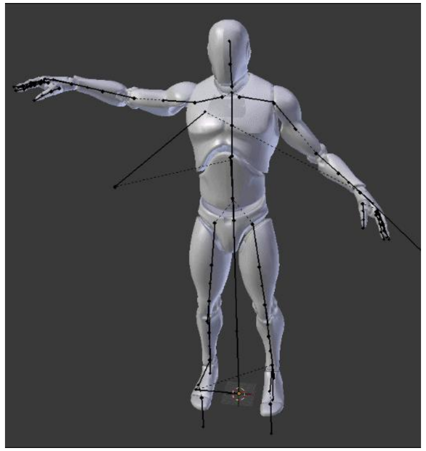
关节(角色)动画



金小刚

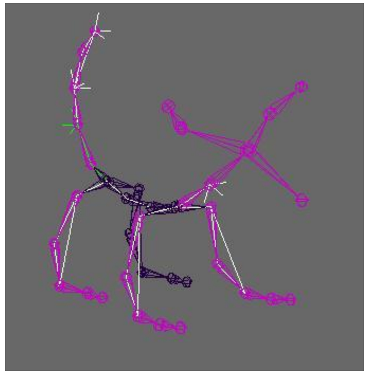
Email: jin@cad.zju.edu.cn

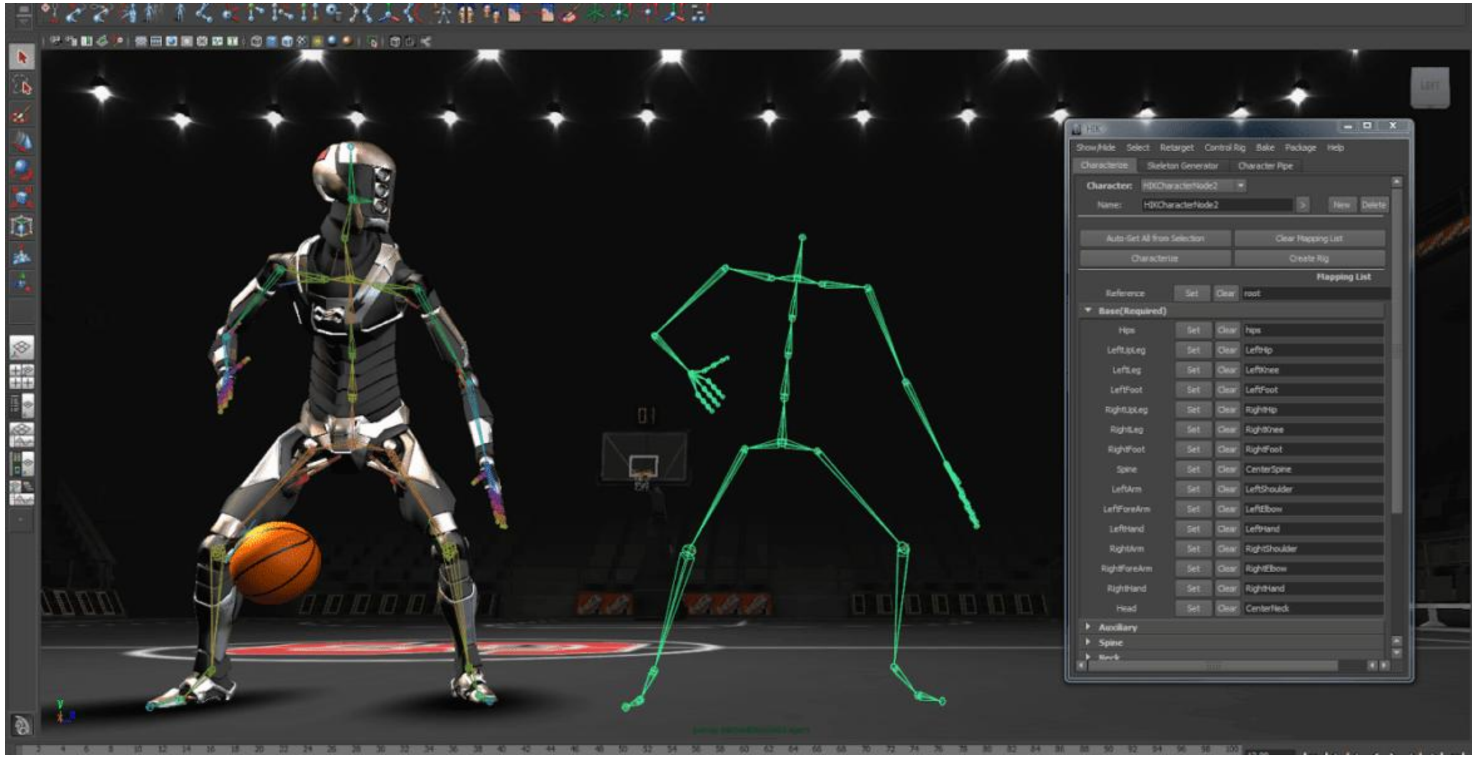
浙江大学CAD&CG国家重点实验室

紫金港校区蒙民伟楼512

关节动画

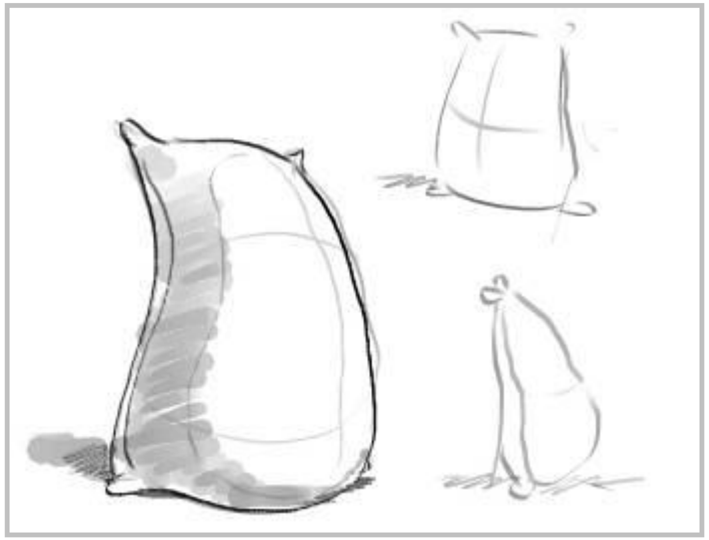
- 在计算机动画中，加入人、动物这样的角色会使画面活泼、具有生机活力。关节动画是实现这类角色动画不可缺少的部分。

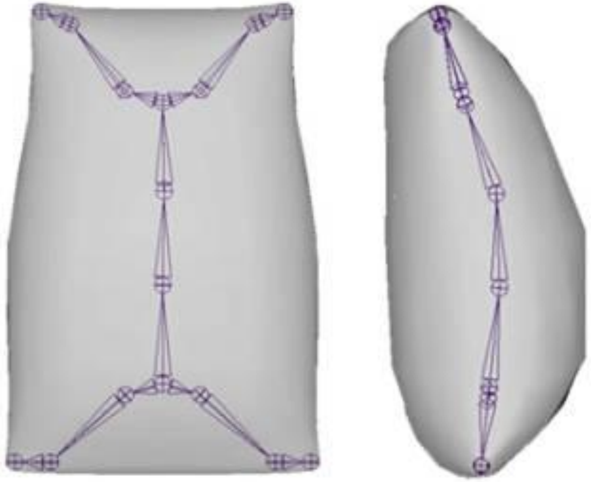


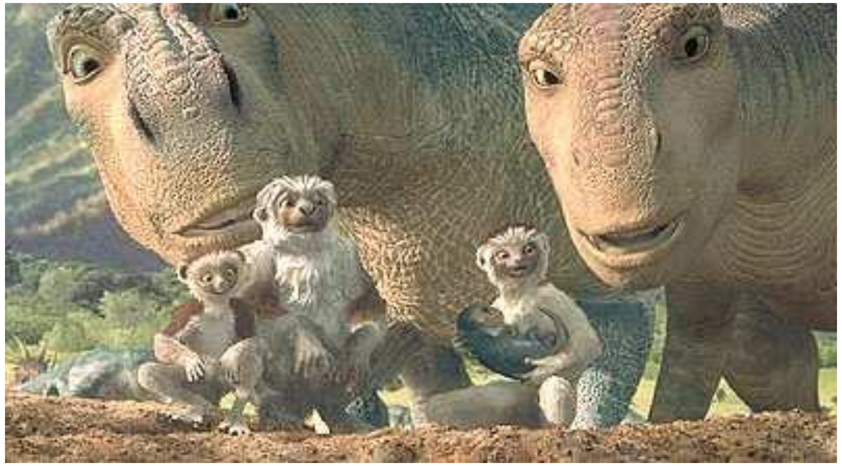
可用于游戏...

也可用于无生命的物体以创建拟人效果





电影



《恐龙》剧照

《海底总动员》剧照

《泰坦尼克号》剧照

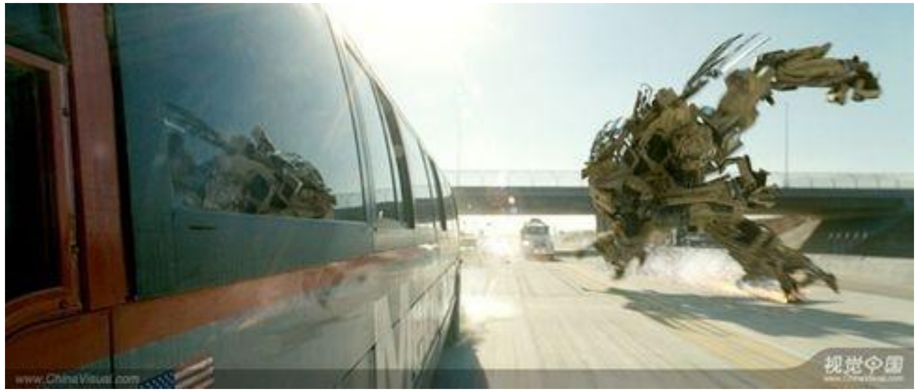
《金刚》剧照

《精灵鼠小弟2》剧照

《纳尼亚传奇》剧照

《加勒比海盗》剧照

《加菲猫2之双猫记》剧照



《变形金刚》剧照

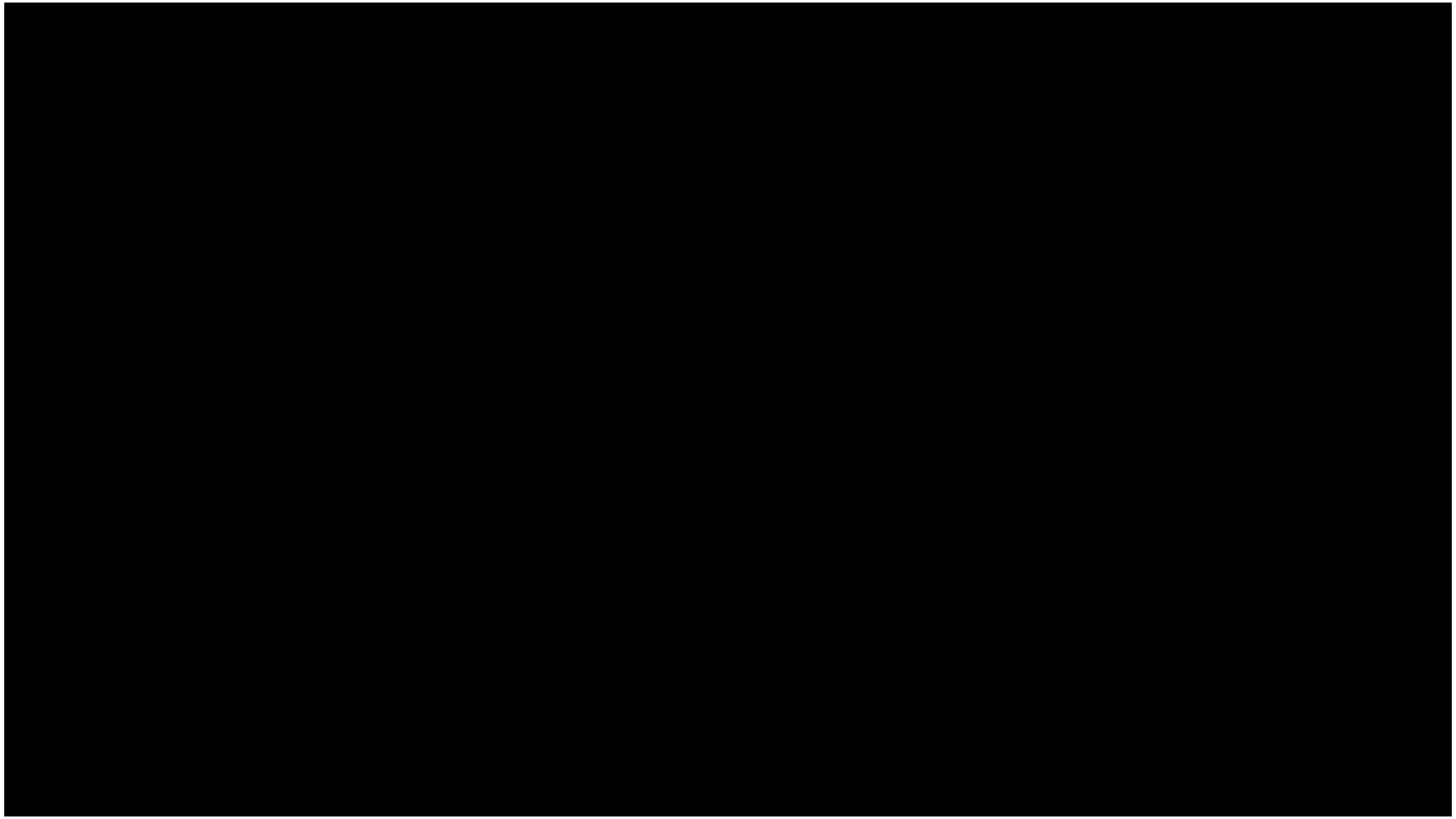



《史前一万年》剧照

《尼斯湖怪：深水传说》剧照

Polar Express (2004)

Alita- Battle Angel (2018)

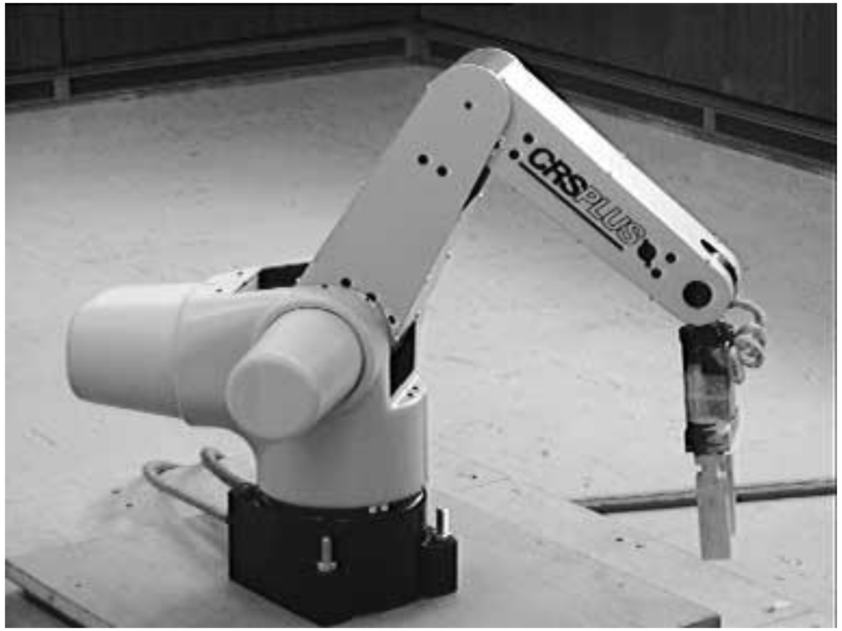


一些概念：模型层次 vs. 运动层次

- 关节动画中通常用**相对运动**
  - 一个物体相对于另外一个物体运动
- 物体层次 + 相对运动 构成 **运动层次**
  - 连杆
  - 运动通常受限制

