Announcements

- Assignment 3
  - **Theory** was due **Friday** (Nov 21st)
- Assignment 4
  - Is **out now**, due on Monday in class
  - Extra office hours for myself and Alex this week
  - Extra credits
  - Hints
    - Implement required options first (should be easy)
    - Implement and test one effect at a time
    - Choose effects that you know how to implement
    - Artistic work does count (!!!)
- Tutorial next week

Animation

Computer Graphics, CSCD18

Fall 2008

Instructor: Leonid Sigal

Key-frame Animation

- Define parameters at **key frames** and interpolate
  - Using splines or cubic interpolants

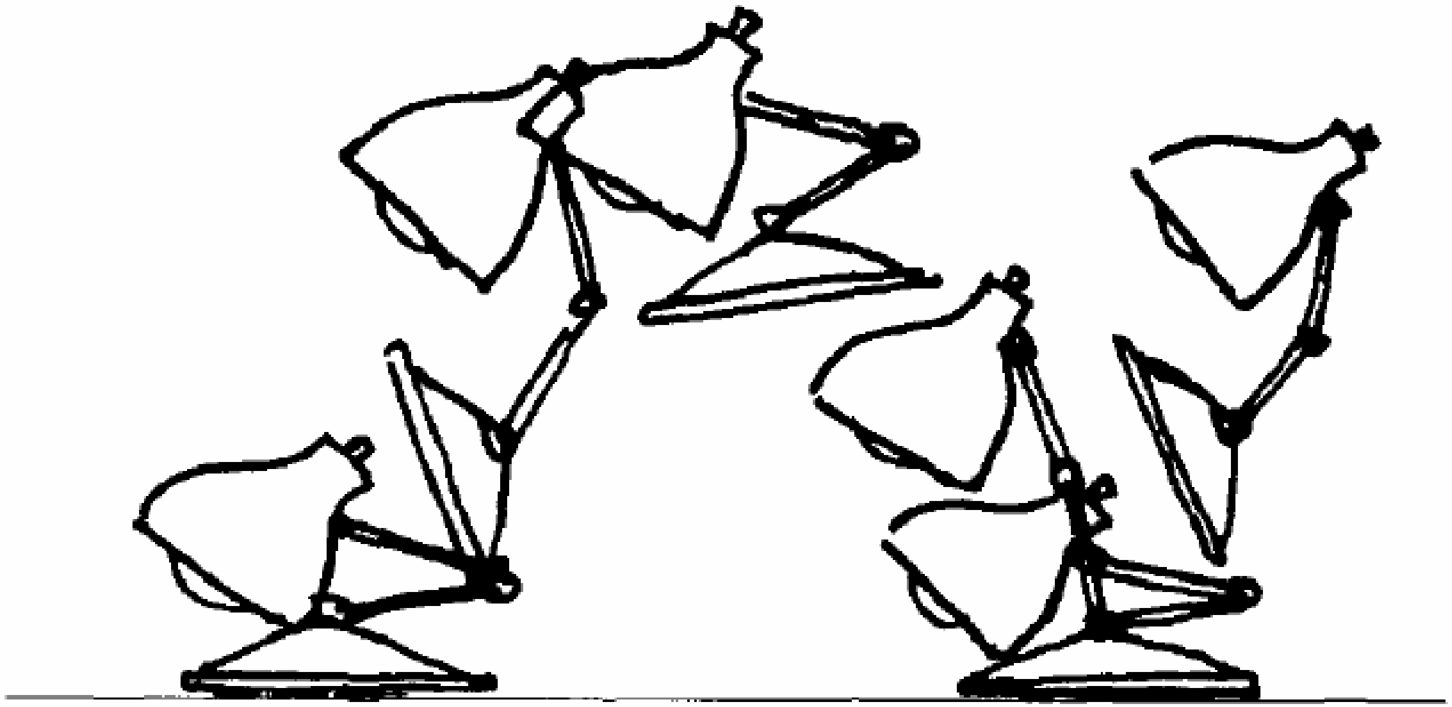


FIGURE 3. Squash & stretch in Luxo Jr.'s hop.

