are balanced along any virtual displacement

$$\delta W_i = \mathbf{f}_i \cdot \delta \mathbf{r}_i = \mu_i \ddot{\mathbf{r}}_i \cdot \delta \mathbf{r}_i = \sum_j \mu_i \ddot{\mathbf{r}}_i \cdot \frac{\partial \mathbf{r}_i}{\partial q_j} \delta q_j$$

Lagrangian dynamics

$$\frac{d}{dt}\left(\frac{\partial T_i}{\partial \dot{q}_j}\right) - \frac{\partial T_i}{\partial q_j} - Q_{ij} = 0$$

- Equations of motion for one mass point in one generalized coordinate
- T_i: Kinetic energy of mass point $\mathbf{r}_i$
- Q_{ij}: Applied force $\mathbf{f}_i$ projected in generalized coordinate q_j
- For a system with n generalized coordinates, there are n such equations, each of which governs the motion of one generalized coordinate

Vector form

- We can combine *n* scalar equations into the vector form

$$M(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{Q}$$

- Mass matrix: $M(\mathbf{q}) = \sum_i \mu J_i^T J_i$

- Coriolis and centrifugal force: $C = \dot{M}\dot{\mathbf{q}} - \frac{1}{2}\left(\frac{\partial M}{\partial \mathbf{q}}\dot{\mathbf{q}}\right)^T \dot{\mathbf{q}}$

- Generalized coordinates
- Virtual work and generalized forces
- Lagrangian dynamics for mass points
- Lagrangian dynamics for a rigid body
- Lagrangian dynamics for a multibody system
- Forward and inverse dynamics

Newton-Euler equations

- There are infinitely many points contained in each rigid body, how do we derive Lagrange's equations of motion?
- Start out with familiar Newton-Euler equations

$$\begin{pmatrix} m\mathbf{I}_3 & 0 \\ 0 & I_c \end{pmatrix} \begin{pmatrix} \dot{\mathbf{v}} \\ \dot{\omega} \end{pmatrix} + \begin{pmatrix} 0 \\ \omega \times I_c \omega \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \tau \end{pmatrix}$$

- Newton-Euler describes how linear and angular velocity of a rigid body change over time under applied force and torque

Jacobian matrix

- To express in Lagrangian formulation, we need to convert velocity in Cartesian coordinates to generalized coordinates
- Define linear Jacobian, J_v

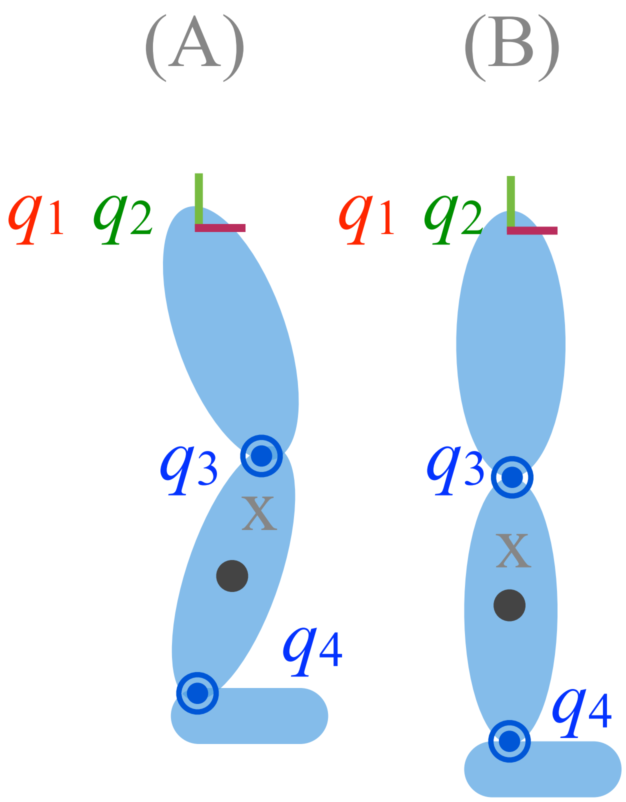
$$\mathbf{v} = \dot{\mathbf{x}}(\mathbf{q}) = \frac{\partial \mathbf{x}}{\partial \mathbf{q}}\dot{\mathbf{q}} \equiv J_v \dot{\mathbf{q}}$$