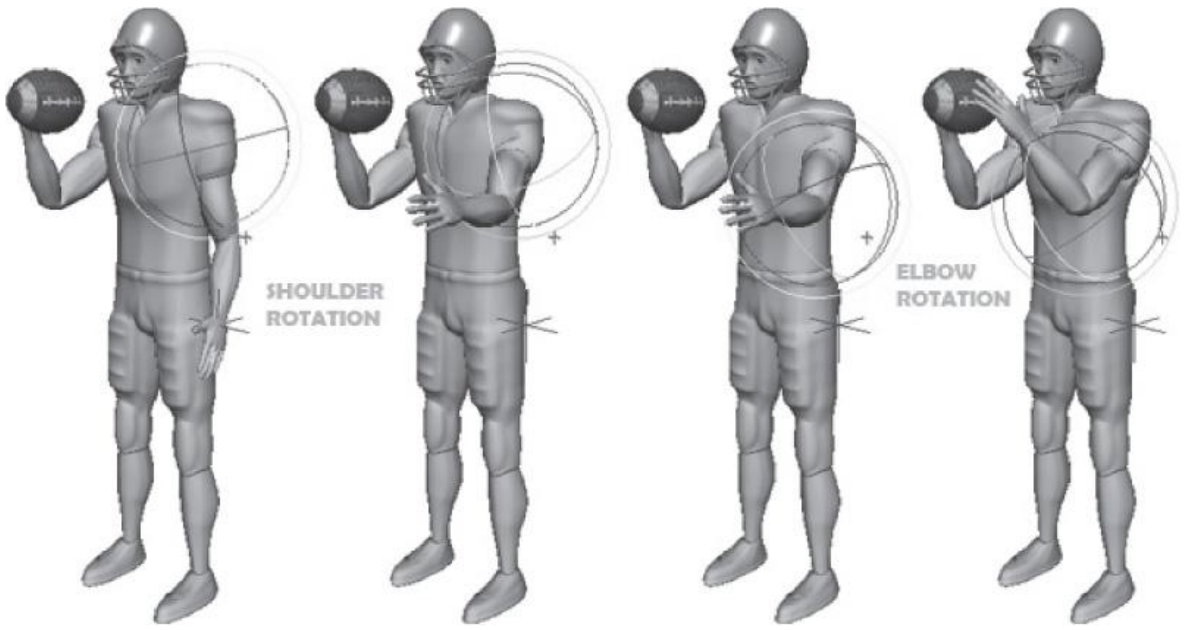
运动学

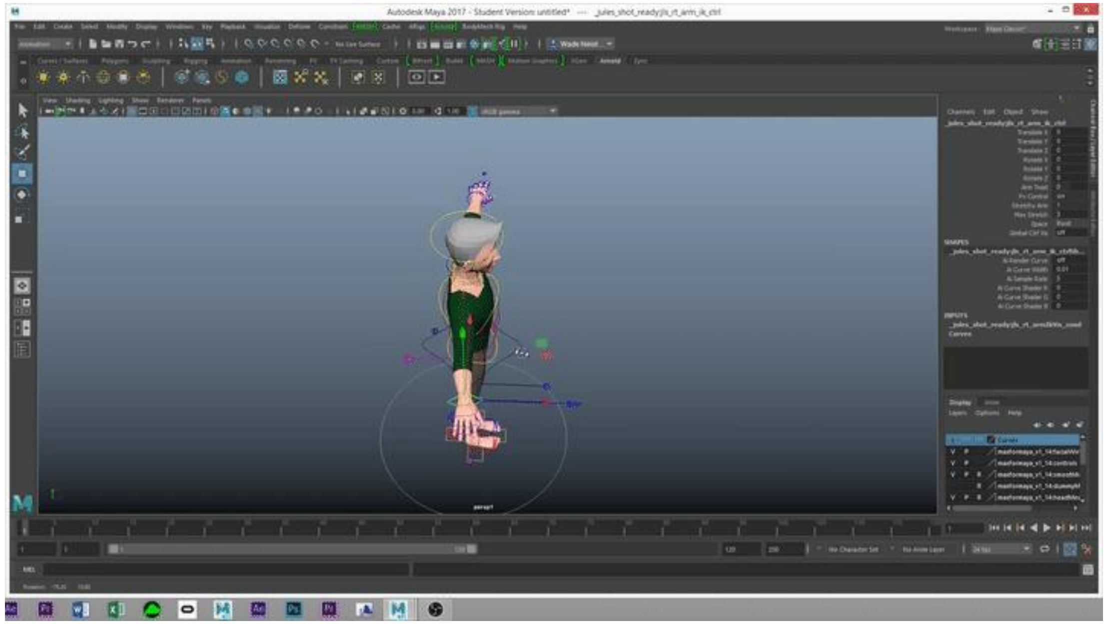
- 如何通过设置位置随时间的参数来对连杆设置动画?
- 运动学不考虑引起运动的力（区别于动力学）



正向运动学(Forward Kinematics)

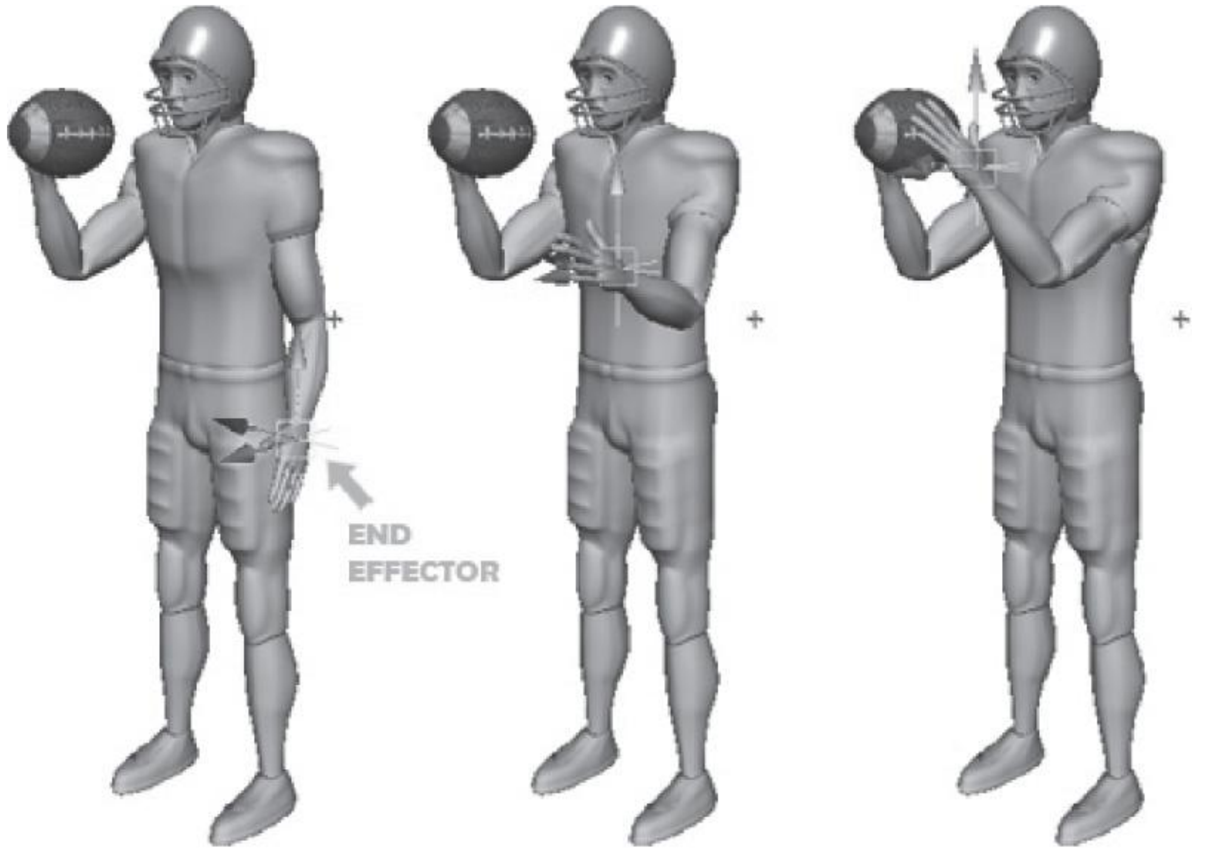
- 动画师通过**直接指定**关节处的关节运动参数来控制物体的运动

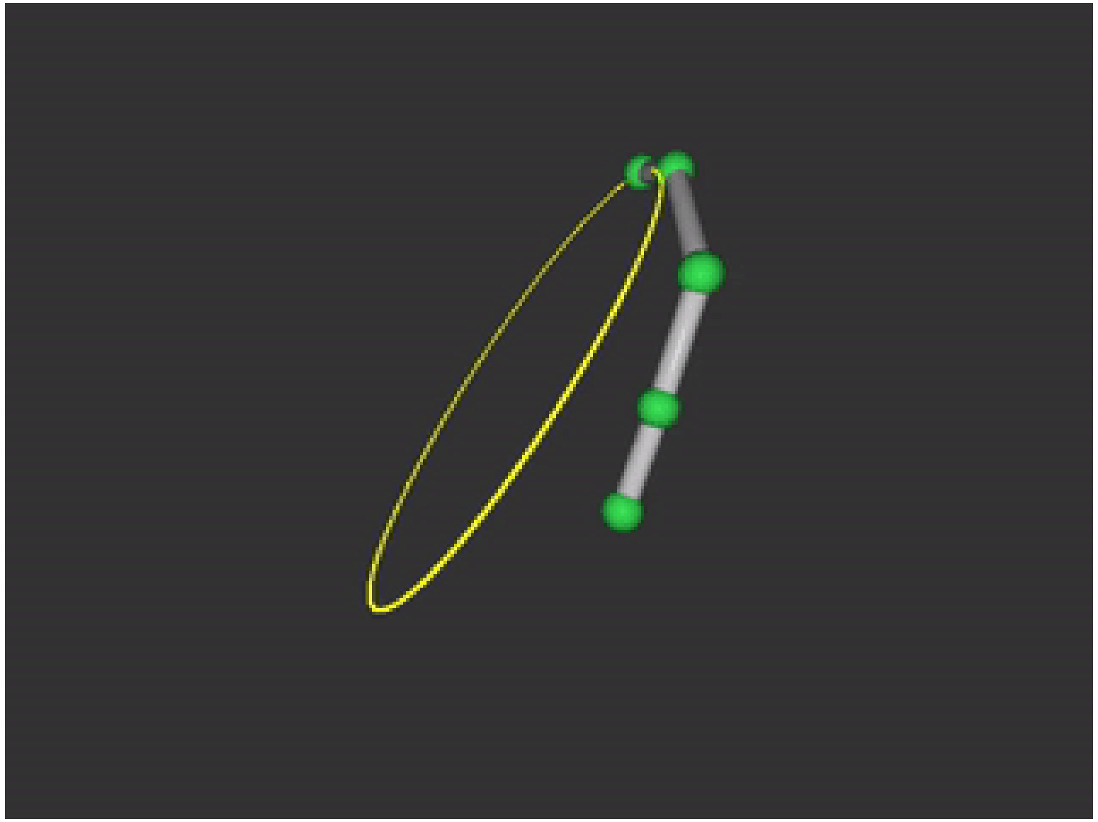




逆向运动学(Inverse Kinematics, IK)

- 动画师指定目标位置，系统求解满足要求的关节角。



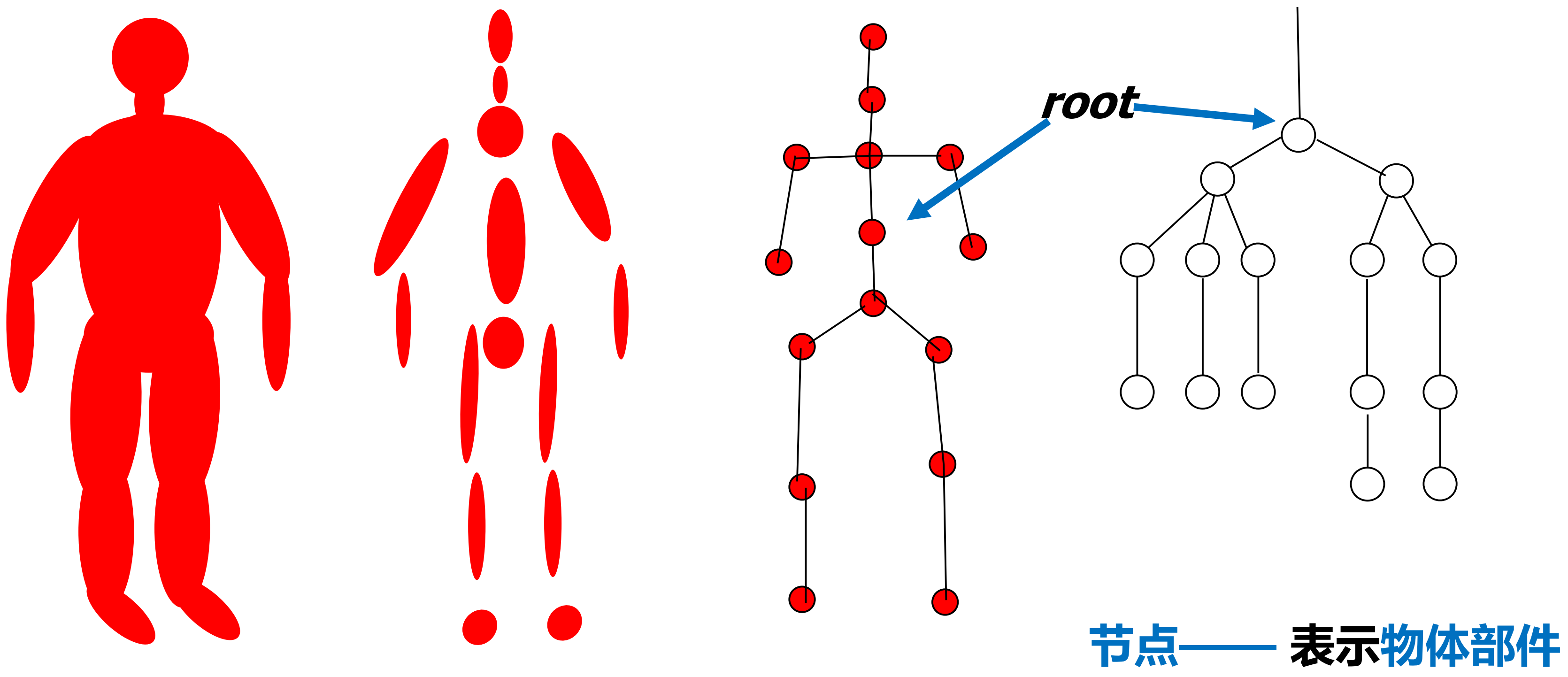


FK vs. IK优缺点

- 正向运动学：
  - **优点：** 精确、确定性、实时。
  - **缺点：** 难以控制末端影响器的位置。如果你想让末端影响器到达一个特定的位置，通过调整关节角度来达到这一点会非常困难和不直观。
- 逆向运动学：
  - **优点：** 直观的末端影响器位置控制、应用广泛。
  - **缺点：** 可能有多个解、可能没有解、计算复杂、可能需要约束。

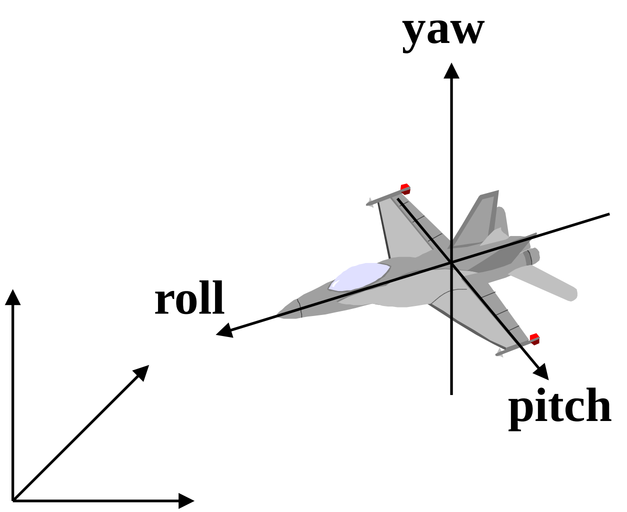
关节模型

- 把关节角色表示为一系列通过关节(joints)相连接的连杆(links)



自由度(Degrees of Freedom, DOF)

- 完全指定一个物体运动所需的最小坐标数目

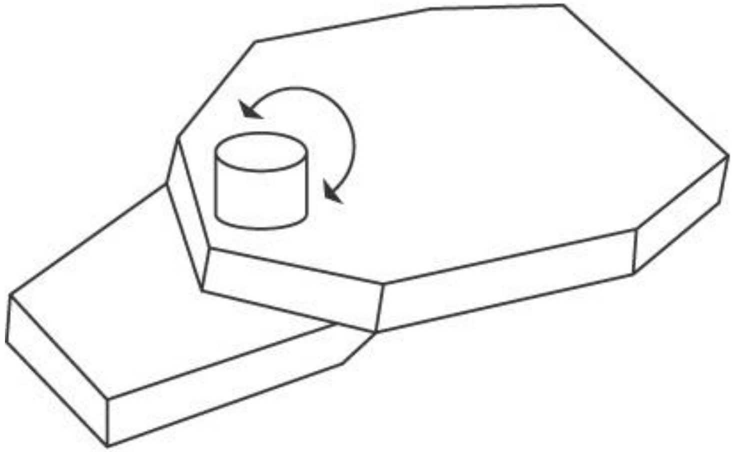


6 DOF: x, y, z, raw, pitch, yaw

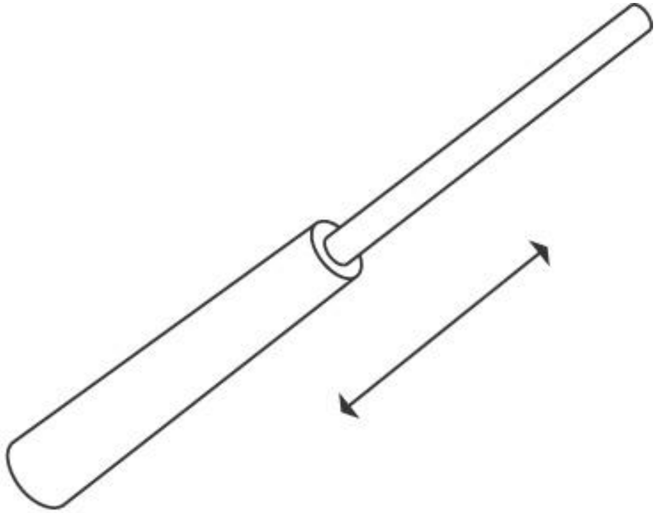
单个自由度关节

- 允许在一个方向运动

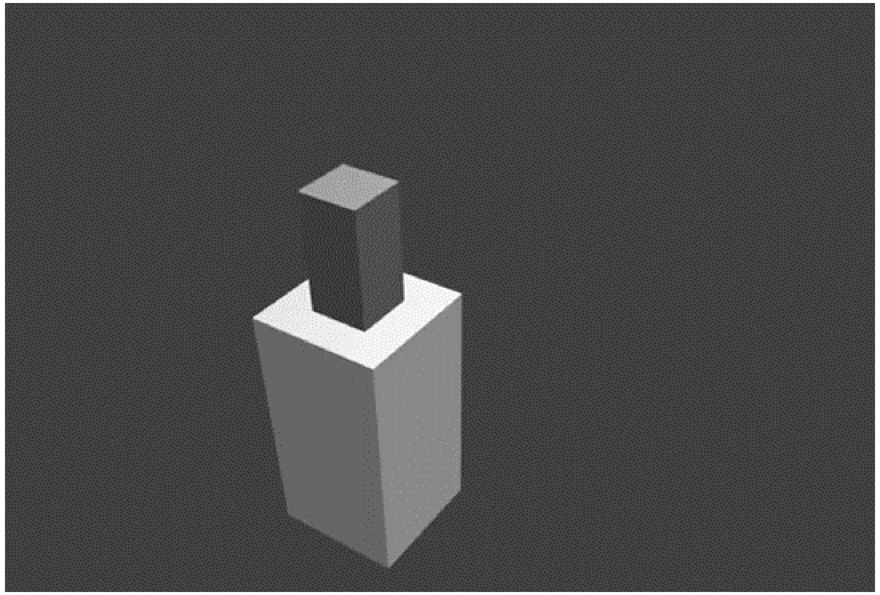
  例如: 转动关节, 移动关节(Prismatic joint)



