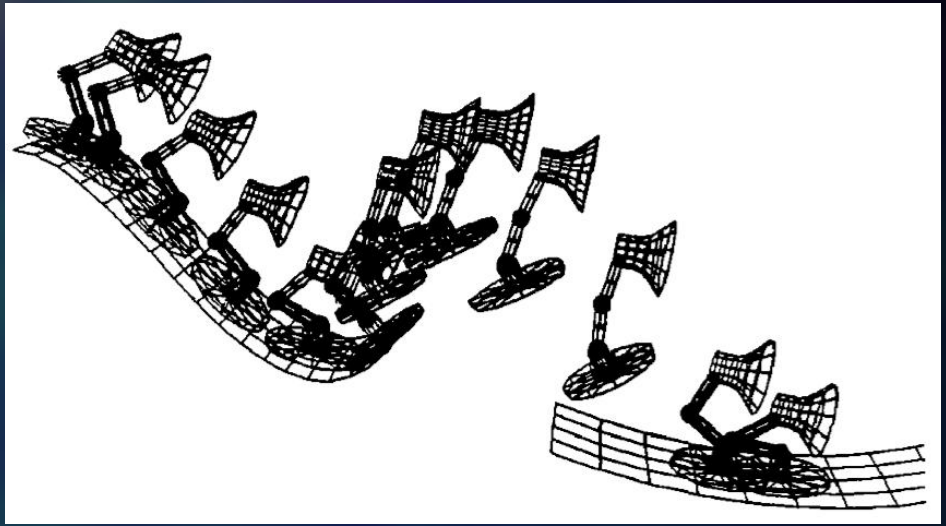


A ski-jumping Luxo, Jr. from Spacetime Constraints, 1988

CNM 190

Advanced Digital Animation

Lec 10 : Inverse Kinematics & Automating Animation

Dan Garcia, EECS (co-instructor)

Greg Niemeyer, Art (co-instructor)

Jeremy Huddleston, EECS (TA)

Overview

Dan

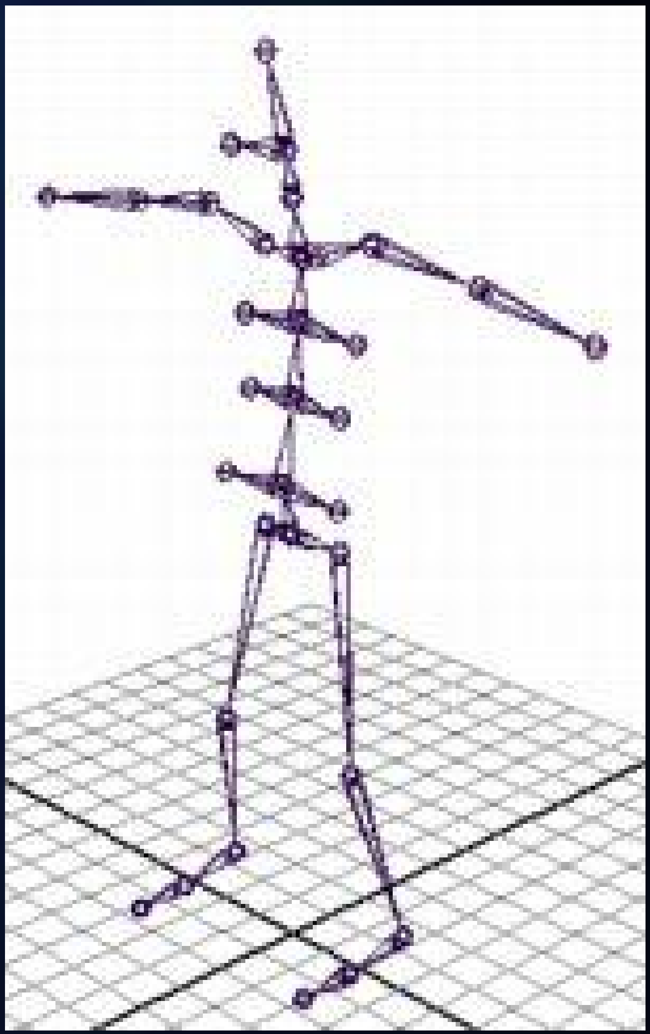
- Forward vs Inverse Kinematics
- Automating Animation
- Short film study
  - Bounding
  - Mike's new car
  - For the Birds
  - In the Rough