- Define angular Jacobian, J_ω

$$\boldsymbol{\omega} = J_\omega \dot{\mathbf{q}}$$

$$\text{where} \quad \begin{aligned} [\boldsymbol{\omega}] &= \dot{R}(\mathbf{q})R^T(\mathbf{q}) \\ &= \sum_j \frac{\partial R}{\partial q_j} R^T \dot{q}_j \equiv \sum_j [\mathbf{j}_j]\dot{q}_j \end{aligned}$$

Quiz



$$\mathbf{v} = \dot{\mathbf{x}}(\mathbf{q}) = \frac{\partial \mathbf{x}}{\partial \mathbf{q}}\dot{\mathbf{q}} \equiv J_v \dot{\mathbf{q}}$$

What is the dimension of the Jacobian?

Which elements in the Jacobian are zero?

Lagrangian dynamics

- Substitute Cartesian velocity with generalized velocity in Newton-Euler equations using Jacobian matrices

$$M_c(\dot{J}\dot{\mathbf{q}}) + \begin{pmatrix} 0 \\ (J_\omega\dot{\mathbf{q}}) \times I_c J_\omega \dot{\mathbf{q}} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \boldsymbol{\tau} \end{pmatrix}$$

$$\Rightarrow M_c J\ddot{\mathbf{q}} + M_c\dot{J}\dot{\mathbf{q}} + [\tilde{\omega}]M_c J\dot{\mathbf{q}} = \begin{pmatrix} \mathbf{f} \\ \boldsymbol{\tau} \end{pmatrix}$$

where, $[\tilde{\omega}] = \begin{pmatrix} 0 & 0 \\ 0 & [J_\omega\dot{\mathbf{q}}] \end{pmatrix}$

Lagrangian dynamics

- Projecting into generalized coordinates by multiplying Jacobian transpose on both sides

$$\left(J^T M_c J\right)\ddot{\mathbf{q}} + \left(J^T M_c \dot{J} + J^T[\tilde{\boldsymbol{\omega}}]M_c J\right)\dot{\mathbf{q}} \;=\; J_v^T \mathbf{f} + J_\omega^T \boldsymbol{\tau}$$

- This equation is exactly the vector form of Lagrange's equations of motion

$$M(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{Q}$$

where,

$$\begin{aligned} M(\mathbf{q}) &= J^T M_c J \\ C(\mathbf{q}, \dot{\mathbf{q}}) &= (J^T M_c \dot{J} + J^T[\tilde{\boldsymbol{\omega}}]M_c J)\dot{\mathbf{q}} \\ \mathbf{Q} &= J_v^T \mathbf{f} + J_\omega^T \boldsymbol{\tau} \end{aligned}$$

- Generalized coordinates
- Virtual work and generalized forces
- Lagrangian dynamics for mass points
- Lagrangian dynamics for a rigid body
- Lagrangian dynamics for a multibody system
- Forward and inverse dynamics

Multibody dynamics

- Once Newton-Euler equations are expressed in generalized coordinates, multibody dynamics is a straightforward extension of a single rigid body