Revolute joint

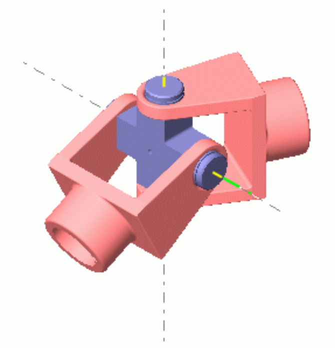


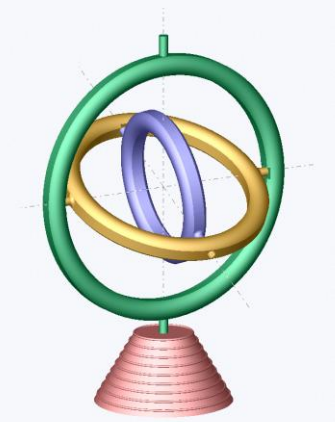
Prismatic joint



复杂关节

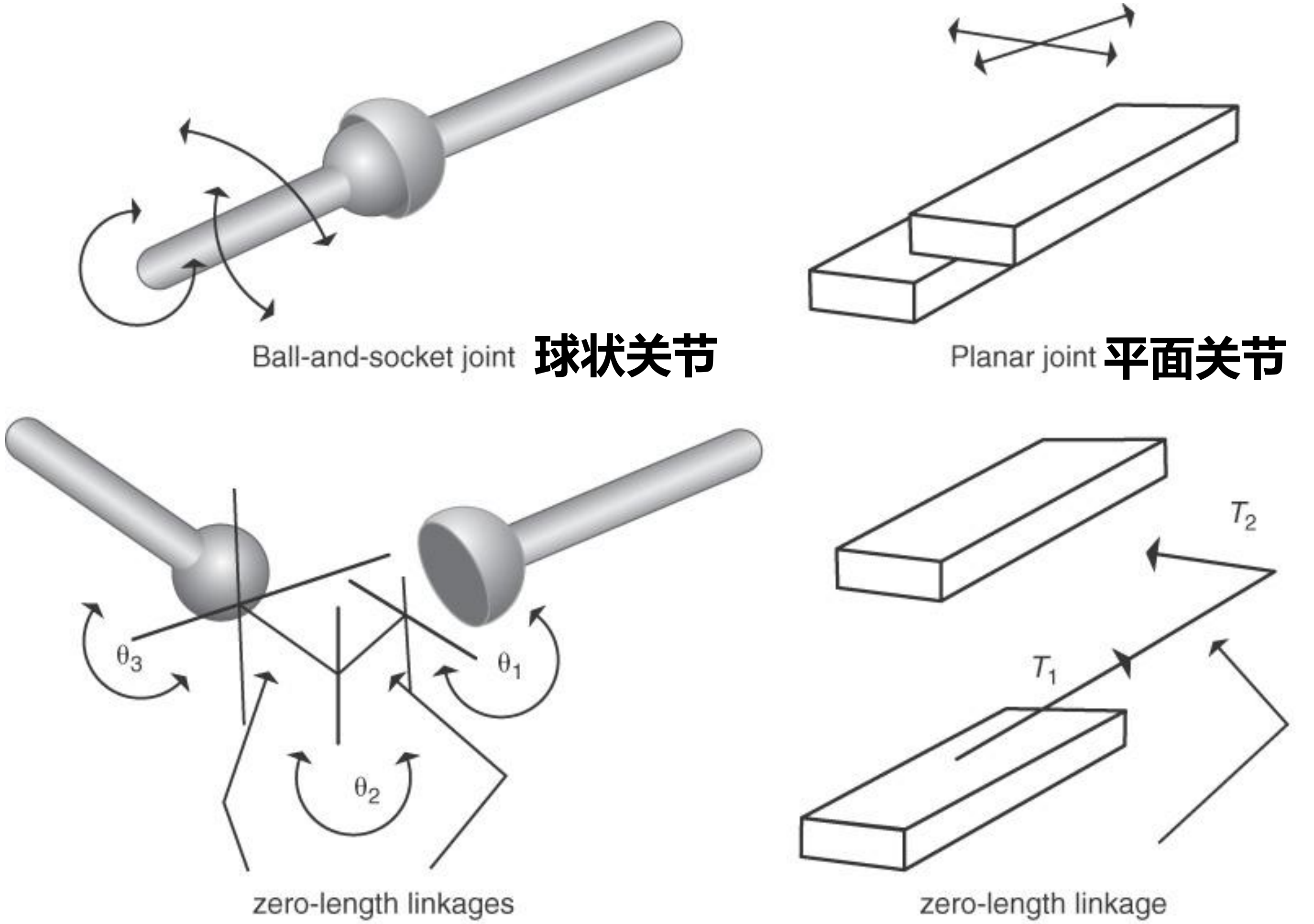
- 复杂关节
  - 2自由度关节
  - 3自由度关节
    - 万向节(Gimbal)
    - 球状关节(如肩、膝关节) (Ball and socket joint)





- n个自由度的复杂关节可看成由n个自由度为1的关节通过 n-1 个长度为0的连杆相连。

复杂关节

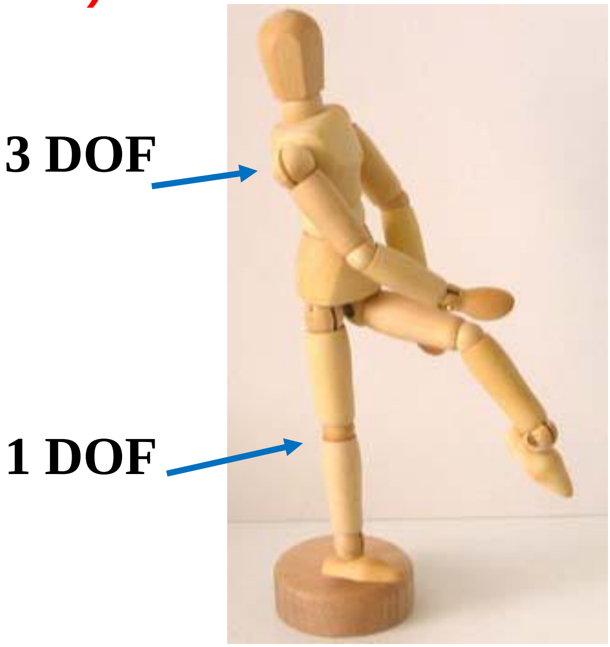


Ball-and-socket joint modeled as 3 one-degree joints with zero-length links

Planar joint modeled as 2 one-degree prismatic joints with zero-length links

人体模型的自由度

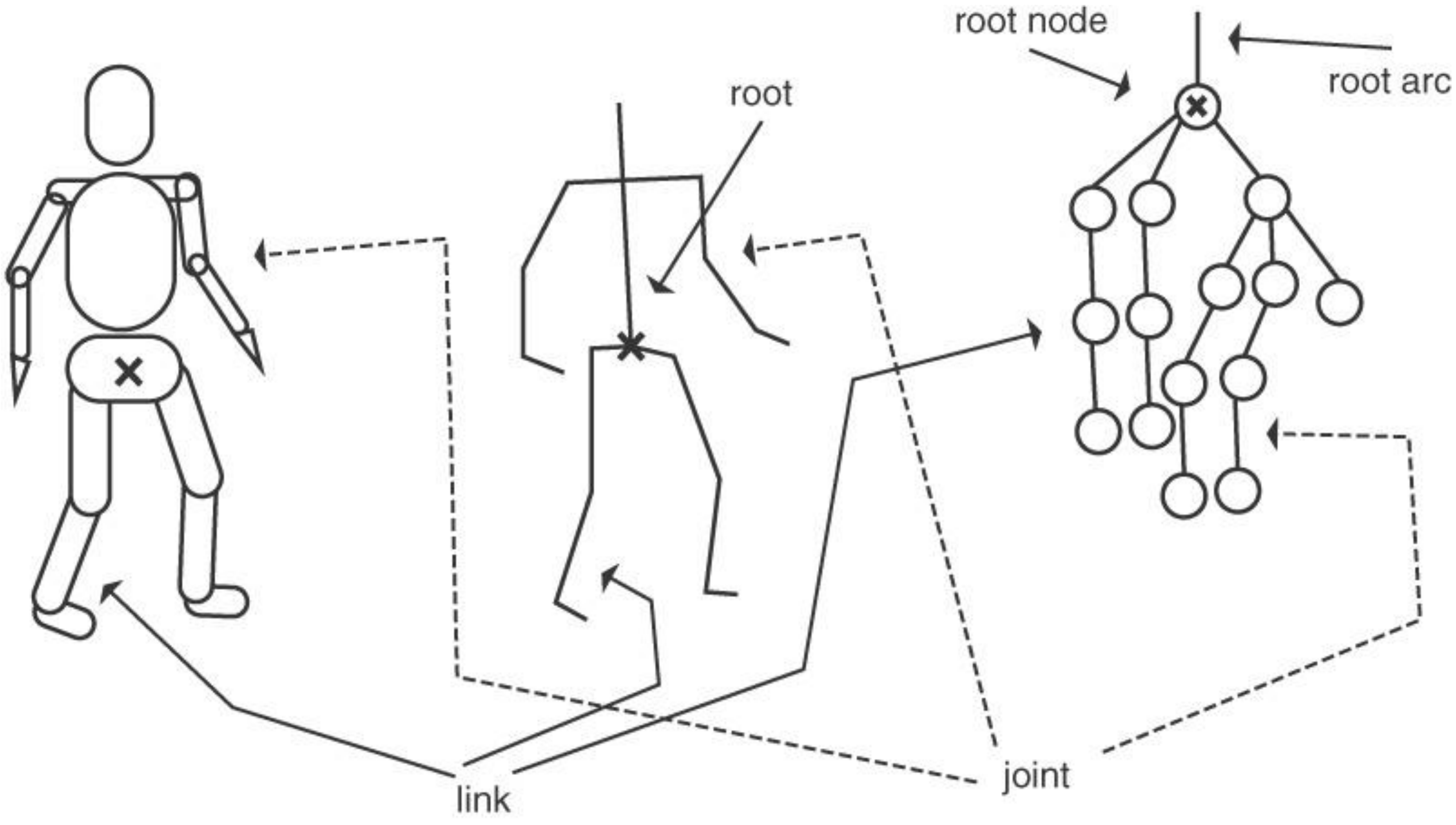
- 根节点(Root): 3 translational DOF + 3 rotational DOF
- 人体模型通常用旋转关节(Rotational joints)
- 每个关节至多有3个自由度
  - 肩关节(Shoulder): 3 DOF
  - 腕关节(Wrist): 2 DOF
  - 膝关节(Knee): 1 DOF



层次模型的数据结构

- 层次连杆可用一树状结构来表示
  - 根节点——对应于物体的根部，其位置在世界坐标系中给出
  - 其它节点——相对于根节点来表示
  - 叶节点
- 层次结构与树之间的映射
  - 节点—— 表示物体部件
  - 连接弧——表示层次结构中应用于物体部件之间的**关节**或变换

层次模型的数据结构



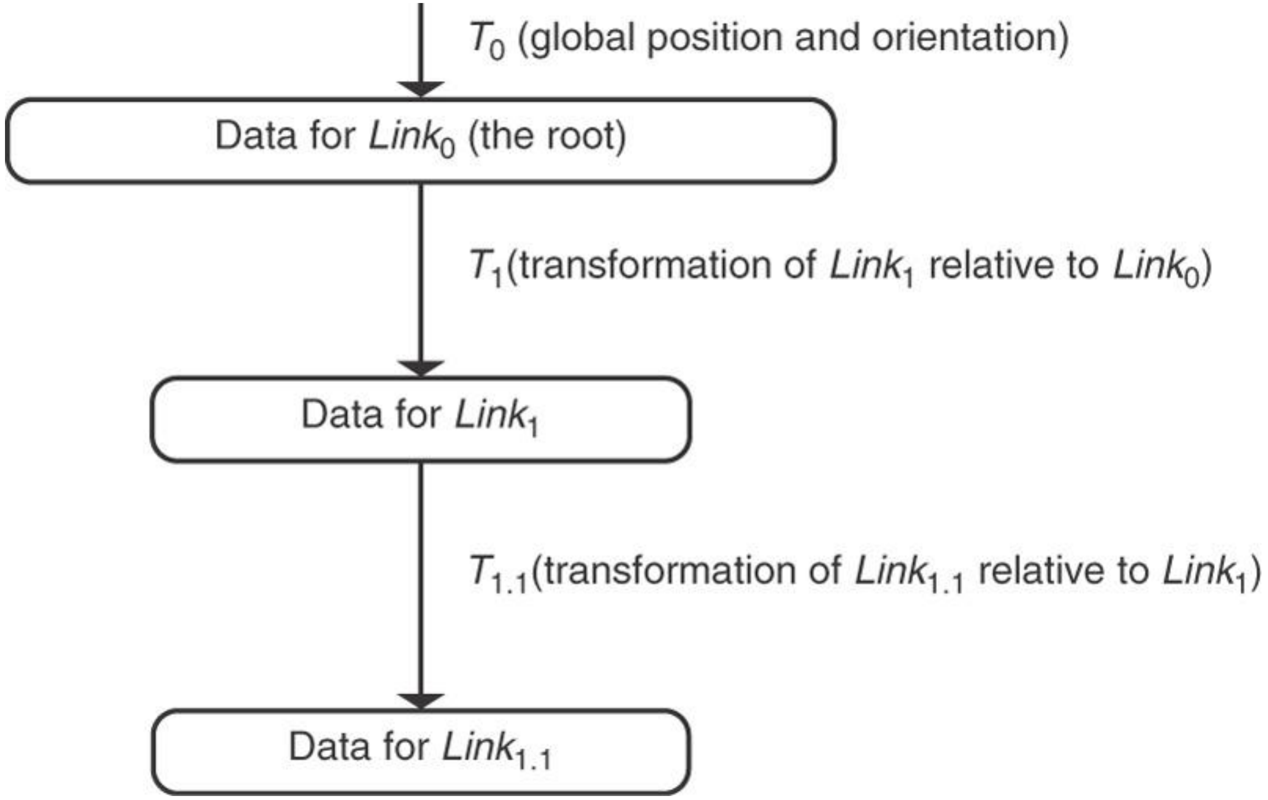
Articulated figure

Abstract hierarchical representation

Tree structure

层次模型的树状结构(平移变换)

