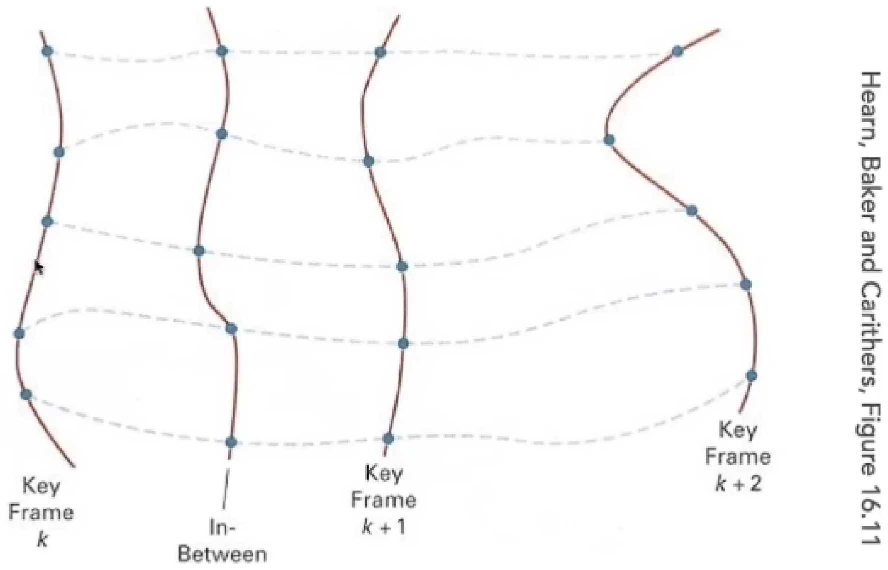
笔记[4]

动画（Animation）

关键帧动画（Keyframe Animation）

- 通过关键帧间插值得到动画



物理模拟（Physical Simulation）

- 牛顿定律

$$F = ma$$

  分别表示力、质量、加速度

- 位置估计：

$$x^{t+\Delta t} = x^t + \Delta t v^t + \frac{1}{2}(\Delta t)^2 a^t$$

- 可用于衣服布料模拟、流体模拟等

质点弹簧系统（Mass Spring System）

- 一系列相互连接的质点和弹簧
- 理想弹簧模型：

$$f_{a \to b} = k_s(b - a)$$
$$f_{b \to a} = -f_{a \to b}$$

基于胡克定律，k_s为弹性系数

- 考虑弹簧长度：

$$f_{a\to b} = k_s \frac{b-a}{\|b-a\|}(\|b-a\|-l)$$

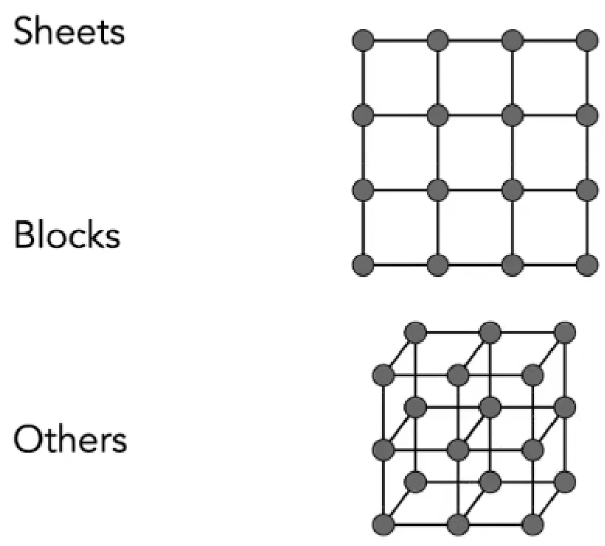
l为弹簧原始长度

- 考虑摩擦力（damping force）：$f = -k_d \dot{b}$，$\dot{b}$表示速度，k_d为摩擦系数
  - 摩擦力与a、b之间的相对运动相关，得到b上的摩擦力表达式为：