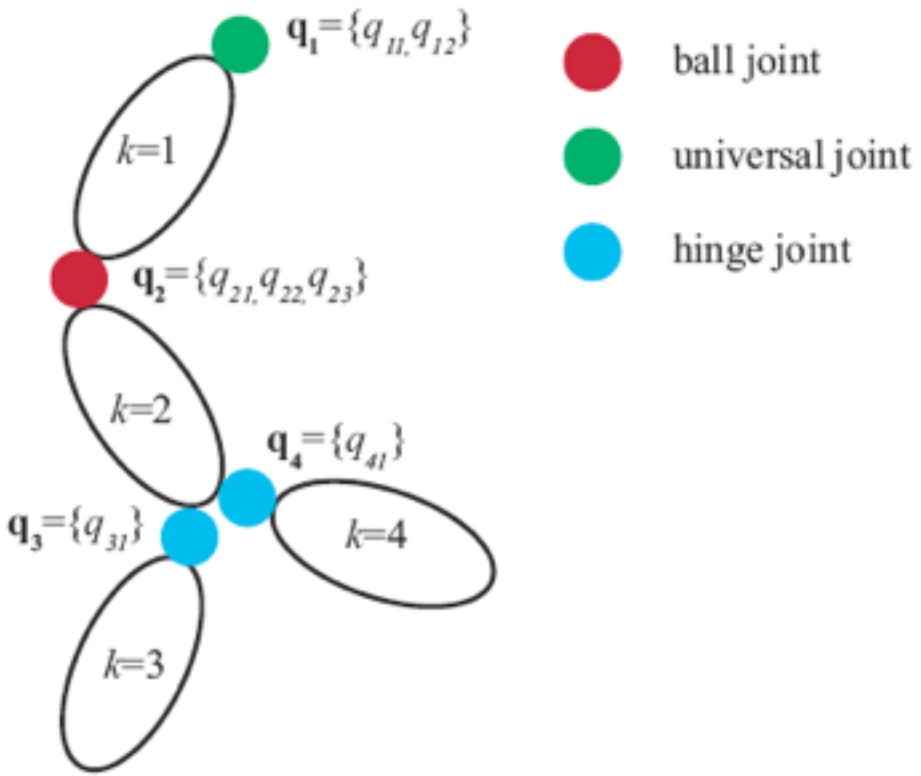
$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\mathbf{q}}}\right) - \frac{\partial T}{\partial \mathbf{q}} = \sum_k \left(\frac{d}{dt}\left(\frac{\partial T_k}{\partial \dot{\mathbf{q}}}\right) - \frac{\partial T_k}{\partial \mathbf{q}} \right)$$

$$= \sum_k \left(J_k^T M_{ck} J_k\right) \ddot{\mathbf{q}} + \sum_k \left(J_k^T M_{ck} \dot{J}_k + J_k^T [\tilde{\omega}_k] M_{ck} J_k \right) \dot{\mathbf{q}}$$

- The only tricky part is to compute Jacobian in a hierarchical multibody system

Notations

- $p(k)$ returns index of parent link of link k
- $n(k)$ returns number of DOFs in joint that connects link k to parent link $p(k)$
- R_k is local rotation matrix for link k and depends only on DOFs $\mathbf{q}_k$
- $R^0{}_k$ is transformation chain from world to local frame of link k



Jacobian for each link

- Define a Jacobian for each rigid link that relates its Cartesian velocity to generalized velocity of entire system
- Define linear Jacobian for link *k*