T_0： Link$_0$到世界坐标系位置和方向的变换



平移

V_0: Link_0的一个顶点

它的世界坐标为: $V_0{}' = T_0 V_0$

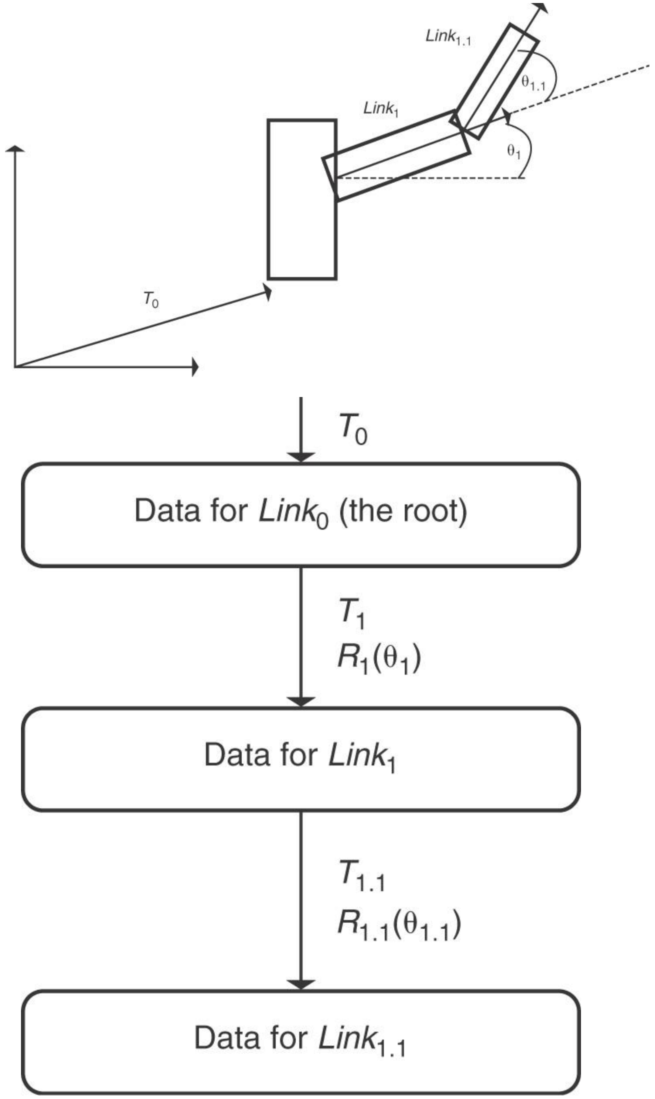
V_1: Link_1的一个顶点

它的世界坐标为: $V_1{}' = T_0 T_1 V_1$

$V_{1,1}$: $\mathrm{Link}_{1,1}$的一个顶点

它的世界坐标为: $V_{1,1}{}' = T_0 T_1 T_{1,1} V_{1,1}$

层次模型的树状结构（旋转变换）



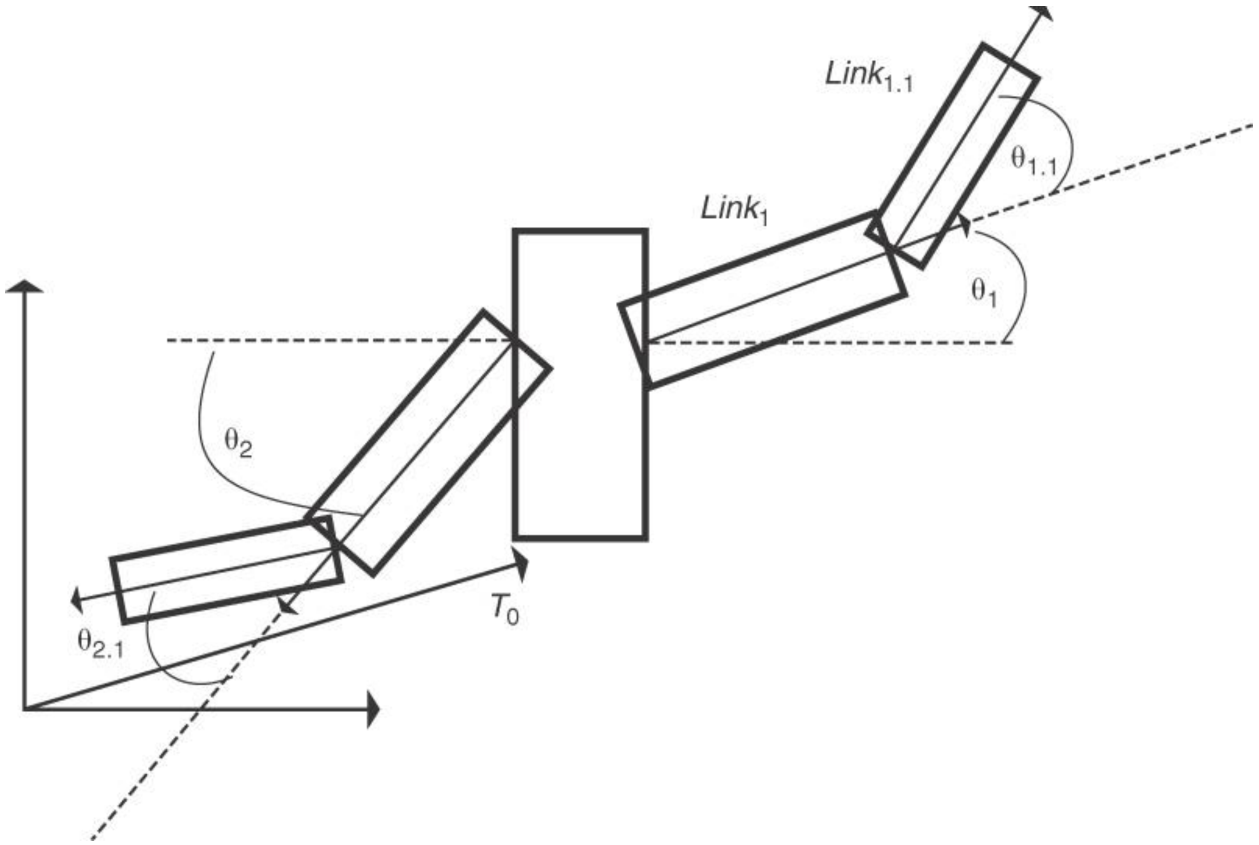
V_0: Link_0 的一个顶点

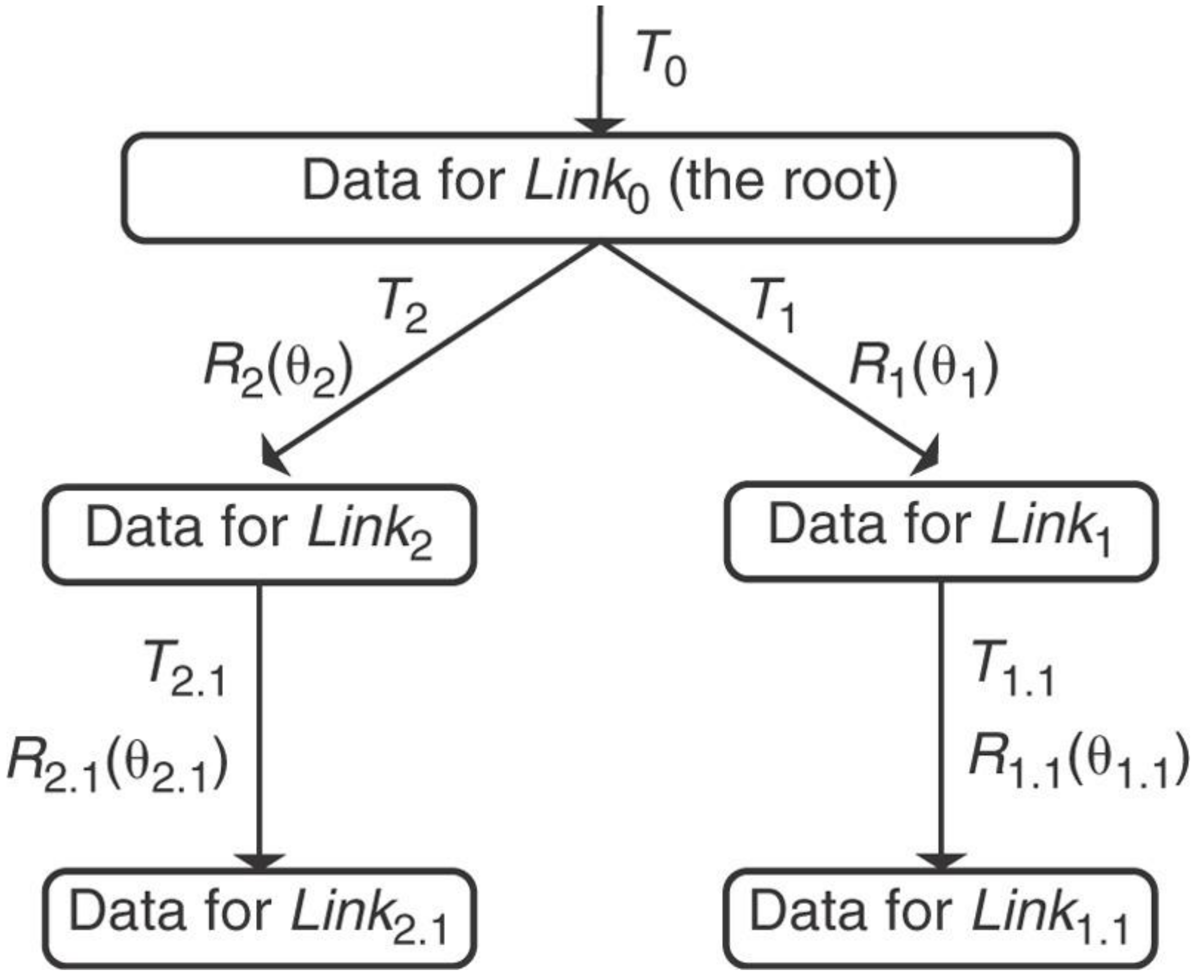
它的世界坐标为: $V_0' = T_0 V_0$

V_1: Link_1 的一个顶点
它的世界坐标为: $V_1{}' = T_0 T_1 R_1(\theta_1) V_1$

$V_{1,1}$: $\mathrm{Link}_{1,1}$ 的一个顶点
它的世界坐标为: $V_{1,1}' = T_0 T_1 R_1(\theta_1) T_{1,1} R_{1,1}(\theta_{1,1}) V_{1,1}$

双肢体(Two Appendages)层次模型

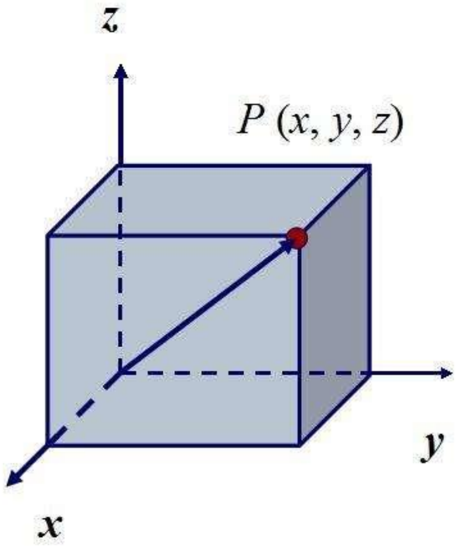




关节空间 vs. 笛卡尔空间

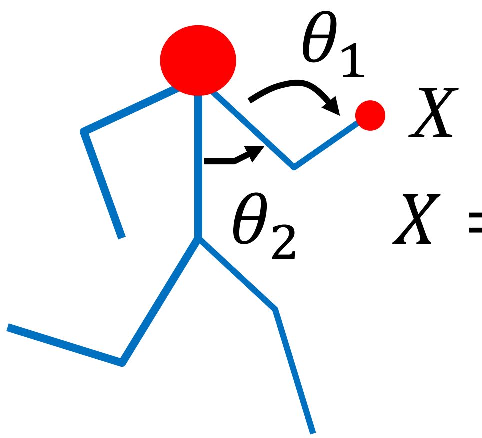
- **关节空间**
  - 由关节角形成的空间
  - 可对所有关节进行细微控制

- **笛卡尔空间**
  - 三维空间
  - 指定与环境的相互作用



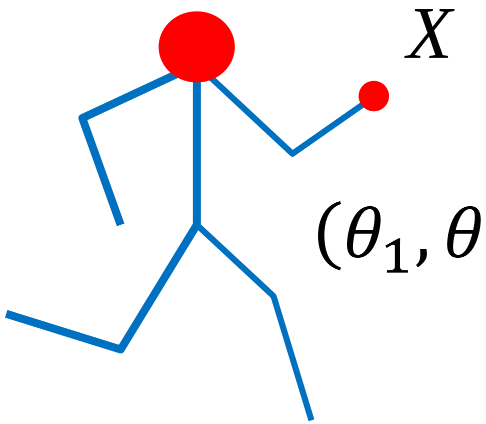
正向和逆向运动学

- 正向运动学(Forward kinematics)
  - 从关节空间映射到笛卡尔空间
- 逆向运动学(Inverse kinematics)
  - 从笛卡尔空间映射到关节空间



$$X = f(\theta_1, \theta_2)$$

Forward Kinematics

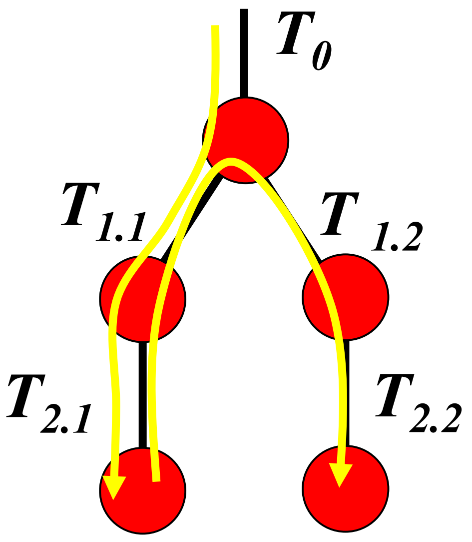


$$(\theta_1, \theta_2) = f^{-1}(X)$$

Inverse Kinematics

正向运动学——树遍历

- 计算整棵树
  - 从根节点到叶节点进行深度优先遍历
  - 重复以下步骤，直到所有节点和连接弧都被访问过
    - 对树进行回溯，直到遇到一个未被访问过的向下连接弧
    - 对向下连接弧进行遍历
  - 在实现阶段...
    - 需要堆栈存贮变换