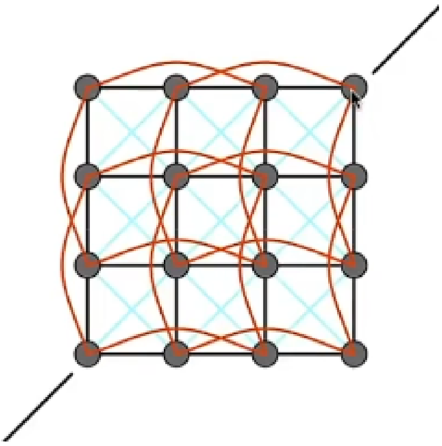
$$f_b = -k_d \left[\frac{b-a}{\|b-a\|} \cdot (\dot{b}-\dot{a})\right] \cdot \frac{b-a}{\|b-a\|}$$

其中，$[]$中的项为标量，表示相对速度向量在a-b位置向量上的投影

- 由弹簧构成的物体：



- 由弹簧构成布料：



有限元方法（Finite Element Method）

- 对力的扩散过程进行建模

粒子系统（Particle System）

- 由很多粒子构成的动力系统，比如雾、灰尘等
- 考虑粒子之间的作用力：吸引力、斥力、粘滞力、电磁力、碰撞等
- 算法过程：
  1. （生成新的粒子）
  2. 计算每个粒子作用力
  3. 根据作用力更新粒子的位置和速度
  4. （移除无效的粒子）
  5. 渲染粒子
- 星系模拟，考虑万有引力：

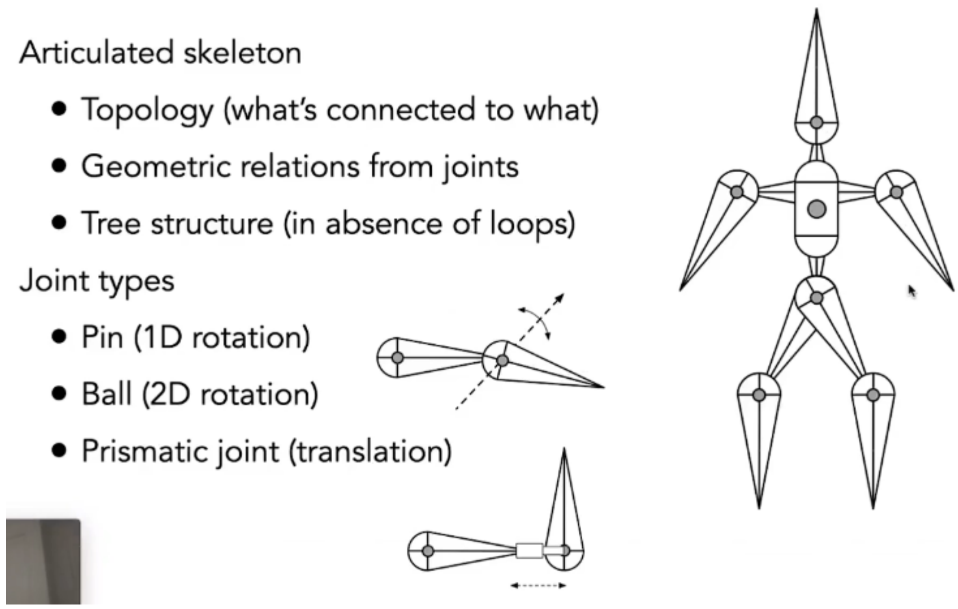
$$F_g = G\frac{m_1 m_2}{d^2},\ G = 6.67428 \times 10^{-11}\mathrm{Nm}^2\mathrm{kg}^{-2}$$