[Lasseter, 1987]

Problems

- We can easily get **physically implausible** solutions

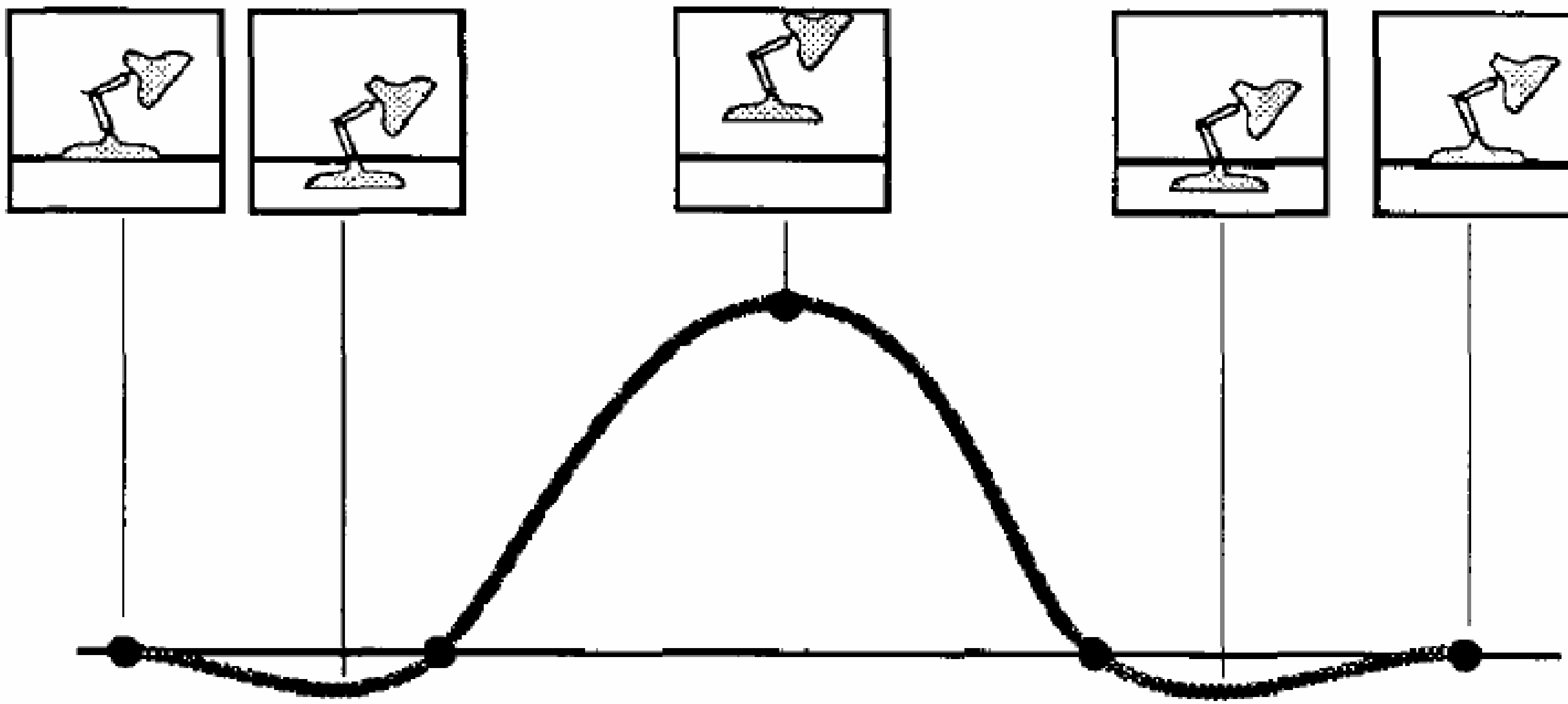


FIGURE 10a. This spline controls the Z (up) translation of Luxo Jr. Dips in the spline cause him to intersect the floor.