$M = I$

$M = T_0$

$M = T_0*T_{1.1}$

$M = T_0*T_{1.1}*T_{2.1}$

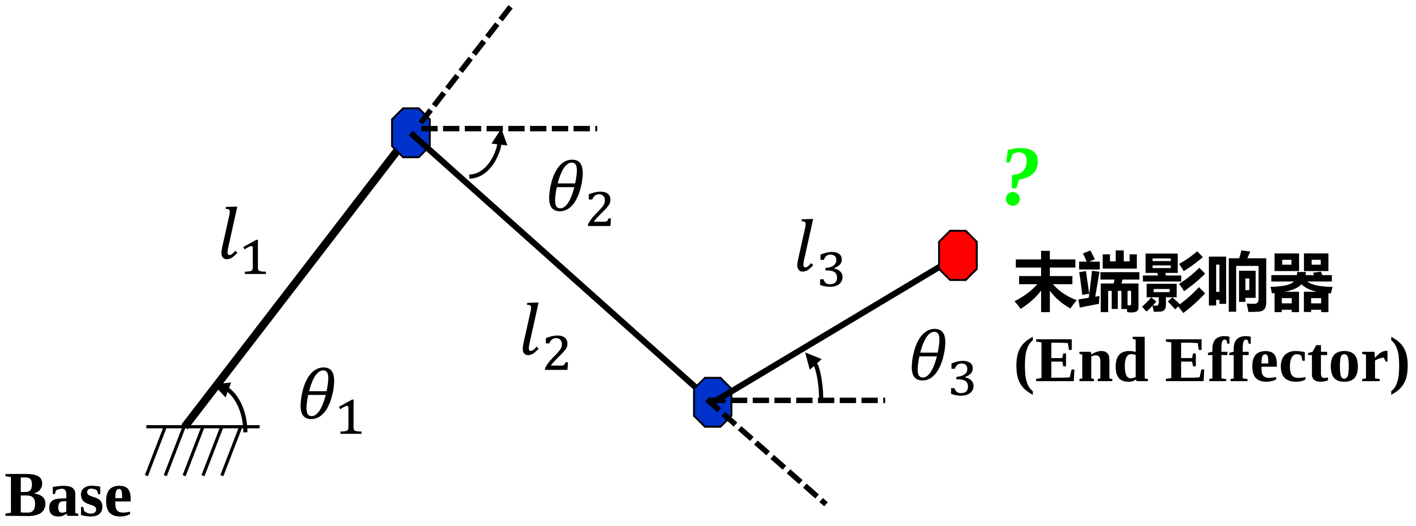
$M = T_0*T_{1.1}$

$M = T_0$

$M = T_0*T_{1.2}$

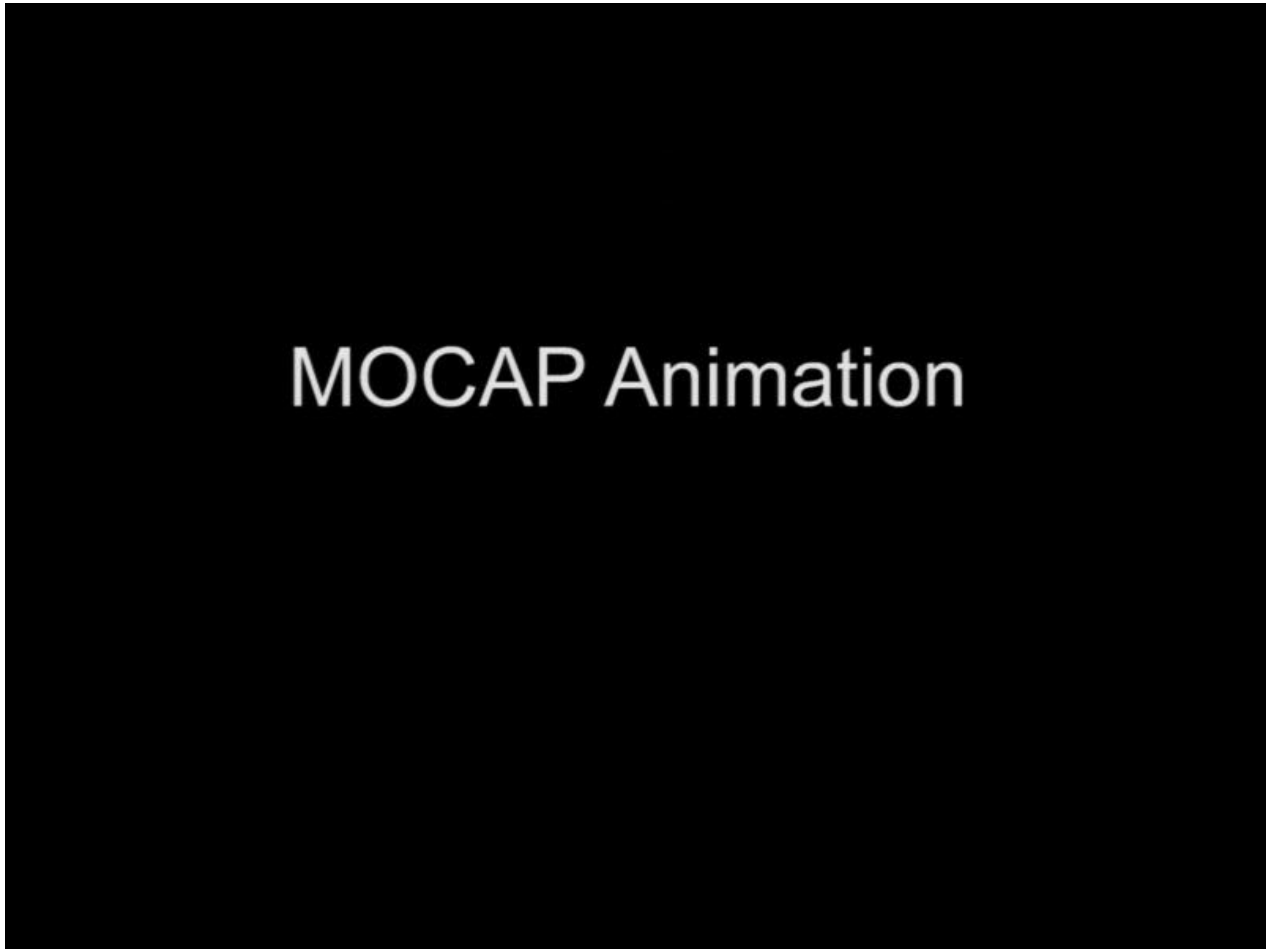
$M = T_0*T_{1.2}*T_{2.2}$

正向运动学例子



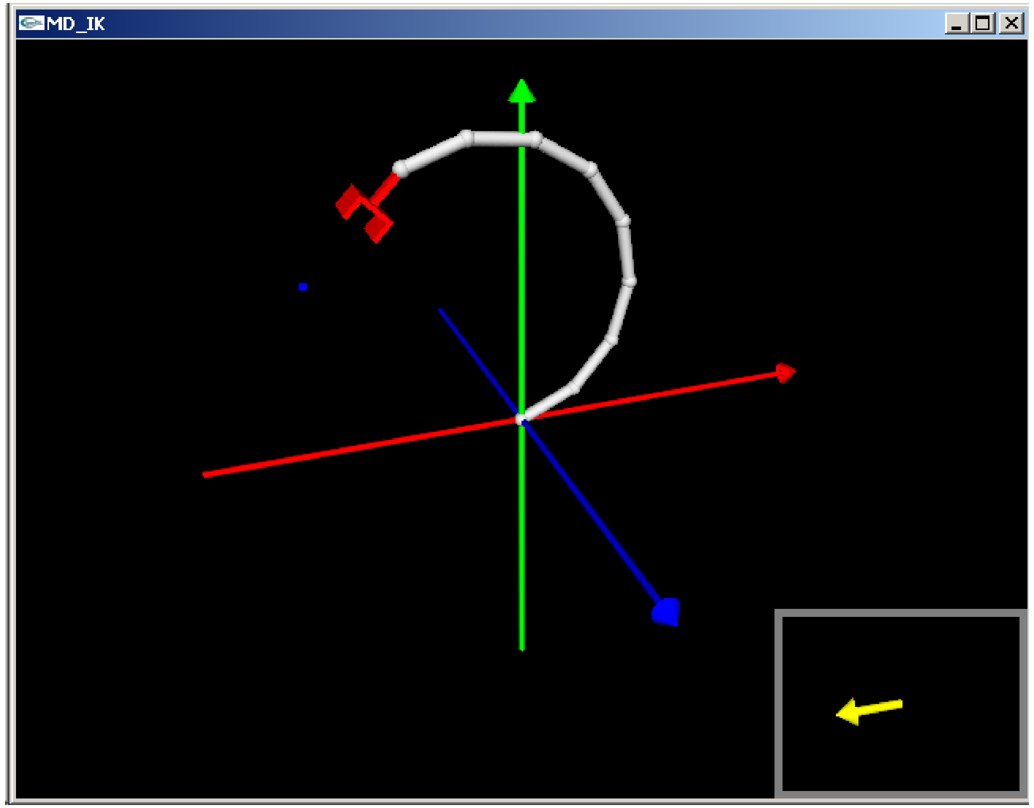
$$x = l_1 \cos(\theta_1) + l_2 \cos(\theta_2) + l_3 \cos(\theta_3)$$
$$y = l_1 \sin(\theta_1) - l_2 \sin(\theta_2) + l_3 \sin(\theta_3)$$

DEMO



逆向运动学(Inverse Kinematics)

- 给定初始姿态向量和目标姿态向量，**计算关节向量的值**，使得物体满足所需的姿势
  - 可以有0个、1个或多个解
    - 过约束系统(Over-constrained system)
    - 欠约束系统(Under-constrained system)
  - 奇异问题



Inverse and Forward kinematics demo application

http://www.fit.vutbr.cz/~dobsik/projects/kinem_INV/kinem_INV.html

逆向运动学

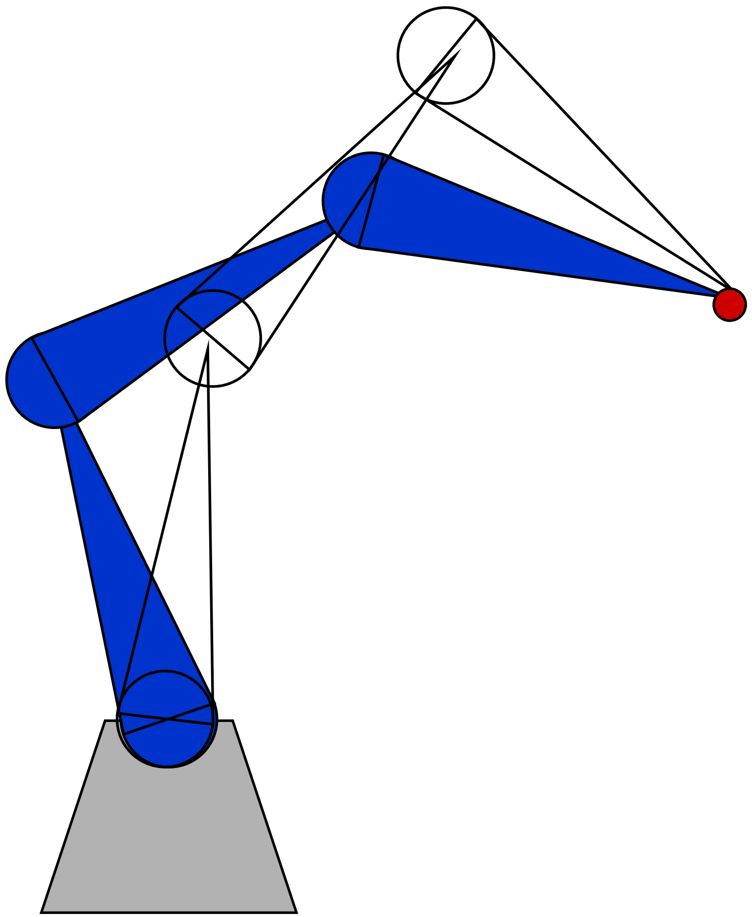
- 例子
  - 2个方程 (约束条件)
  - 3个未知数

$$x = l_1 \cos(\theta_1) + l_2 \cos(\theta_2) + l_3 \cos(\theta_3)$$
$$y = l_1 \sin(\theta_1) - l_2 \sin(\theta_2) + l_3 \sin(\theta_3)$$

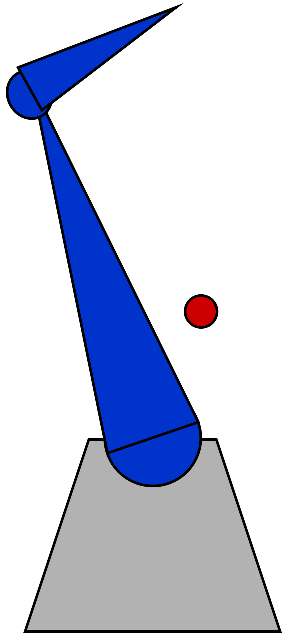
  - 有无穷多个解!
- 这种情况常常遇到!
  - 在保持你的手指碰到鼻子时，可以移动你的肘

IK中的其它问题

- 有无穷多解！



- 无解



逆向运动学

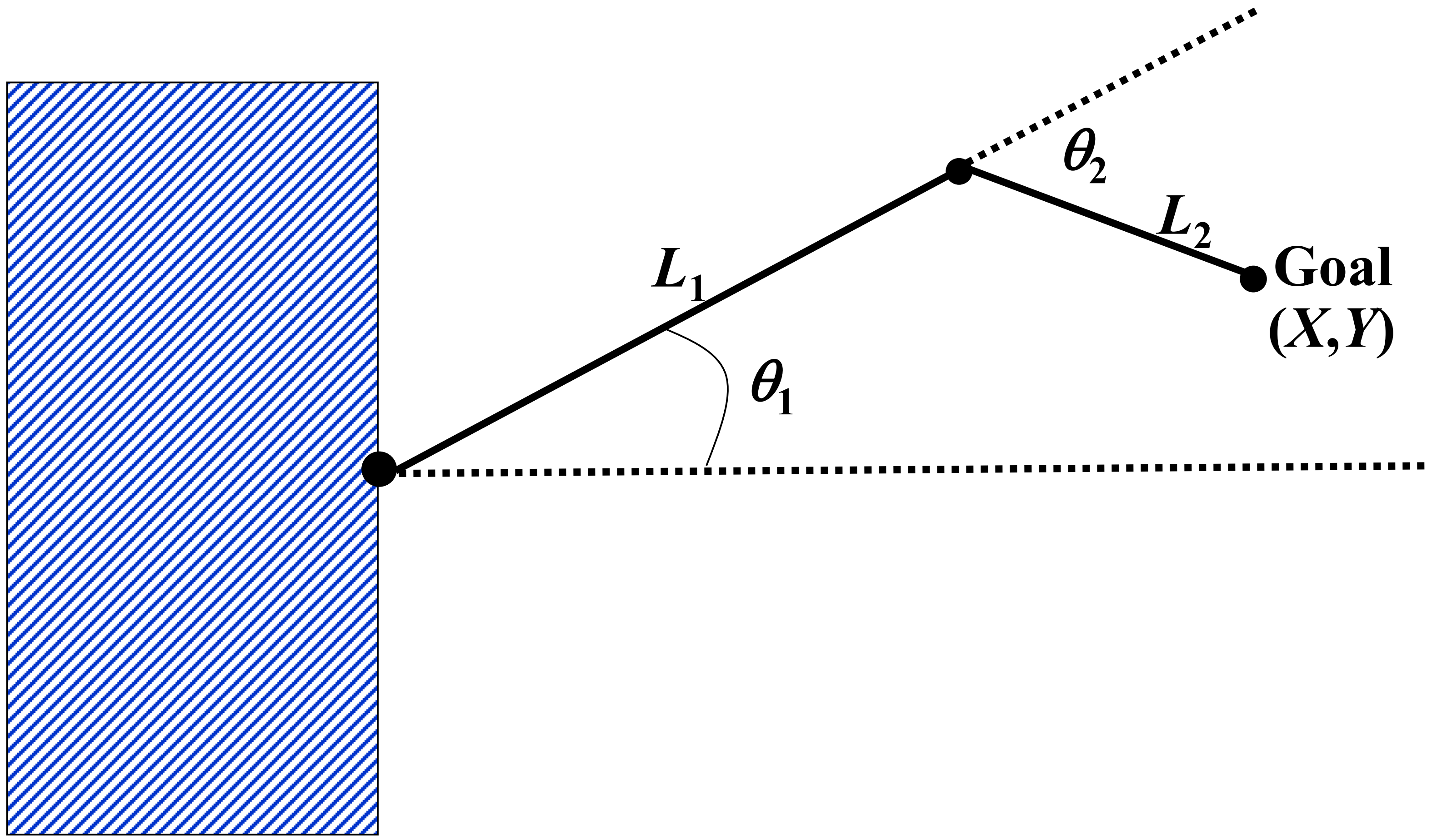
- 一旦得到关节向量值后，我们可对角色的初始姿势和最终姿势的**关节向量值**进行插值，从而得到角色的动画。
  - 缺点：如果初始和最终姿势向量有很大差异时，动画效果可能并不好。
  - 解决办法：
    - 插值姿态向量
    - 对每个得到的插值姿态向量进行IK计算，得到中间某一时刻的关节向量值