Jeremy

- Demo of Mel tools for
  - Importing data
  - Creating UI elements

Background You Know Already

- 3D rigid model
  - Usually given in "da Vinci" or "relaxed bind" pose
- Rigging
  - Designing a hierarchical skeleton
  - Use fewest joints as possible!
- Binding
  - Connecting the character's geometry to its skeleton
- Animation
  - Moving skeleton moves character due to binding
  - How to animate?



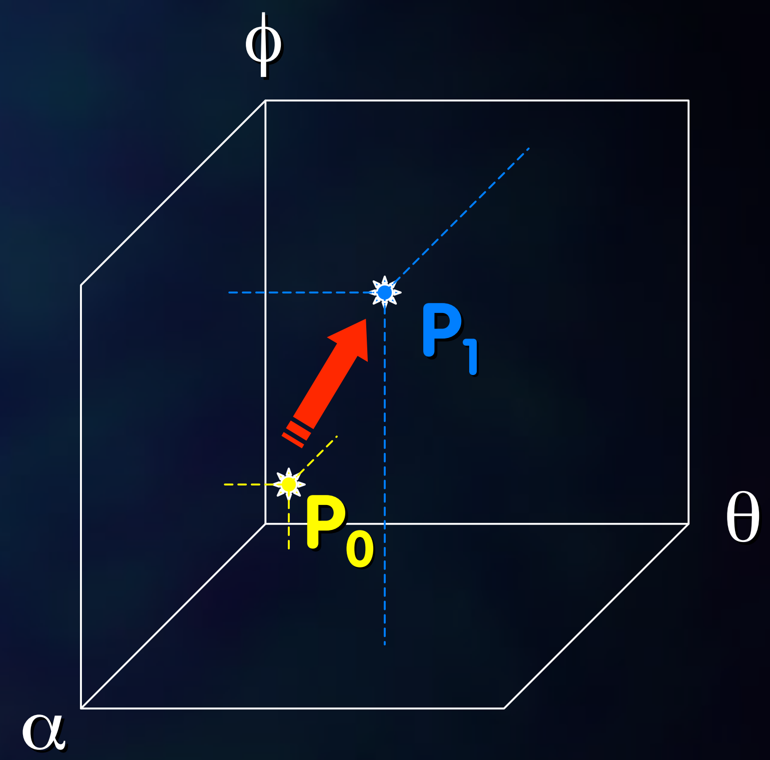
http://inst.eecs.berkeley.edu/~selfpace/

Forward vs Inverse Kinematics

- Forward Kinematics
  - (θ,α,φ) → (x,y,z)
  - Rotate top-level joints and the children joints follow automatically
  - No ambiguity
- Inverse Kinematics
  - (x,y,z) → (θ,α,φ)
  - Position the end joints and the inner joints bend to compensate
  - Usu lots of solutions!

How Does IK Work?