[*Lasseter, 1987*]

Solution

- Add sufficient number of keyframes so that animation looks physically plausible

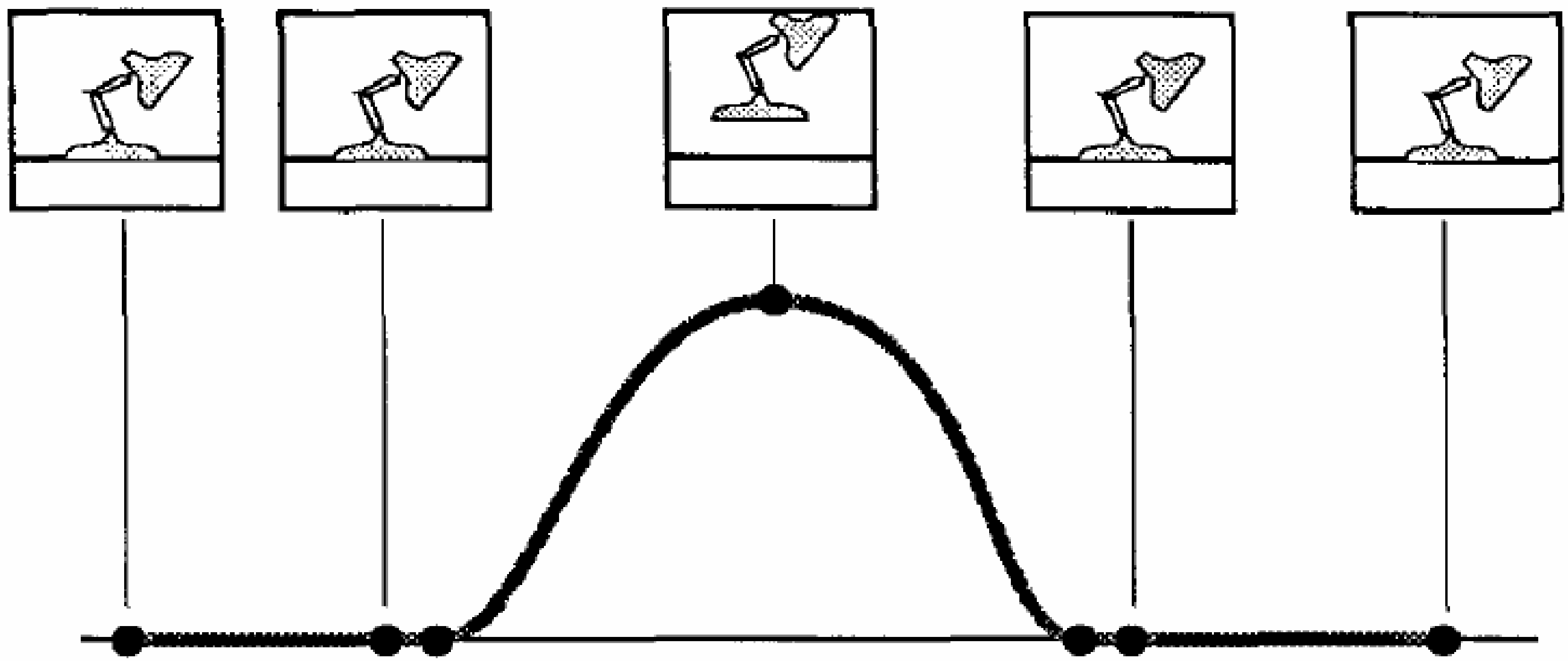


FIGURE 10b. Two extra extremes are added to the spline which removes the dips and prevents Jr. from going into the basement.