逆向运动学
解析法 vs. 数值计算

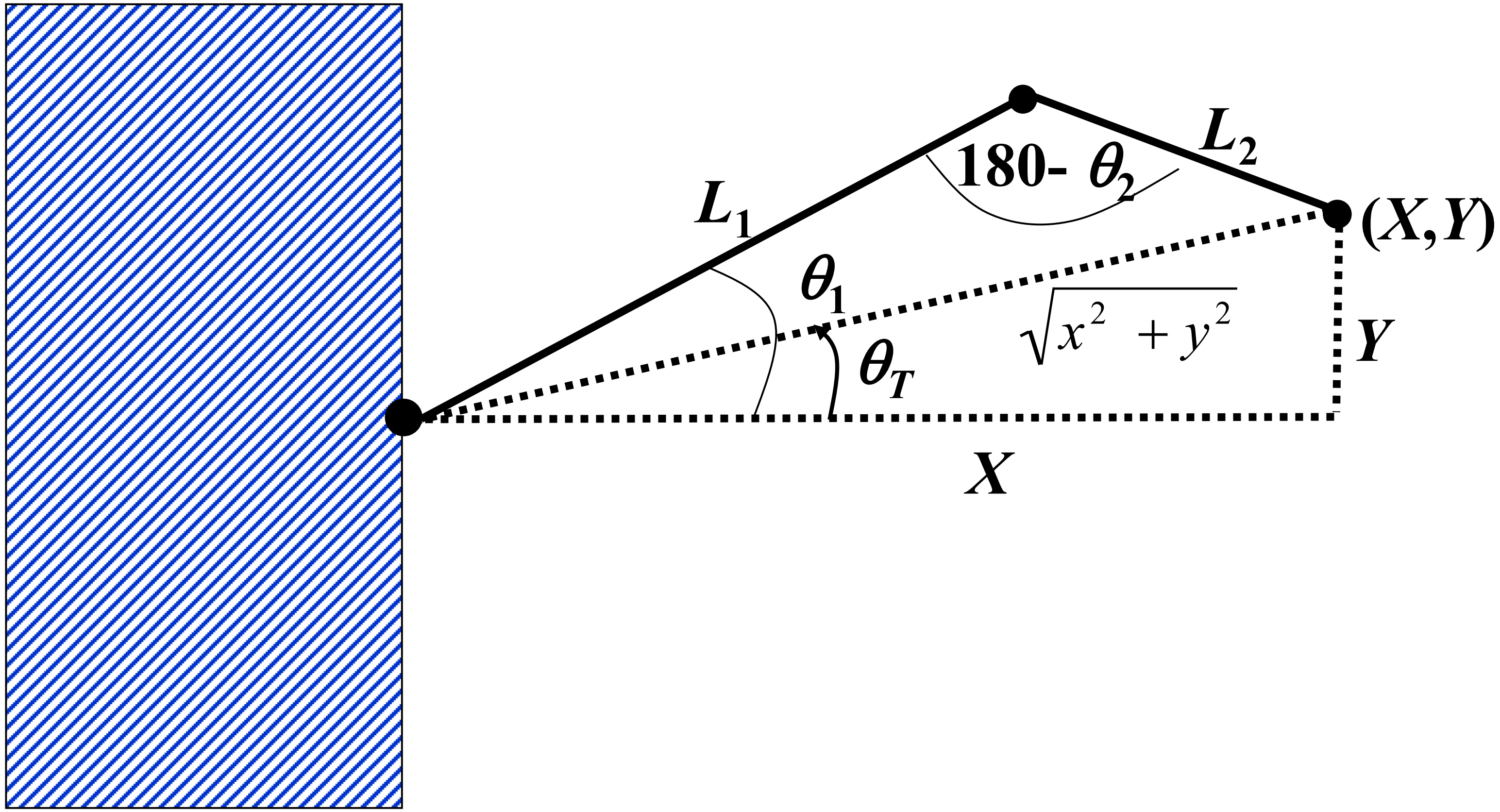
- 只有当关节链非常简单时，才存在解析解

- 一般可用数值增量法求解
  - 逆向雅可比方法(Inverse Jacobian method)
  - 其它方法

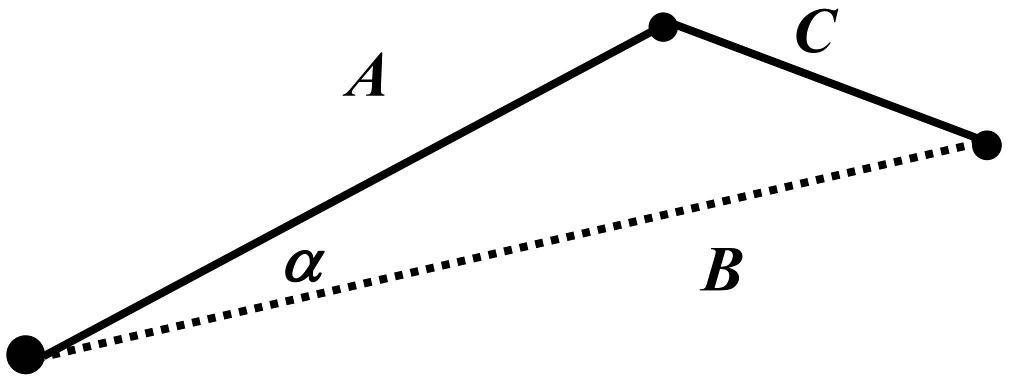
逆向运动学——解析求解法



逆向运动学——解析求解法

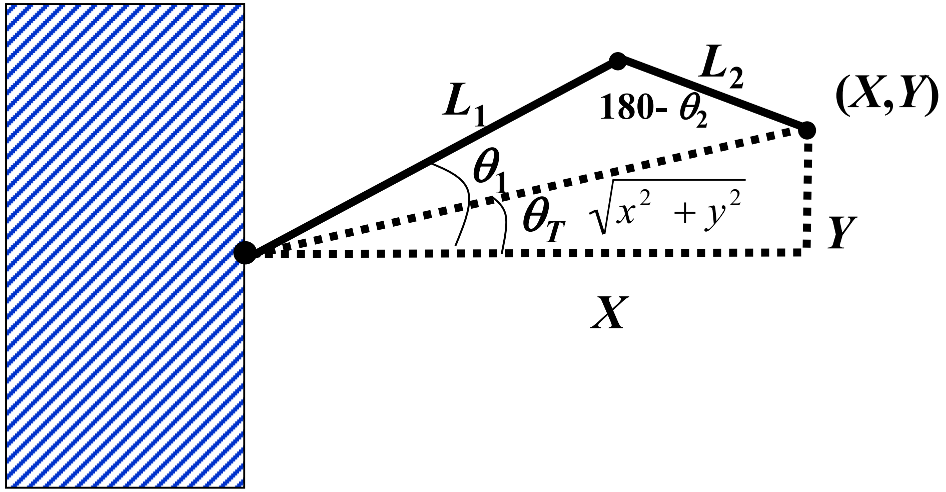


逆向运动学——解析求解法



$$\cos(\alpha) = \frac{A^2 + B^2 - C^2}{2AB}$$

逆向运动学——解析求解法



$$\cos(\theta_T) = \frac{X}{\sqrt{X^2+Y^2}}$$

$$\theta_T = \cos^{-1}\left(\frac{X}{\sqrt{X^2+Y^2}}\right)$$

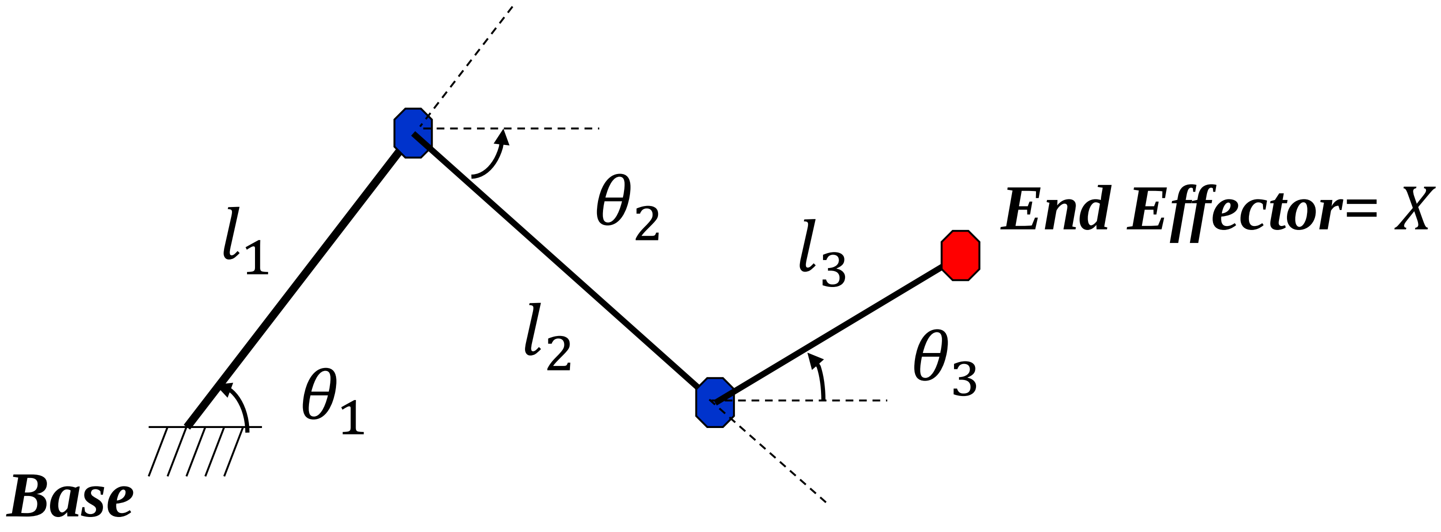
$$\cos(180-\theta_2) = \frac{{L_1}^2+{L_2}^2-(X^2+Y^2)}{2L_1L_2}$$

$$\cos(\theta_1-\theta_T) = \frac{{L_1}^2+X^2+Y^2-{L_2}^2}{2L_1\sqrt{X^2+Y^2}}$$

$$\theta_2 = 180-\cos^{-1}\left(\frac{{L_1}^2+{L_2}^2-(X^2+Y^2)}{2L_1L_2}\right)$$

$$\theta_1 = \cos^{-1}\left(\frac{{L_1}^2+X^2+Y^2-{L_2}^2}{2L_1\sqrt{X^2+Y^2}}\right)+\theta_T$$

逆向运动学——更复杂的关节



$$\theta_1, \theta_2, \theta_3 = f^{-1}(X)$$

为什么求解IK是困难的?

- 冗余(Redundancy)
- 要求自然的运动控制
  - 关节限制
  - 最小的抖动(minimum jerk)
  - 运动方式
- 奇异问题(Singularities)
  - 病态方程(ill-conditioned)
  - 奇异方程Singular

数值求解IK

- **逆向雅克比方法**(Inverse-Jacobian method)
- **循环坐标下降法**(Cyclic Coordinate Descent (CCD))
- **FABRIK法**(Forward And Backward Reaching Inverse Kinematics)
- **基于优化的方法**(Optimization-based method)
- **基于样例的方法**(Example-based method)

逆向雅克比方法

$$X(t) = f(\theta(t))$$

$$X \in R^n \quad (\text{通常} n = 6)$$

$$\theta \in R^m (m = \text{自由度})$$

- 雅克比矩阵为$n \times m$的矩阵，它把θ的微分($d\theta$)与X的微分相关联 (dX)

$$\frac{dX}{dt} = J(\theta)\frac{d\theta}{dt} \qquad \text{其中} J_{ij} = \frac{\partial f_i}{\partial \theta_j}$$

- 所以雅克比矩阵把**关节空间的速度**映射到**笛卡尔空间的速度**

$$V = J(\theta)\dot{\theta}$$

逆向雅克比方法

- 逆向雅克比问题为： $\theta = f^{-1}(X)$
  - f 是一个高度非线性函数

- 通过把雅克比矩阵求逆，把该问题在当前位置局部线性化

$$\dot{\theta} = J^{-1}V$$

  - 通过一系列增量步骤，迭代到所需要的位置

逆向雅克比方法

- 给定初始姿势和所需要的姿势，**迭代**变化关节角，使得末端影响器朝目标位置和方向前进
  - 对于中间帧，插值得到所需的姿态向量
    - 对于每一步k, 通过上述公式得到关节的角速度 $\dot{\theta}$ ，然后执行

$$\theta_{k+1} = \theta_k + \Delta t \dot{\theta}$$

逆向雅克比方法——迭代法

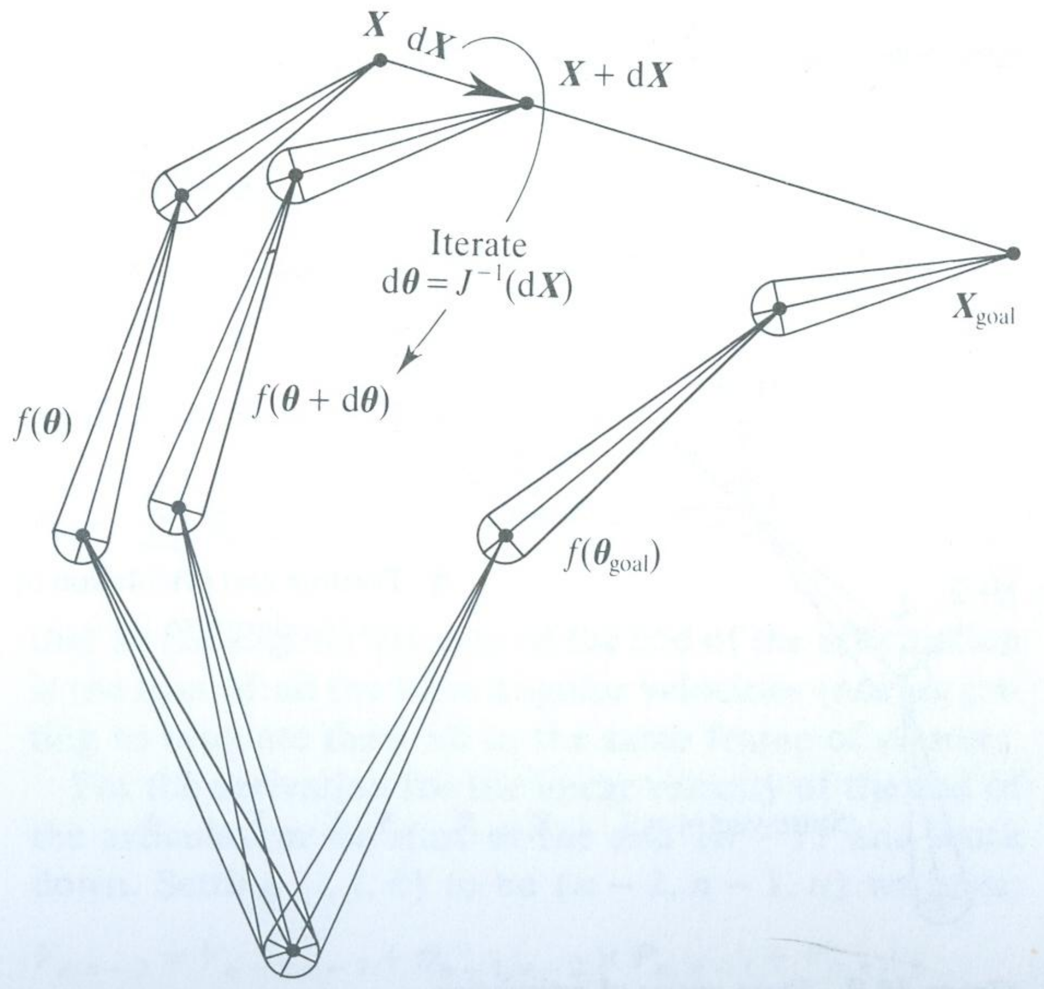


Figure 16.7 One iteration step towards the goal.

逆向雅克比方法——迭代法

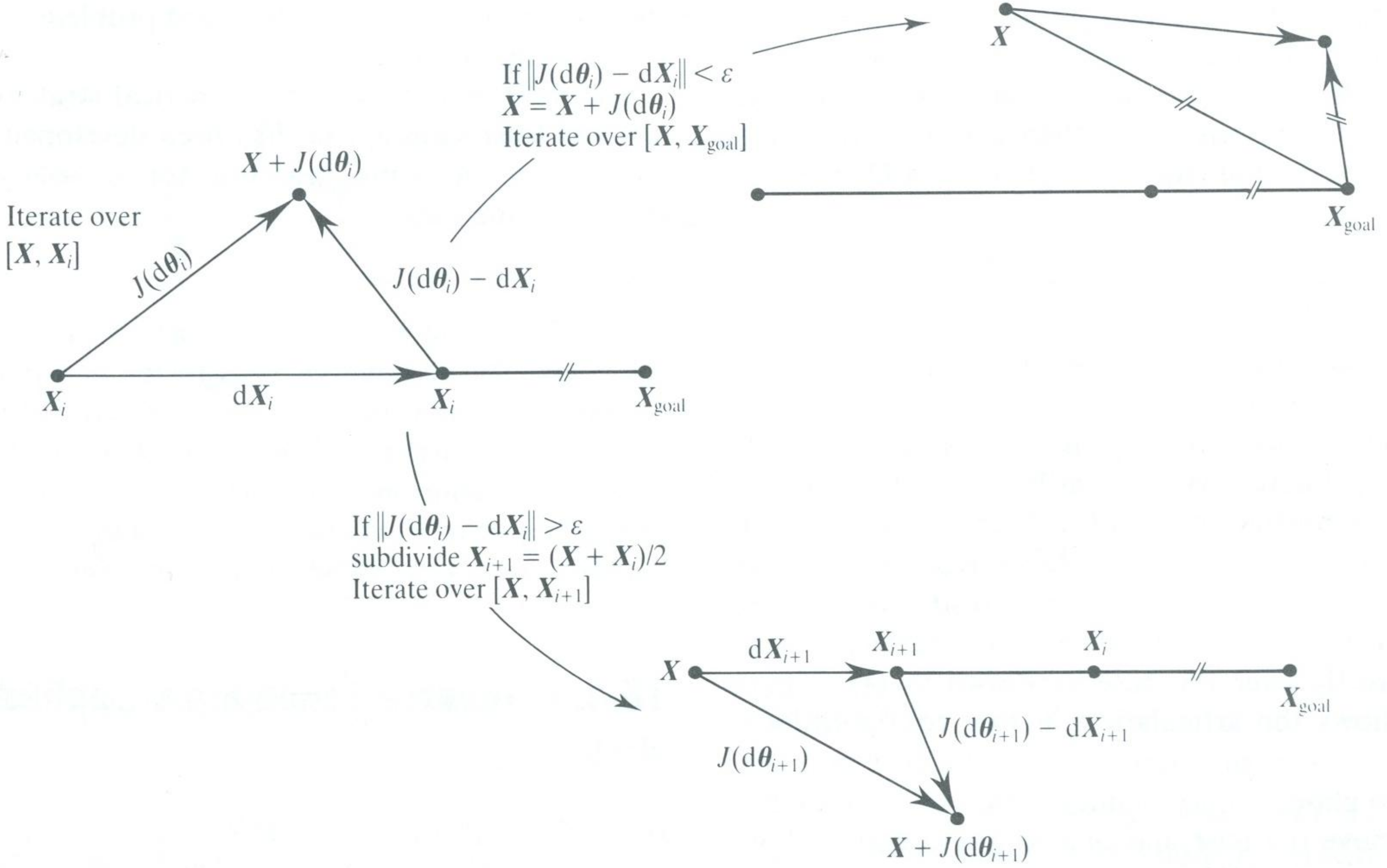
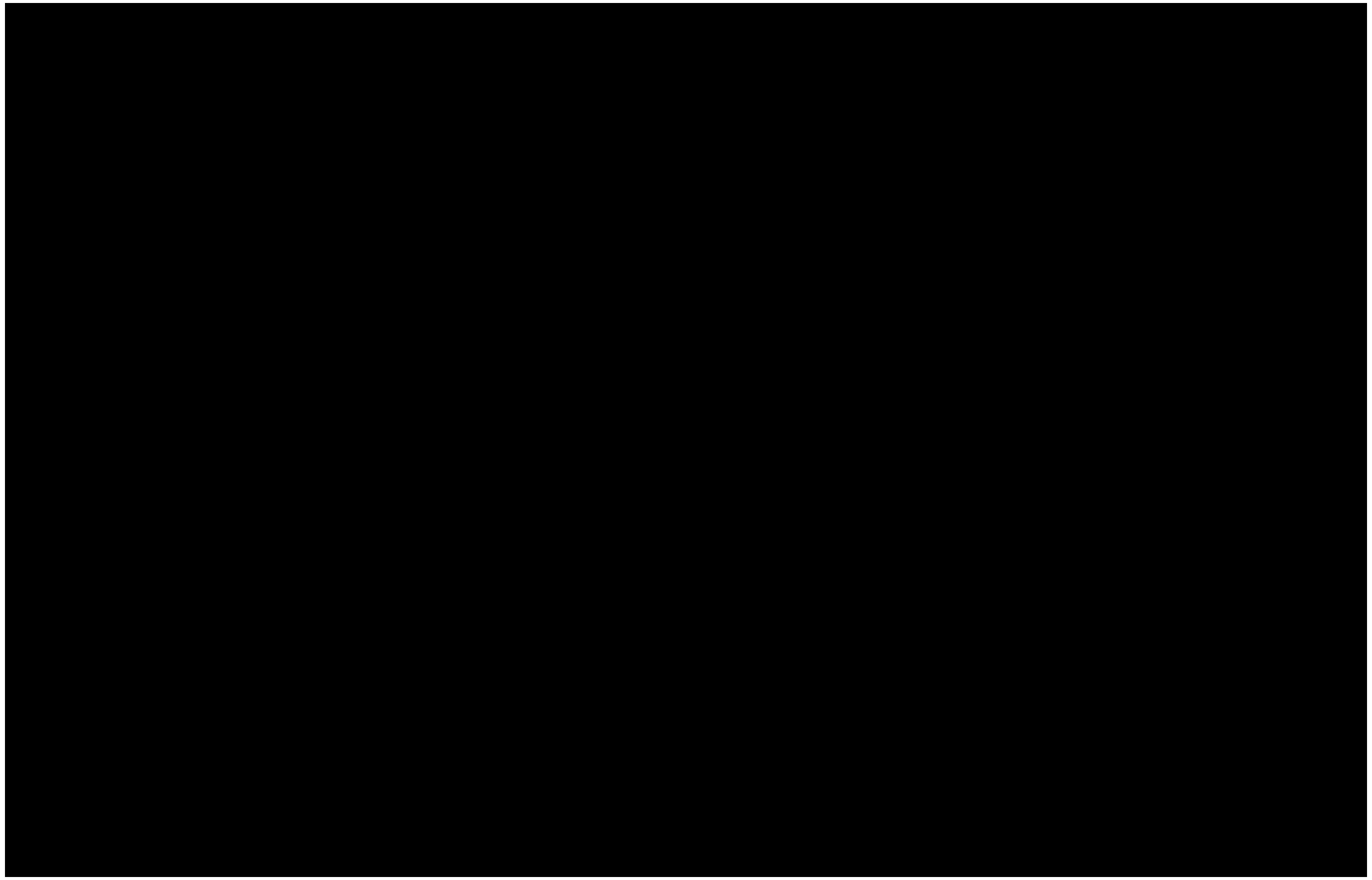


Figure 16.11 Minimizing tracking error.