- 可用于模拟鸟群、鱼群、人群、蜜蜂等，考虑引力、斥力、朝向，先定义个体属性，再进行模拟

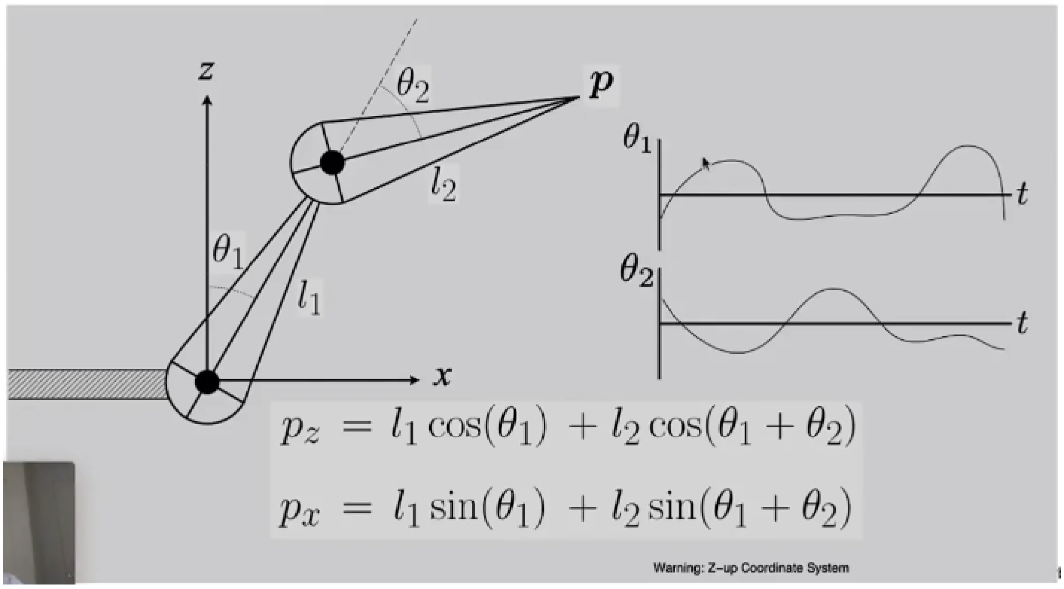
正向运动学（Forward Kinematics）

- 骨骼系统（Skeleton）：包括拓扑结构、几何连接等
- 连接（关节）类型：

1. Pin：单向旋转
2. Ball：双向旋转
3. Prismatic joint：位移

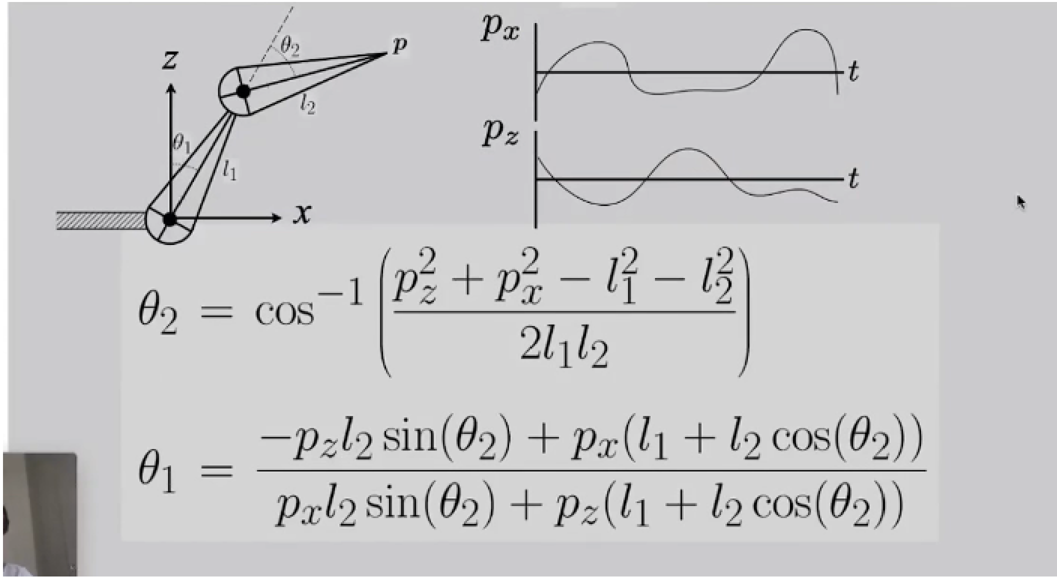


- 计算示例：



逆向运动学（Inverse Kinematics）

- 先指定尖端位置（末端位置），再通过运动学得到运动过程（参数θ_1, θ_2等）
- 计算示例：



- 解不一定唯一，不一定存在
- 采用优化方法进行求解，结合梯度下降法
  1. 初始化参数
  2. 计算误差（比如目标位置与当前位置的平方距离）
  3. 计算误差关于参数的梯度
  4. 应用梯度下降（牛顿法等）