[*Lasseter, 1987*]

Key-frame Animation

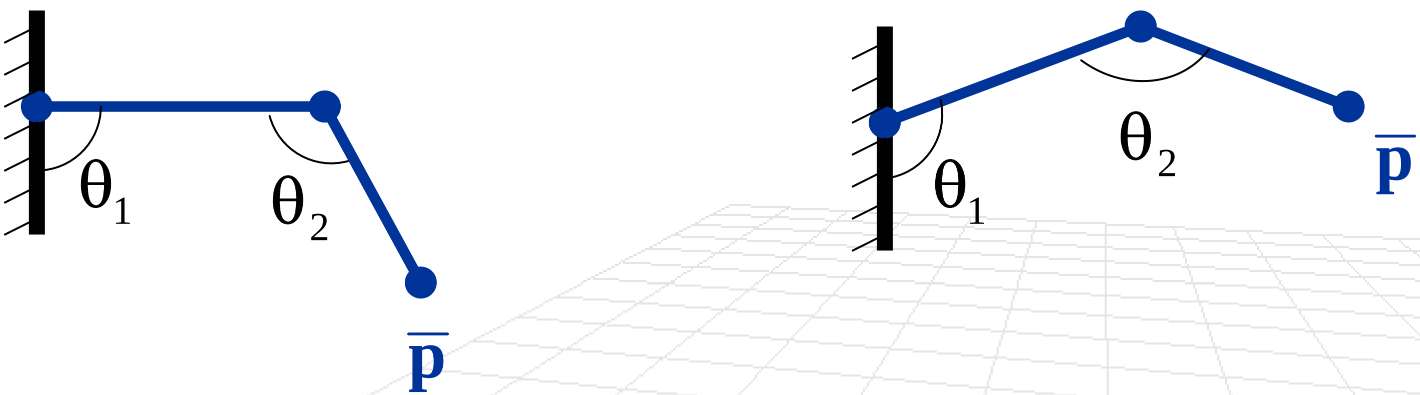
- **Pros:**
  - Very expressive
  - Animator has full control over animation
- **Cons:**
  - Very labor intensive (for the animator)
  - Difficult to create convincing physical realism
- **Use**
  - Potentially anything except complex physical simulations (e.g. smoke, water)

Articulated Structures

- How can we animate articulated structures (e.g. people, animals)
  - Forward Kinematics
  - Inverse Kinematics
  - Motion Capture

Forward Kinematics

- What if you have articulated structure?
  - Specify joint parameters at key frames

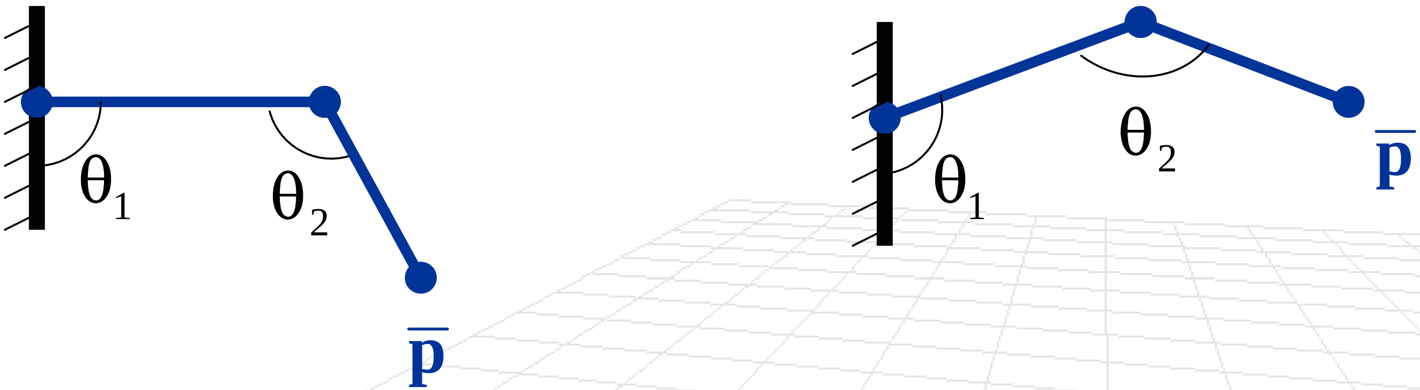


$$\mathbf{p_x} = \mathbf{l}_1 \cos(\theta_1) + \mathbf{l}_2 \cos(\theta_1 + \theta_2)$$

$$\mathbf{p_y} = \mathbf{l}_1 \sin(\theta_1) + \mathbf{l}_2 \sin(\theta_1 + \theta_2)$$