逆向雅克比方法——DEMO



https://github.com/kzahedi/Robotik/blob/master/Inverse%20Kinematics%20-%20Jacobian.ipynb
https://github.com/myindrata/Pseudo-Inverse-Jacobian-Inverse-Kinematics?tab=readme-ov-file

逆向雅克比方法

- 雅克比矩阵把**关节空间的速度**映射到**笛卡尔空间的速度。**

$$X(t) = f(\theta(t)), \qquad V = J\dot{\theta}$$

- 逆向雅克比矩阵把**笛卡尔空间的速度**映射到**关节空间的速度。**

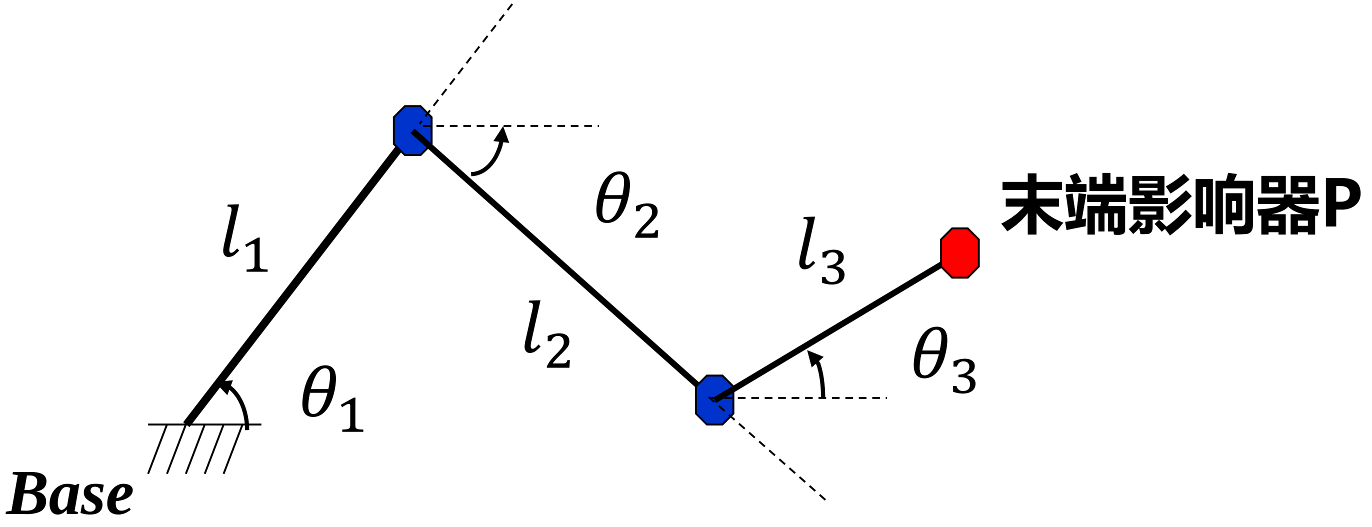
$$\dot{\theta} = J^{-1}V, \qquad d\theta = J^{-1}dX$$

剩下的问题：如何计算雅克比矩阵？

- 解析计算
  - 对于简单关节可行
- 几何方法
  - 适合于复杂关节
  - 比如说，我们现在只关心末端影响器的位置e
    - 这时，雅克比矩阵为$3\times N$矩阵，其中N为自由度的数目
    - 对于每个自由度，我们分析e如何随自由度变化

$$d\theta = J^{-1} dX$$

解析计算雅克比矩阵——一个例子



解析计算雅克比矩阵——一个例子

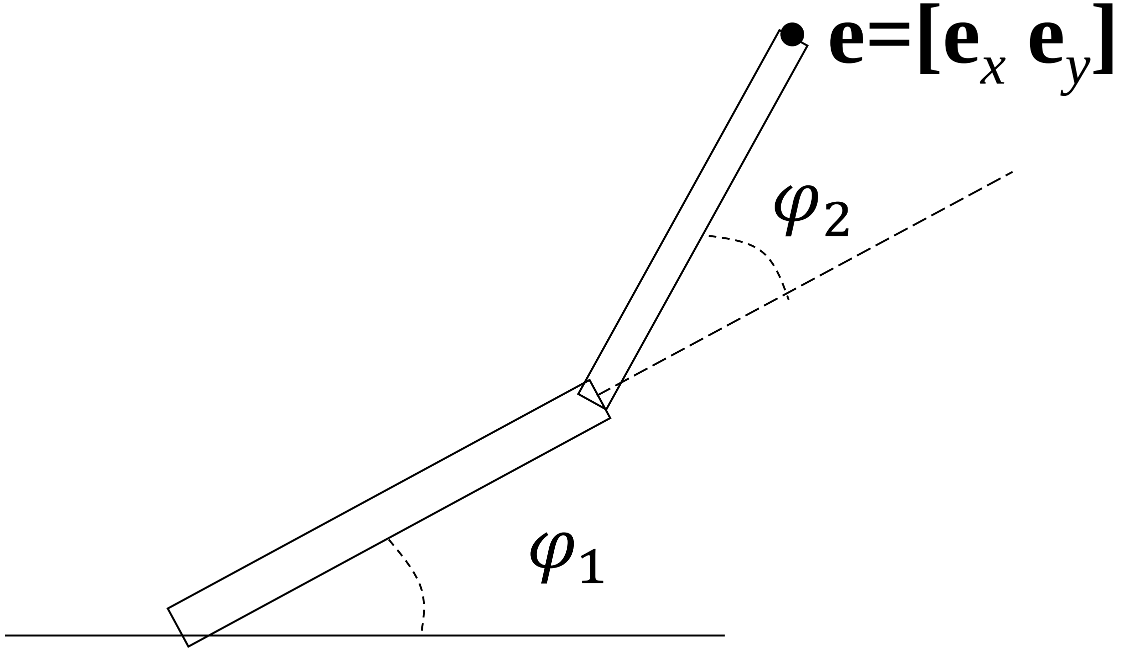
$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f_1(\boldsymbol{\theta}) \\ f_2(\boldsymbol{\theta}) \end{bmatrix} = \begin{bmatrix} l_1 \cos\theta_1 + l_2 \cos\theta_2 + l_3 \cos\theta_3 \\ l_1 \sin\theta_1 - l_2 \sin\theta_2 + l_3 \sin\theta_3 \end{bmatrix}$$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \mathbf{J} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix}$$

$$\mathbf{J} = \begin{bmatrix} \frac{\partial f_1(\boldsymbol{\theta})}{\partial \theta_1} & \frac{\partial f_1(\boldsymbol{\theta})}{\partial \theta_2} & \frac{\partial f_1(\boldsymbol{\theta})}{\partial \theta_3} \\ \frac{\partial f_2(\boldsymbol{\theta})}{\partial \theta_1} & \frac{\partial f_2(\boldsymbol{\theta})}{\partial \theta_2} & \frac{\partial f_2(\boldsymbol{\theta})}{\partial \theta_3} \end{bmatrix} = \begin{bmatrix} -l_1 \sin\theta_1 & -l_2 \sin\theta_2 & -l_3 \sin\theta_3 \\ l_1 \cos\theta_1 & -l_2 \cos\theta_2 & l_3 \cos\theta_3 \end{bmatrix}$$

雅克比矩阵计算的几何法——一段二维手臂

- 假设我们有一段简单的二维机器人手臂，包含两个自由度为1的旋转关节:



雅克比矩阵计算的几何法——一段二维手臂

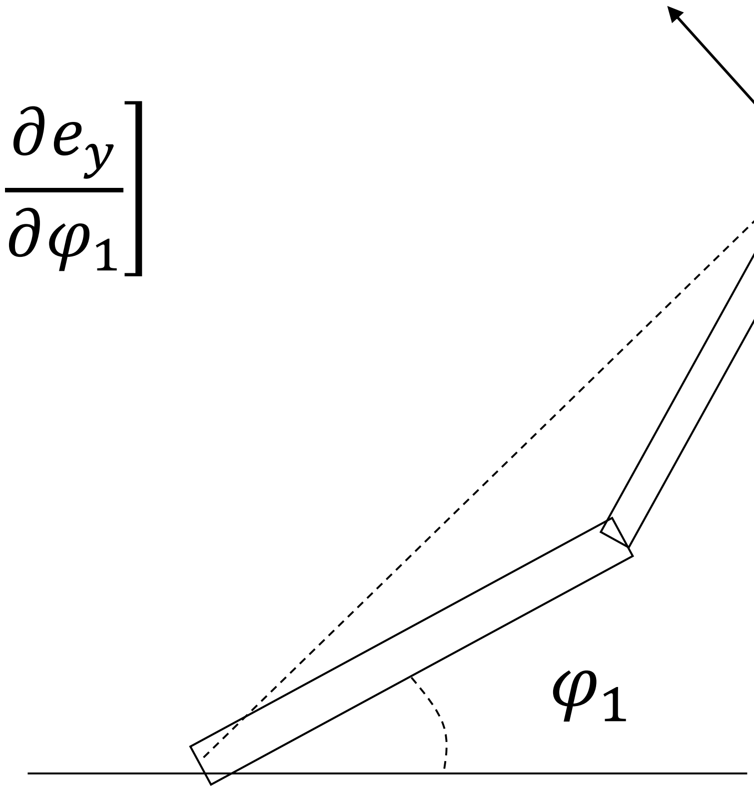
- 雅克比矩阵$J(\mathbf{\Phi})$显示了e的每个分量如何随每个关节角变化

$$J(\mathbf{\Phi}) = \begin{bmatrix} \dfrac{\partial e_x}{\partial \varphi_1} & \dfrac{\partial e_x}{\partial \varphi_2} \\ \dfrac{\partial e_y}{\partial \varphi_1} & \dfrac{\partial e_y}{\partial \varphi_2} \end{bmatrix}$$

雅克比矩阵计算的几何法——一段二维手臂

- 考虑如果把φ_1 少量增加一点，e 会发生什么变化?

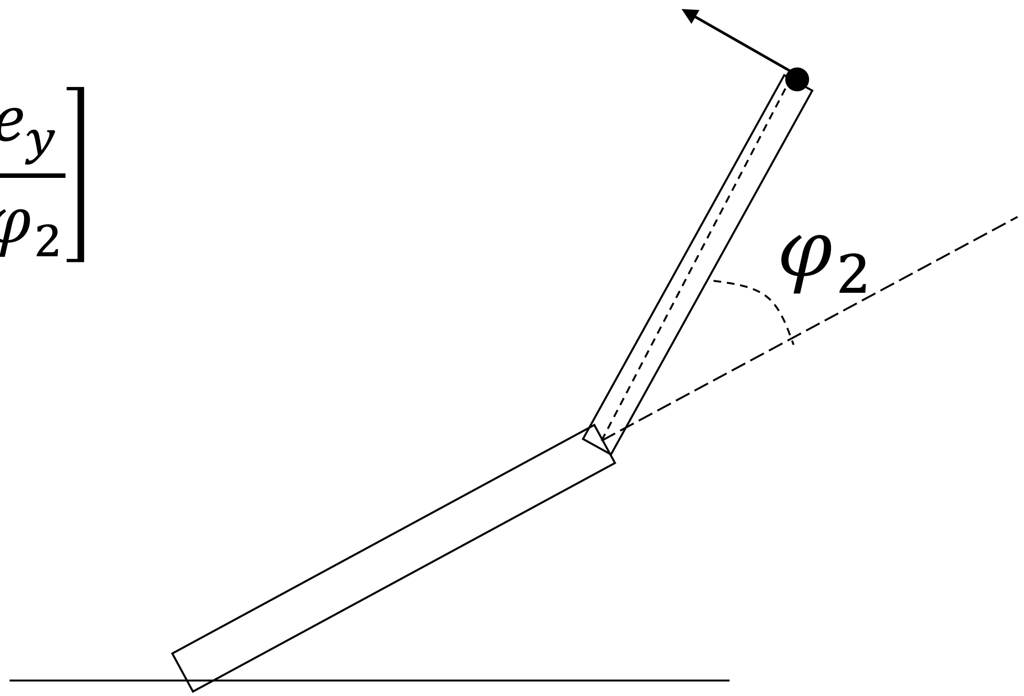
$$\frac{\partial \mathbf{e}}{\partial \varphi_1} = \begin{bmatrix} \frac{\partial e_x}{\partial \varphi_1} & \frac{\partial e_y}{\partial \varphi_1} \end{bmatrix}$$



雅克比矩阵计算的几何法——一段二维手臂

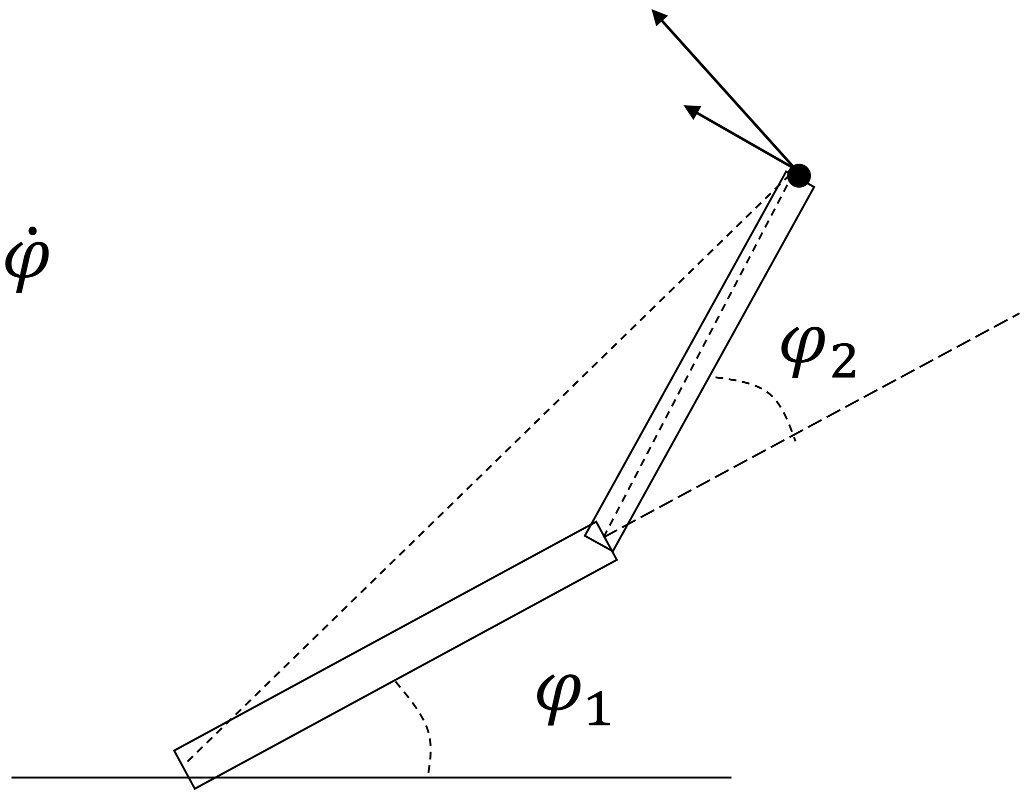
- 同理，如果把φ_2少量增加一点，e 会发生什么变化?

$$\frac{\partial \mathbf{e}}{\partial \varphi_2} = \begin{bmatrix} \frac{\partial e_x}{\partial \varphi_2} & \frac{\partial e_y}{\partial \varphi_2} \end{bmatrix}$$

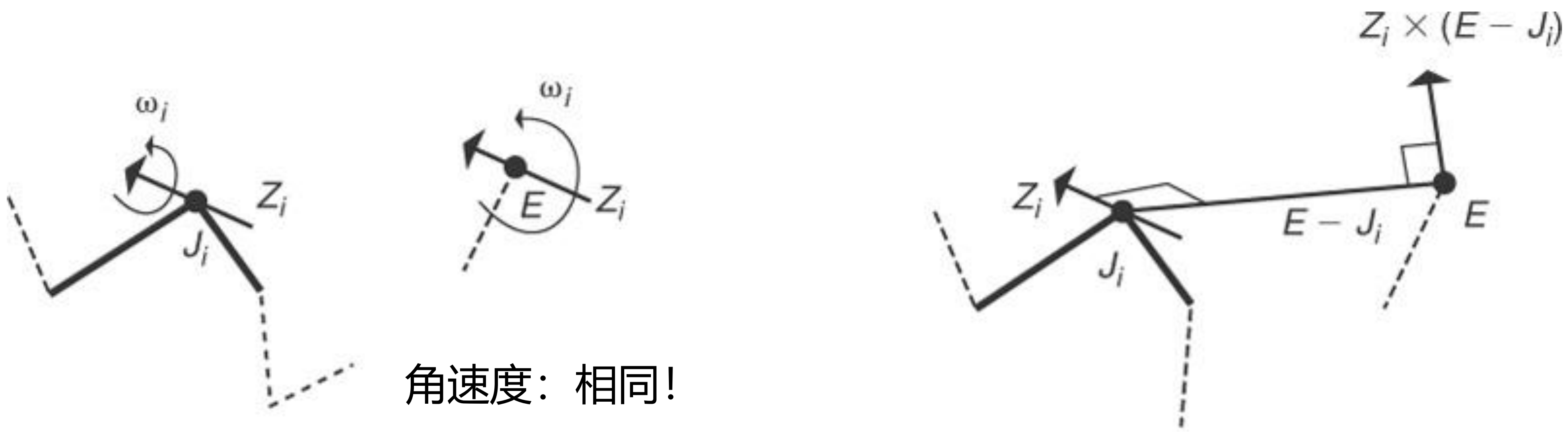


雅克比矩阵计算的几何法——一段二维手臂

$$V = J(\mathbf{\Phi})\dot{\varphi} = \begin{bmatrix} \dfrac{\partial e_x}{\partial \varphi_1} & \dfrac{\partial e_x}{\partial \varphi_2} \\ \dfrac{\partial e_y}{\partial \varphi_1} & \dfrac{\partial e_y}{\partial \varphi_2} \end{bmatrix} \dot{\varphi}$$