Rigging

- 对形状进行的控制
- 逆运动学应用
- 关键点控制动作
- 形状插值

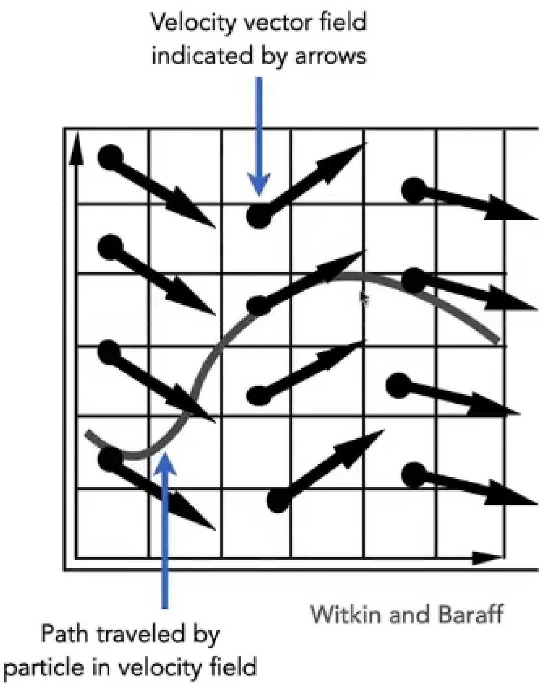
动作捕捉（Motion Capture）

- 捕捉真人动作，映射到虚拟人物上

单粒子模拟（Single Particle Simulation）

- 定义速度场（Velocity Vector Field）

$$v(x,t) = \frac{dx}{dt} = \dot{x}$$



- 已知速度场，求质点位置，即求解一阶常微分方程（first-order ordinary differential equation，ODE）
- 初始位置为x_0

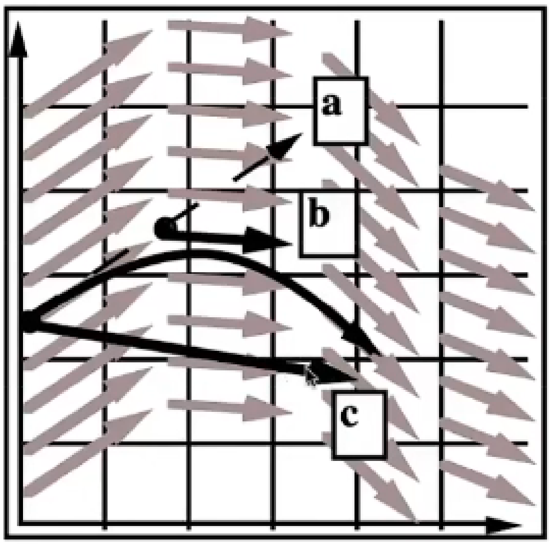
欧拉方法（数值解法）

- 离散化迭代操作：

$$\begin{aligned} x^{t+\Delta t} &= x^t + \Delta t \dot{x}^t \\ \dot{x}^{t+\Delta t} &= \dot{x}^t + \Delta t \ddot{x}^t \end{aligned}$$

- 不稳定，可能发散，误差会叠加
- 中点法（Midpoint Method）
  - 计算欧拉步长(a)
  - 计算欧拉步长中点的速度(b)
  - 使用中点速度更新位置(c)

$$\begin{aligned} x^t_{mid} &= x^t + \Delta t/2 \cdot \dot{x}^t \\ x^{t+\Delta t} &= x^t + \Delta t \cdot \dot{x}^t_{mid} \end{aligned}$$