Forward Kinematics

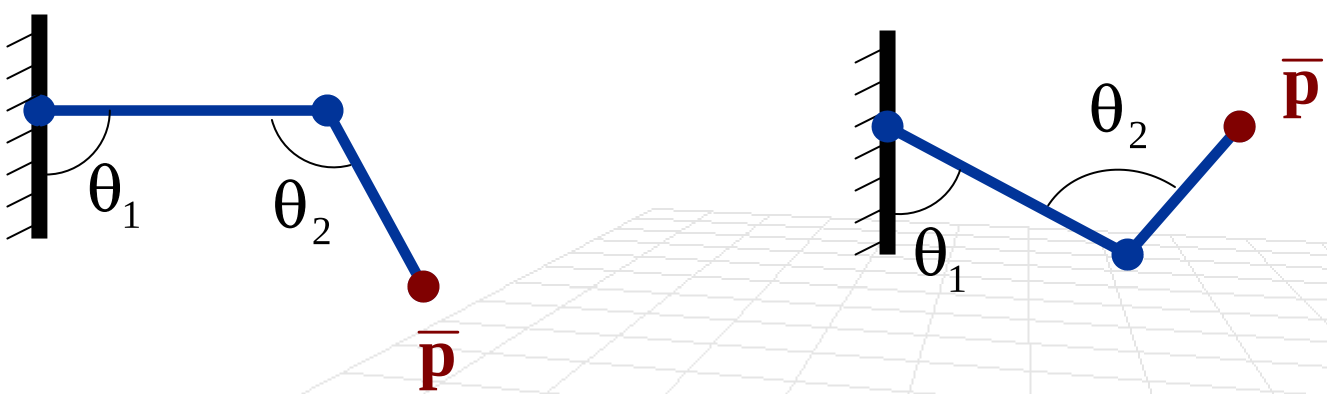
- What if you have articulated structure?
  - Specify joint parameters at key frames



$$\bar{\mathbf{p}} = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & \mathbf{l}_1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 \\ \sin\theta_2 & \cos\theta_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & \mathbf{l}_2 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

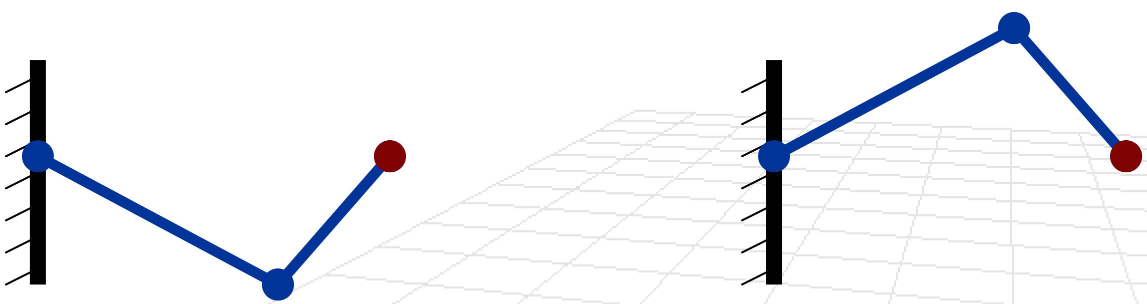
Inverse Kinematics

- What if you have articulated structure?
  - Instead specify the **“goal” states** and solve for **joint parameters**