雅克比矩阵计算的几何法——一段二维手臂



角速度：相同！

a) Angular velocity, ω_i

b) Linear velocity, $Z_i \times (E - J_i)$

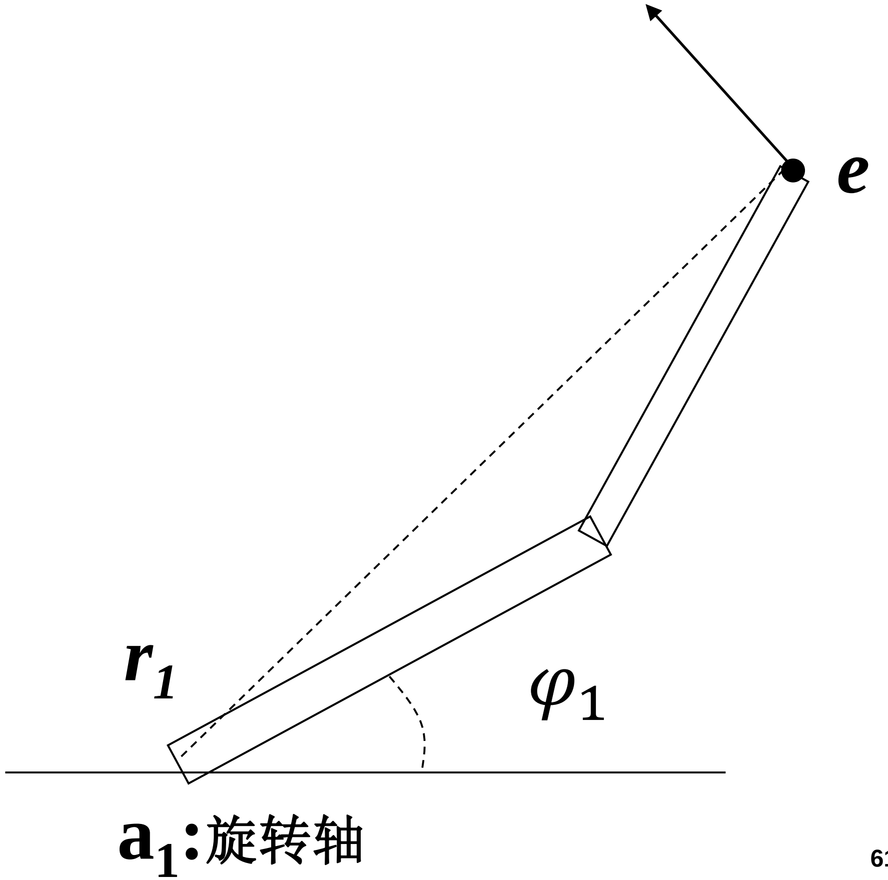
E — end effector
J_i — i th joint
Z_i — i th joint axis
ω_i — angular velocity of i th joint

线速度：两个矢量的叉乘

由关节轴旋转引起的角速度和线速度

雅克比矩阵计算的几何法——一段二维手臂

- 让我们首先考虑1个旋转自由度关节的问题
- 我们想知道如果绕该关节的轴旋转，整体位置e如何变化?

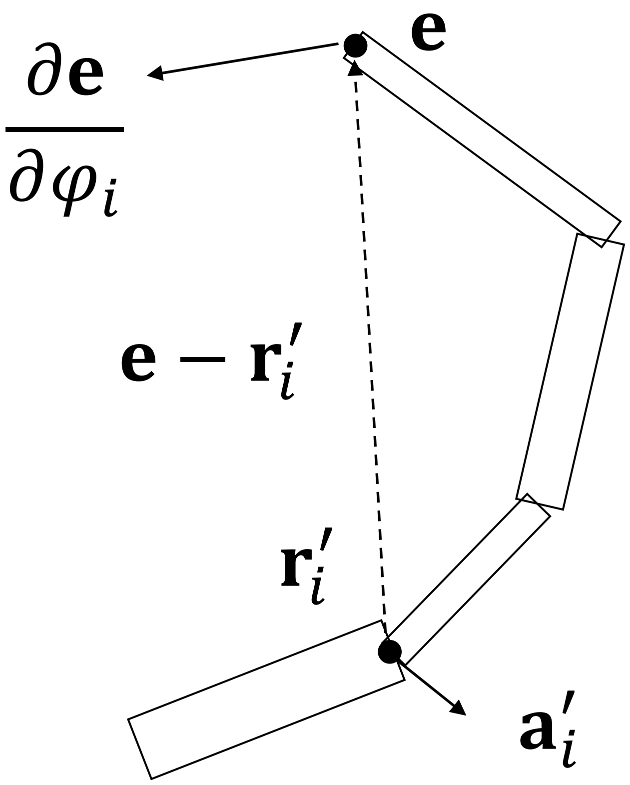


$$\frac{\partial \mathbf{e}}{\partial \varphi_1} = \begin{bmatrix} \frac{\partial e_x}{\partial \varphi_1} & \frac{\partial e_y}{\partial \varphi_1} \end{bmatrix}$$

$$= \mathbf{a}_1 \times (\mathbf{e} - \mathbf{r}_1)$$

雅克比矩阵计算的几何法——一段二维手臂

$$\frac{\partial \mathbf{e}}{\partial \varphi_i} = \mathbf{a}_i' \times (\mathbf{e} - \mathbf{r}_i')$$



a'$_i$: 第i个关节在世界坐标系中的单位旋转轴
r'$_i$: 第i个关节在世界坐标系中旋转中心
e: 末端影响器在世界坐标系中位置

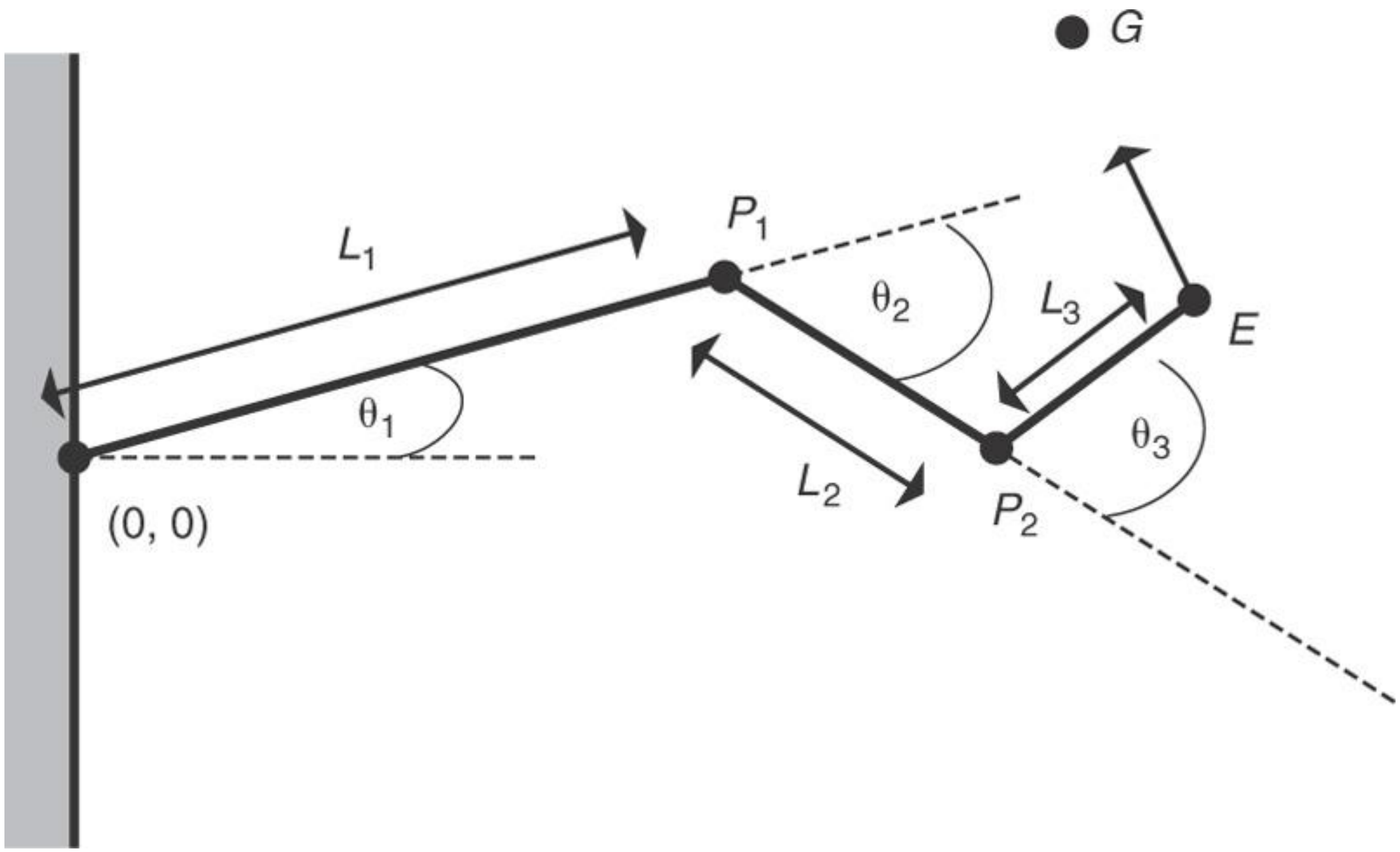
雅克比矩阵计算的几何法——3自由度旋转关节

- 一旦我们有了关节在世界坐标系的每个轴，我们就得到了雅克比矩阵的一列
- 我们把3自由度旋转关节**当成3个1自由度的关节**，以便可以用相同的公式来计算导数：

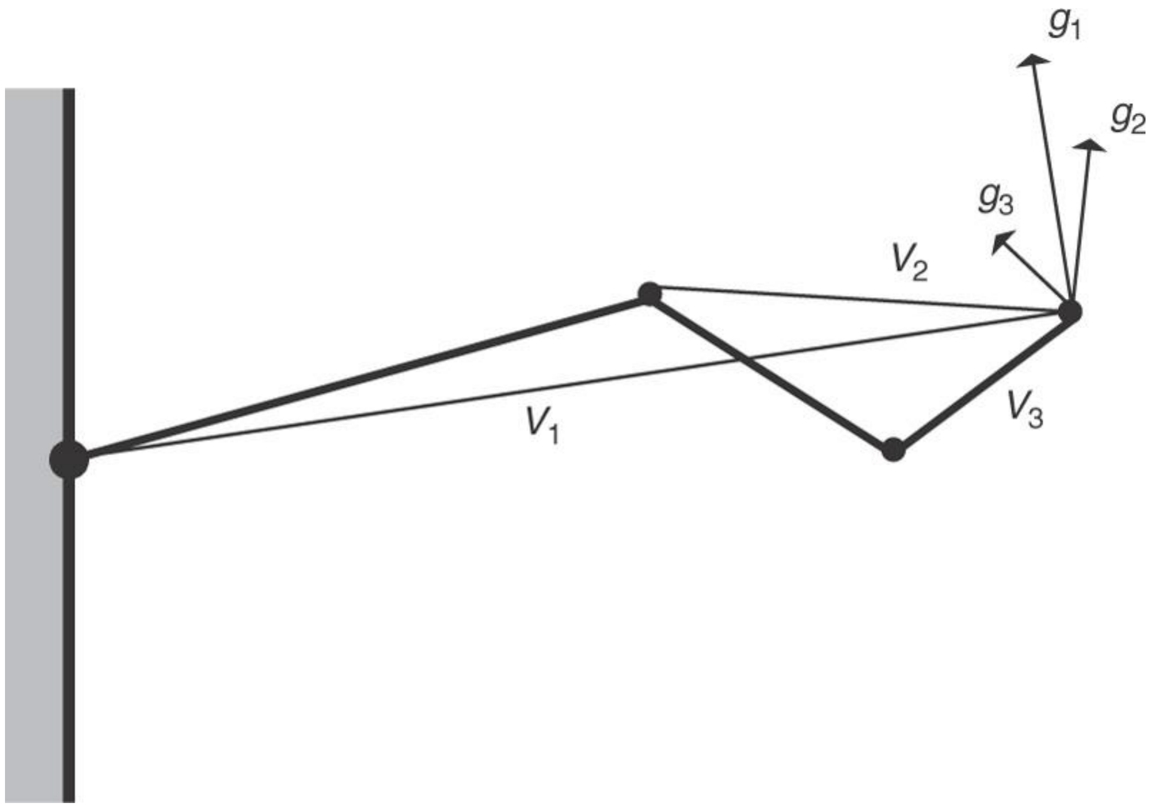
$$\frac{\partial \mathbf{e}}{\partial \varphi_i} = \mathbf{a}_i' \times (\mathbf{e} - \mathbf{r}_i')$$

  - 对三个轴的每个轴重复应用上述公式

雅克比矩阵计算的几何法——另一段二维手臂



雅克比矩阵计算的几何法——另一段二维手臂



- 问题变成：计算由各个关节引起的速度的线性组合，从而得到末端影响器的速度。

雅克比矩阵计算的几何法——另一段二维手臂

$$V = \begin{bmatrix} (G-E)_x \\ (G-E)_y \\ (G-E)_z \end{bmatrix} = J(\mathbf{\Phi})\dot{\varphi} = \begin{bmatrix} \dfrac{\partial e_x}{\partial \varphi_1} & \dfrac{\partial e_x}{\partial \varphi_2} & \dfrac{\partial e_x}{\partial \varphi_3} \\ \dfrac{\partial e_y}{\partial \varphi_1} & \dfrac{\partial e_y}{\partial \varphi_2} & \dfrac{\partial e_y}{\partial \varphi_3} \end{bmatrix} \dot{\varphi}$$

$$= \begin{bmatrix} ((0,0,1) \times E)_x & ((0,0,1) \times (E-P_1))_x & ((0,0,1) \times (E-P_2))_x \\ ((0,0,1) \times E)_y & ((0,0,1) \times (E-P_1))_y & ((0,0,1) \times (E-P_2))_y \\ ((0,0,1) \times E)_z & ((0,0,1) \times (E-P_1))_z & ((0,0,1) \times (E-P_2))_z \end{bmatrix} \dot{\varphi}$$

逆向雅克比计算方法

$$X = f(\theta)$$

$$\theta = f^{-1}(X)$$

$$V = J(\theta)\dot{\theta}$$

$$\dot{\theta} = J^{-1}(\theta)V$$

雅克比矩阵依赖于当前的关节布局

(configuration)。需要逐帧计算！

逆向雅克比方法中的问题

- 在局部化 $X = f(\theta)$中的问题（**误差**）
- 在求解逆矩阵中的问题
  - **非方阵问题**
    - 用伪逆！
  - **奇异问题**
    - 如果雅克比矩阵的逆不存在
    - 当用任何 $\dot{\theta}$ 也无法得到所需的 V
  - **接近奇异问题**
    - 病态矩阵

逆向雅克比方法
——雅克比矩阵的伪逆的计算

$V = J\dot{\theta}$

$J^T V = J^T J\dot{\theta}$

当J满行秩时，$(J^T J)^{-1}$存在！

$(J^T J)^{-1} J^T V = (J^T J)^{-1} J^T J\dot{\theta}$

$J^+ V = \dot{\theta}$

得到 J 的伪逆：

$J^+ = (J^T J)^{-1} J^T = J^T (JJ^T)^{-1}$

伪逆本质上为最小二乘！

$J^+ V = \dot{\theta}$

$J^T (JJ^T)^{-1} V = \dot{\theta}$

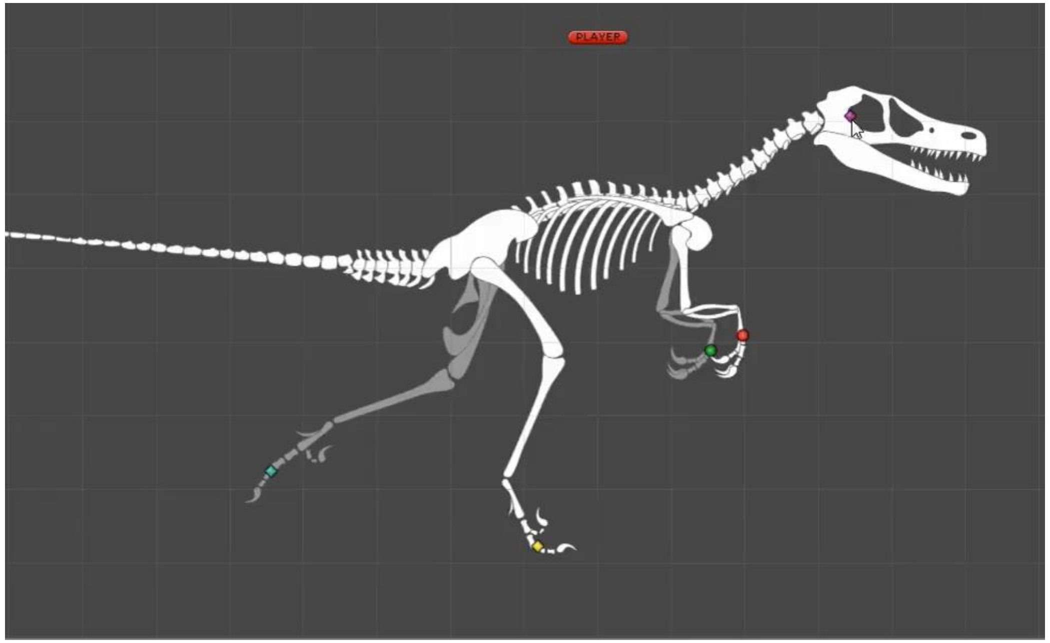
假设 $\beta = (JJ^T)^{-1} V$，它表明 $(JJ^T)\beta = V$，

可以用**LU**分解求解 (求得β).

$J^T \beta = \dot{\theta}$(求得$\dot{\theta}$)

总结

- 关节动画的基本概念
- 逆向雅克比方法的迭代求解
- 雅克比矩阵计算的几何法



The End