Witkin and Baraff

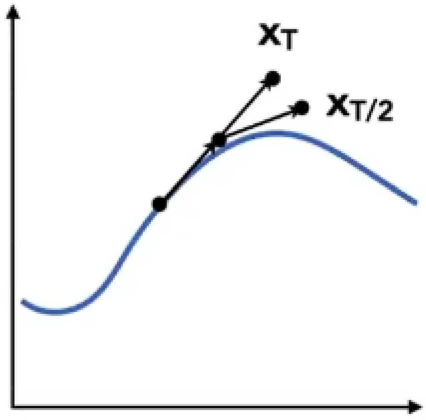
- 推导可得：

$$x^{t+\Delta t} = x^t + \Delta t \dot{x}^t + \frac{(\Delta t)^2}{x} \ddot{x}^t$$

  相较于原本的牛顿法，增加了二阶项

自适应步长法（Adaptive Step Size）

- 不断减小步长，直到误差小于阈值
  - 计算欧拉步长x_T
  - 计算两步欧拉步长$x_{T/2}$
  - 计算误差$\|x_T - x_{T/2}\|$
  - 若误差足够小则停止迭代，否则减小步长T



隐式欧拉法（Implicit Euler Method）

- 使用下一步的速度和加速度更新位置：

$$
\begin{aligned}
x^{t+\Delta t} &= x^t + \Delta t \dot{x}^{t+\Delta t} \\
\dot{x}^{t+\Delta t} &= \dot{x}^t + \Delta t \ddot{x}^{t+\Delta t}
\end{aligned}
$$

- 稳定性更好
  - 局部误差为$O(h^2)$
  - 全局误差为$O(h)$，h为步长
  - 阶数越高越好，说明随着步长减小，误差减小越快

Runge-Kutta方法族

- 四阶方法（RK4）
  - 初始设定：

$$
\frac{dy}{dt} = f(t, y), \quad y(t_0) = y0
$$

  - 解法：

$$
\begin{aligned}
y_{n+1} &= y_n + \frac{1}{6}h(k_1 + 2k_2 + 2k_3 + k_4) \\
t_{n+1} &= t_n + h
\end{aligned}
$$

其中：

$$
\begin{aligned}
k_1 &= f(t_n, y_n) \\
k_2 &= f\left(t_n + \frac{h}{2},\ y_n + h\frac{k_1}{2}\right) \\
k_3 &= f\left(t_n + \frac{h}{2},\ y_n + h\frac{k_2}{2}\right) \\
k_4 &= f\left(t_n + h,\ y_n + hk_3\right)
\end{aligned}
$$

Verlet Integration / Position-Based

- 对欧拉更新后的粒子位置进行约束，防止发散
- 使用约束位置计算速度

刚体模拟（Rigid Body Simulation）

- 满足公式：

$$
\frac{d}{dt}\begin{bmatrix} X \\ \theta \\ \dot{X} \\ \omega \end{bmatrix} = \begin{bmatrix} \dot{X} \\ \omega \\ F/M \\ \Gamma/I \end{bmatrix}
$$

  - X表示位置，θ表示旋转角度，ω表示角速度，F表示受力，Γ表示扭矩，I表示惯性动量
- 基于上式ODE，使用牛顿法等数值方法求解

流体模拟（Fluid Simulation）

Position-Based方法

- 假定水体由刚体小球构成、水体不可压缩
- 若某处的密度产生变换，需要通过改变小球位置修正该处的密度，需要得到密度关于小球位置的导数→梯度下降
- 模拟小球运动→渲染得到水体运动

大规模物体模拟方法

1. 质点法（Lagrange Approach）：对每个粒子的运动规律、材质属性进行建模

2. 网格法（Eulerian Approach）：分网格，考虑网格内随时间变化规律，以网格为观察对象
3. 混合方法（Materal Point Method）：同时考虑粒子属性和网格，粒子将属性传递给网格，网格进行更新，将更新后的结果返回给粒子