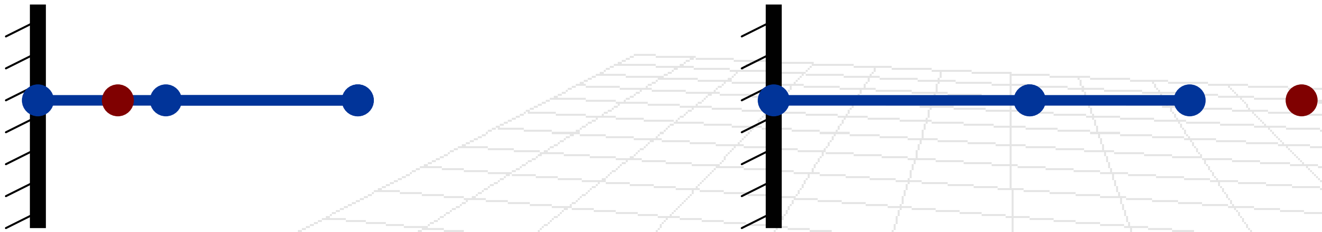
Inverse Kinematics

- What if you have articulated structure?
  - Instead specify the **"goal" states** and solve for **joint parameters**
- Problem: Solution may not be unique



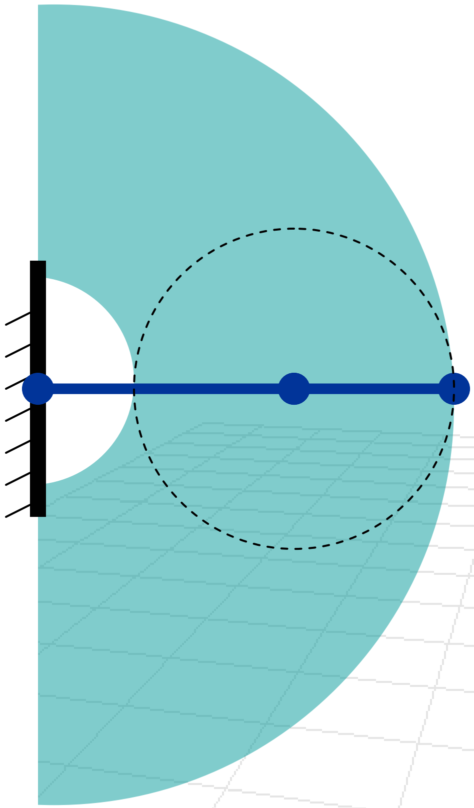
Inverse Kinematics

- What if you have articulated structure?
  - Instead specify the **"goal" states** and solve for **joint parameters**
- Problem: Solution may not exist

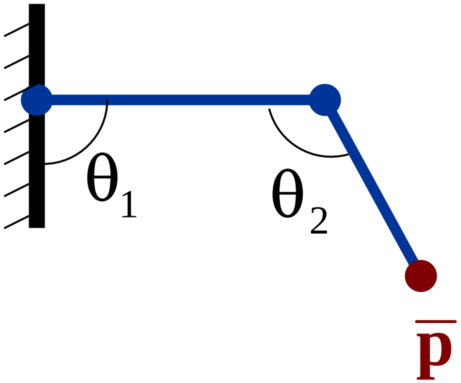


Inverse Kinematics

- We can characterize what is “reachable”



Direct Inverse Kinematics Solution

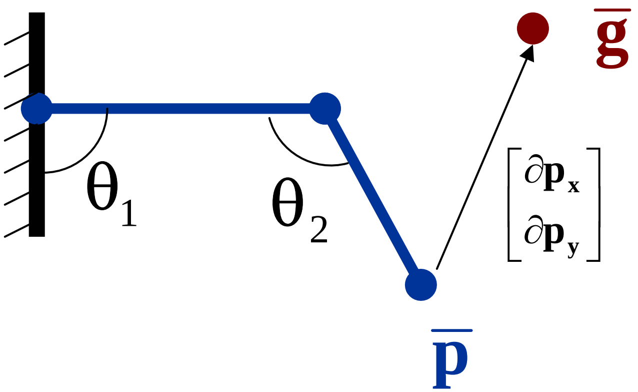


This is not always possible, and often hard to derive explicitly

$$\theta_2 = \cos^{-1}\left(\frac{\mathbf{p}_\mathbf{y}^2 + \mathbf{p}_\mathbf{x}^2 - \mathbf{l}_1^2 - \mathbf{l}_2^2}{2\mathbf{l}_1\mathbf{l}_2}\right)$$