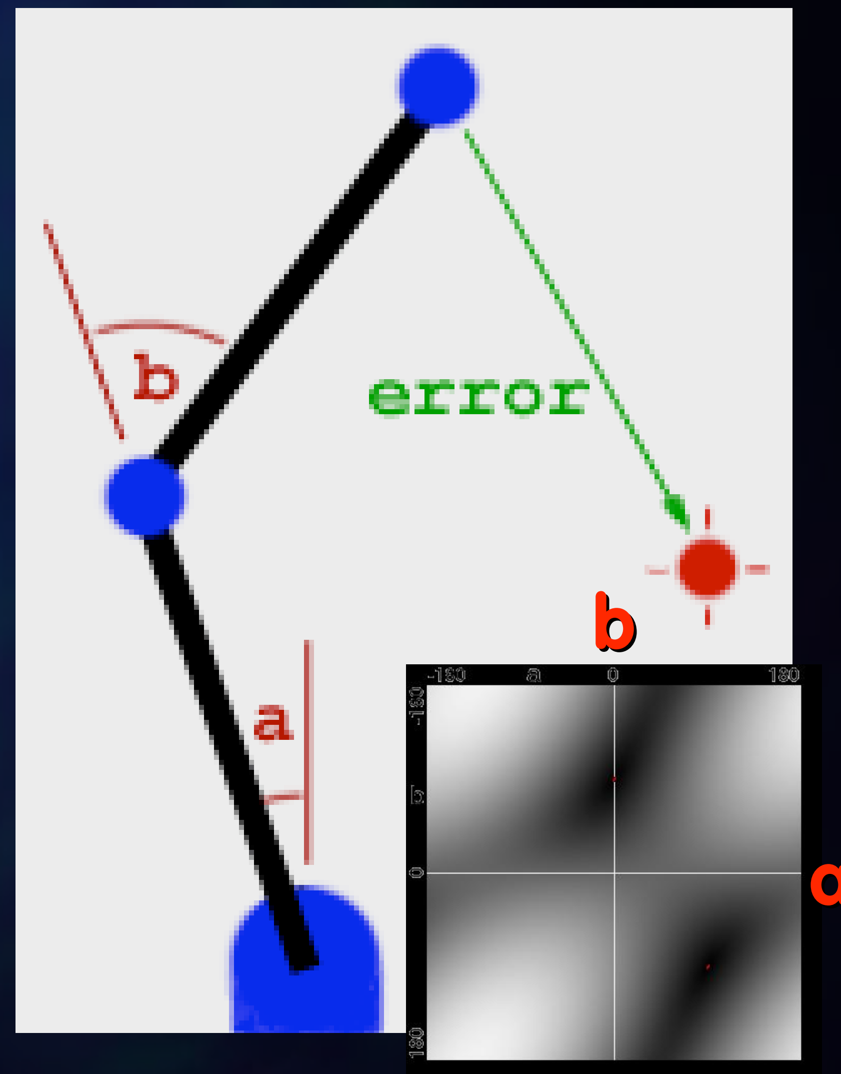
- You are at point P_0 in N-dim joint space $(\theta,\alpha,\phi,\ldots)$
- You're asked to move to (x,y,z,…) point P_1
  - It's not clear what values of $(\theta,\alpha,\phi,\ldots)$ yield that (x,y,z,…)
  - How do we get there?
- Answer: IK Solver
  - Any algorithm for doing this successfully
  - Must factor in torques, dead spaces due to constraints, total energy



Let's Look at a 2D Example (1)

freespace.virgin.net/hugo.elias/models/m_ik2.htm

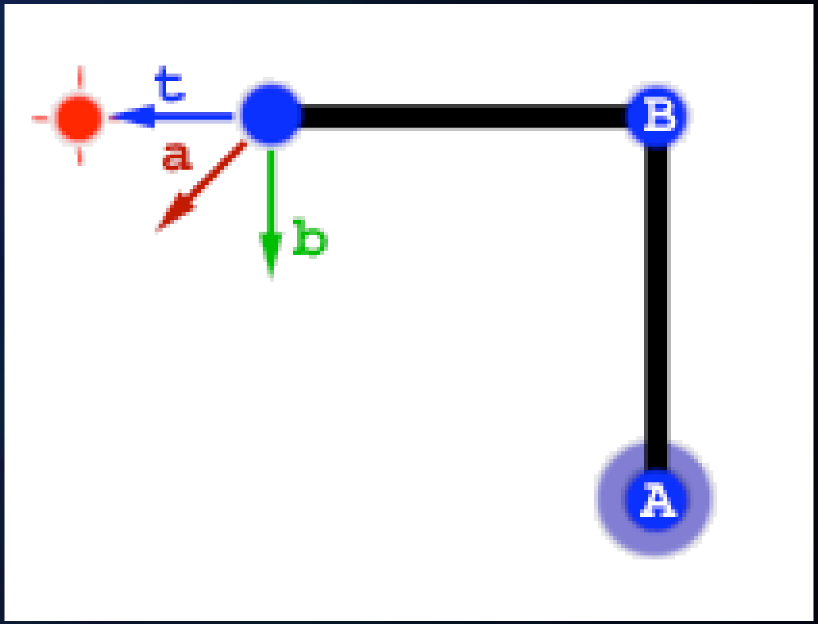
- Two-jointed, robot arm with red target
- We can measure how close we are at any point (from a,b->x,y)
- If we did this for all a,b angles, we'd get the graph to the right
  - Brightness is distance to the red goal
- We only know answer locally -- we search!