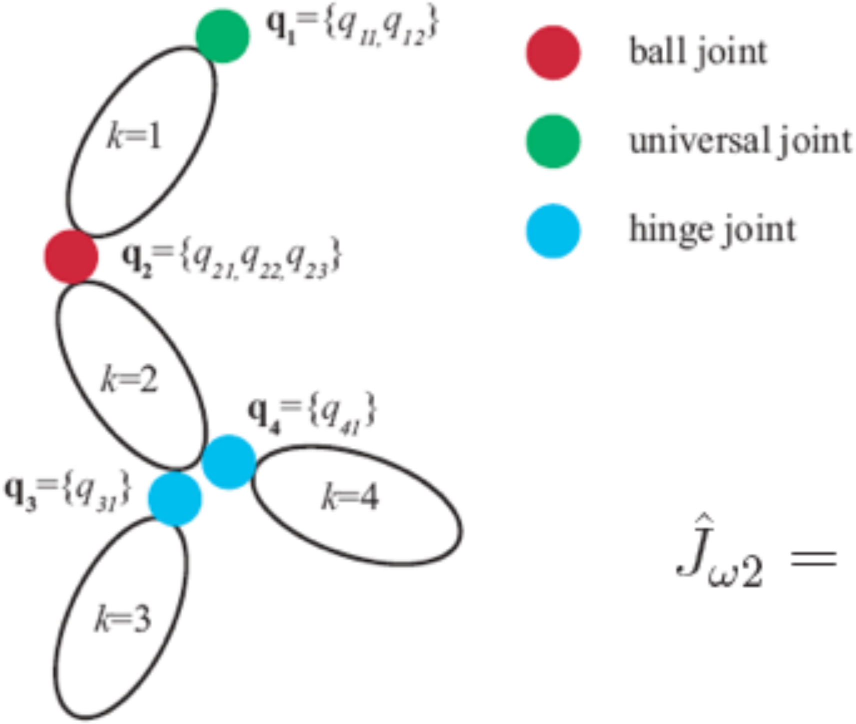
$$\mathbf{v}_k = J_{vk}\dot{\mathbf{q}}, \quad \text{where} \;\; J_{vk} = \frac{\partial \mathbf{x}_k}{\partial \mathbf{q}} = \frac{\partial W_k^0 \mathbf{c}_k}{\partial \mathbf{q}}$$

- Define angular Jacobian for link *k*

$$\boldsymbol{\omega}_k \;\; = \;\; \boldsymbol{\omega}_{p(k)} + R_{p(k)}^0 \hat{J}_{\omega k}\dot{\mathbf{q}}_k \;\; \equiv \;\; J_{\omega k}\dot{\mathbf{q}}$$

$$\text{where} \quad J_{\omega k} = \begin{pmatrix} \hat{J}_{\omega 1} & \ldots & R_{p(l)}^0 \hat{J}_{\omega l} & \ldots & \mathbf{0} & \ldots \end{pmatrix}$$

Example



$$
\begin{aligned}
\omega_1 &= (\hat{J}_{\omega 1} \;\; 0 \;\; 0 \;\; 0)\dot{\mathbf{q}} \\
\omega_4 &= (\hat{J}_{\omega 1} \;\; R_1^0 \hat{J}_{\omega 2} \;\; 0 \;\; R_2^0 \hat{J}_{\omega 4})\dot{\mathbf{q}}
\end{aligned}
$$

$$
\hat{J}_{\omega 2} = \left(\begin{matrix} 1 \\ 0 \\ 0 \end{matrix} \quad R^{(x)} \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \quad R^{(x)} R^{(y)} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \right)
$$

- Generalized coordinates
- Virtual work and generalized forces
- Lagrangian dynamics for mass points
- Lagrangian dynamics for a rigid body
- Lagrangian dynamics for a multibody system
- Forward and inverse dynamics

Forward vs inverse dynamics

- Same equations of motion can solve two problems

$$M(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{Q}$$

- Forward dynamics $\ddot{\mathbf{q}} = -M(\mathbf{q})^{-1}(C(\mathbf{q}, \dot{\mathbf{q}}) - \mathbf{Q})$
    - given a set of forces and torques on the joints, calculate the motion
- Inverse dynamics $\mathbf{Q} = M(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}})$
    - given a description of motion, calculate the forces and torques that give rise to it

Quiz

- Which problem is inverse dynamics?
  - Given the current state of a robotic arm, compute its next state under gravity.
  - Given desired joint angle trajectories for a robotic arm, compute the joint torques required to achieve the trajectories.
  - Given the desired position for a point on a robotic arm, compute the joint angles of the arm to achieve the position.