$$\theta_1 = \frac{-\mathbf{p}_\mathbf{y}\mathbf{l}_2 \sin\theta_2 + \mathbf{p}_\mathbf{x}\left(\mathbf{l}_1 + \mathbf{l}_2\cos\theta_2\right)}{\mathbf{p}_\mathbf{x}\mathbf{l}_2 \sin\theta_2 + \mathbf{p}_\mathbf{y}\left(\mathbf{l}_1 + \mathbf{l}_2\cos\theta_2\right)}$$

Numerical Inverse Kinematics Solution



- Start from some initial configuration
- Define the goal state
- Define an error metric

$$\bar{\mathbf{g}} - \bar{\mathbf{p}}(\theta_1, \theta_2)$$

- Use linear approximation (Jacobian) and Newton's method to get a numeric solution

$$\begin{bmatrix} \partial \mathbf{p}_x \\ \partial \mathbf{p}_y \end{bmatrix} = \underbrace{\begin{bmatrix} \frac{\partial \mathbf{p}_x}{\partial \theta_1} & \frac{\partial \mathbf{p}_x}{\partial \theta_2} \\ \frac{\partial \mathbf{p}_y}{\partial \theta_1} & \frac{\partial \mathbf{p}_y}{\partial \theta_2} \end{bmatrix}}_{\text{Jacobian}} \begin{bmatrix} \partial \theta_1 \\ \partial \theta_2 \end{bmatrix}$$

Numerical Inverse Kinematics Solution

- **Problems**
  - Inverse of the **Jacobian may not always be invertible** (most cases actually)
    - Use pseudo-inverse
    - Robust iterative method
  - Jacobian **is not constant**

$$\mathbf{J}\begin{pmatrix}\theta_1 \\ \theta_2\end{pmatrix} = \begin{bmatrix} \dfrac{\partial \mathbf{p}_\mathbf{x}}{\partial \theta_1} & \dfrac{\partial \mathbf{p}_\mathbf{x}}{\partial \theta_2} \\ \dfrac{\partial \mathbf{p}_\mathbf{y}}{\partial \theta_1} & \dfrac{\partial \mathbf{p}_\mathbf{y}}{\partial \theta_2} \end{bmatrix}$$

  - Hence, this is an iteratively linear approximation to non-linear problem, but usually it is well behaved

Motion Capture

- Attach markers to the body (use the suite)
- Solve for the 3D positions of these markers by triangulating location observed by multiple cameras

Motion Capture
(Mechanical and Hybrid Systems)

Marker-less video-based motion capture is the future (remember computer vision)

(e.g. http://www.stanford.edu/~stefanoc/Markerless/Markerless.html)

Motion Capture is Diverse

Motion Capture

- What you see is what you get

- A lot of work in graphics is focused on
  - How can MoCap be adjusted to a given body size/shape, a.k.a. **Motion Re-targeting**
  - How can it be adopted to other environments, a.k.a. **Motion Adaptation**
  - How can it be used to enhance emotions, stylistic variations
  - How can we change the timings (e.g. make the motion faster/slower), a.k.a. **Motion Warping**

Motion Capture

- What you see is what you get
  - Because of this in some animated films MoCap is not used (directors want cartoon-like motions)

New animation controls make the characters' lips stick together a bit before opening. Farquaad's forehead wrinkles automatically.

Animators could use more than 750 controls to create Shrek's performance. Some controlled one joint or muscle, others controlled groups of several.

Motion Capture

- **Pros:**
  - Captures specific motion and style of an actor
- **Cons:**
  - Often not expressive enough
  - Time consuming and expensive
  - Difficult to edit
- **Use**
  - Character animation (especially for articulated characters)

High Fidelity Motion Capture

- Video