Let's Look at a 2D Example (2)

freespace.virgin.net/hugo.elias/models/m_ik2.htm

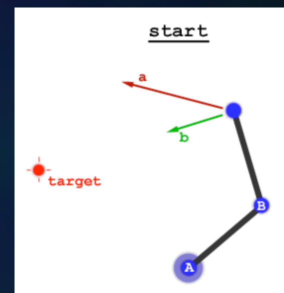
- Two-jointed robot arm with red goal target
- Rotating joint A moves the tip in a direction
  - This gets us closer to the solution
- Rotating joint B moves the tip in b direction
  - Here, this is of no use
- Most joints can rotate both clockwise and counter-clockwise
  - After a rotates a bit, we need to reverse-rotate b to extend
- We do this entire process incrementally, with small a, b

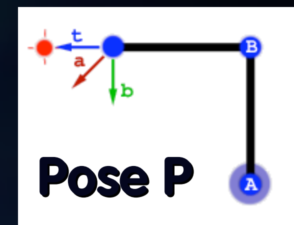


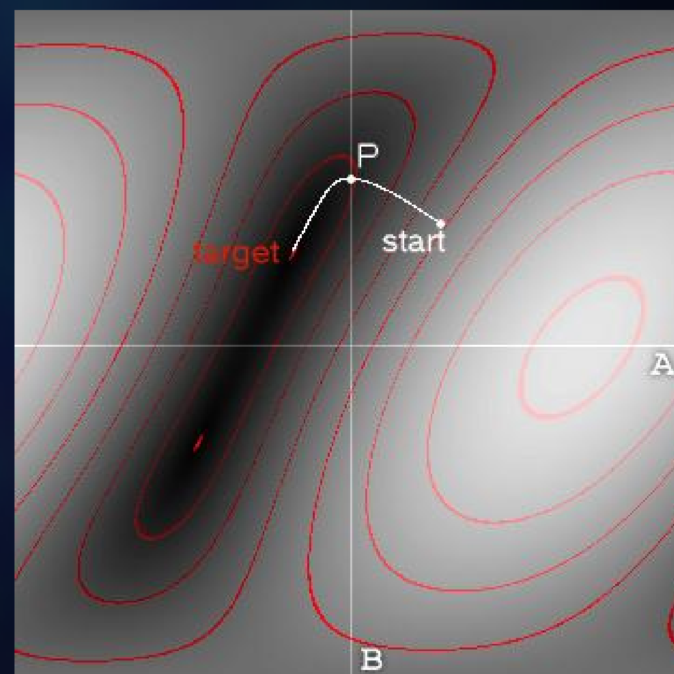
Let's Look at a 2D Example (3)

freespace.virgin.net/hugo.elias/models/m_ik2.htm

- Graph below shows error (distance to target) with contour equal value lines
- We examine local gradient
  - Which direction leads us fastest downhill? Go there!
    - "Simple Gradient Following", also known as "greedy"
  - Do this over and over until you can't go anymore
- When you stop, either you're
  - There, and you're done!
  - Not there, and you've reached
    - A local minimum
    - A constraint-based minimum



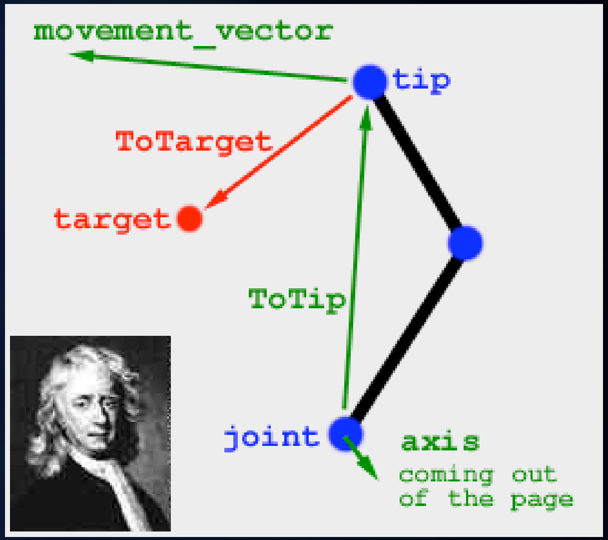




Let's Look at a 2D Example (4)

freespace.virgin.net/hugo.elias/models/m_ik2.htm

- Two ways to calculate gradient:
  - By measurement (move, calc)
  - By calculation (thanks, Newton)

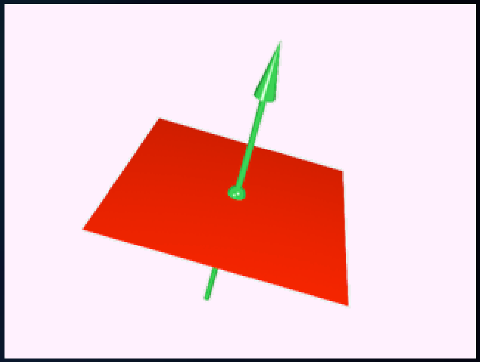


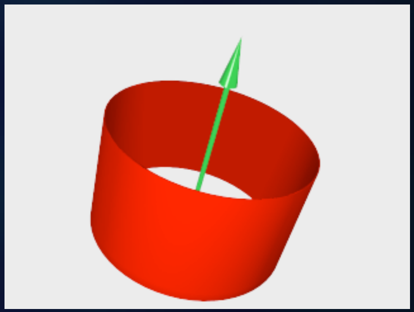

```
for each joint
  if 3D: axis = axis of rotation for this joint
  if 2D: axis = (0, 0, 1)
ToTip = tip - joint_centre
ToTarget = target - tip
movement_vector = crossproduct(ToTip, axis)
gradient = dotproduct(movement_vector, ToTarget)
end loop
```

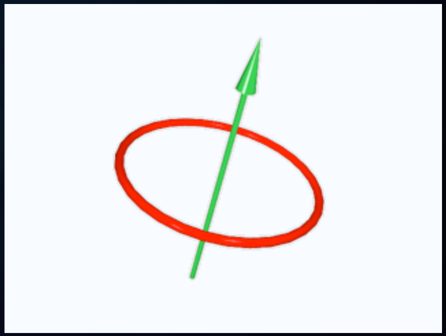
Let's Look at a 3D Example

freespace.virgin.net/hugo.elias/models/m_ik2.htm

- You don't need to always specify a fixed point in space for your end joint.
- Alternatively, you could specify another locus of points and move to the closest point on the surface







When to use FK vs IK?

FK

- Can control the rotation of a single joint and lock others
- No unexpected elbow/knee flipping
- Natural "arcs" by default
- Works with multi-joints as expected

IK

- Only need to move a single object to pose
- Can lock down an end effector (like wrist or ankle) while rest of body moves
- Good w/2-joint chains
- Great for legs!

Automating Animation

- **Delightful SIGGRAPH 1988 paper entitled: “Spacetime Constraints”**
  - Set up a system with real physics (torques, gravity)
  - …with space AND time as part of the object’s pose
  - Let the system try to figure out how to optimize
  - Automatic animation!
- **Very powerful idea, could use it in Olympic training simulation**