Character Animation

COS 426

Syllabus

I. Image processing

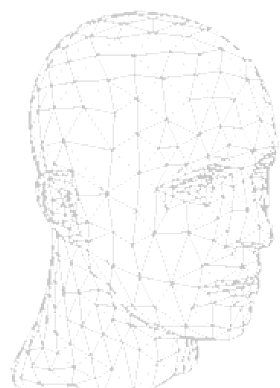
II. Modeling

III. Rendering

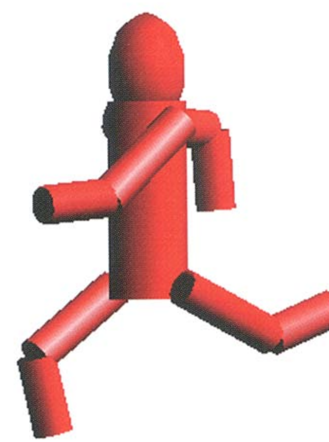
IV. Animation

Image Processing
(Rusty Coleman, CS426, Fall99)

Rendering
(Michael Bostock, CS426, Fall99)



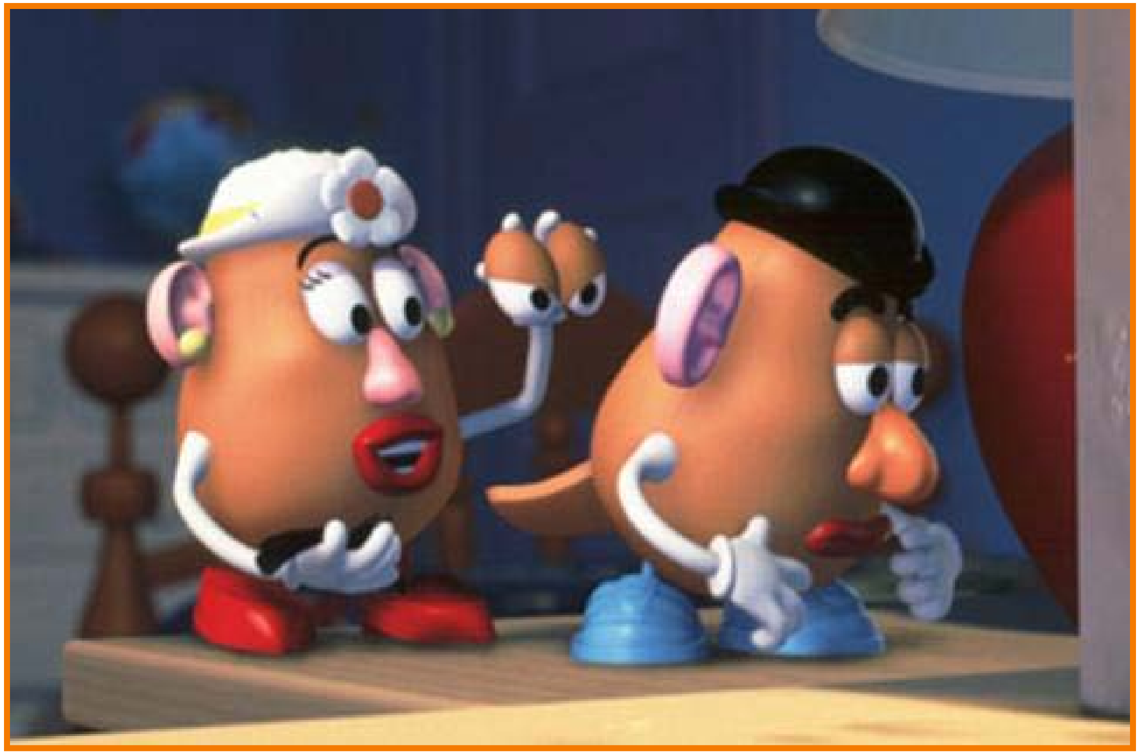
Modeling
(Dennis Zorin, CalTech)



Animation
(Angel, Plate 1)

Computer Animation

- Describing how 3D objects move over time



Pixar

Computer Animation

- Challenge is balancing between …
  - Animator control
  - Physical realism

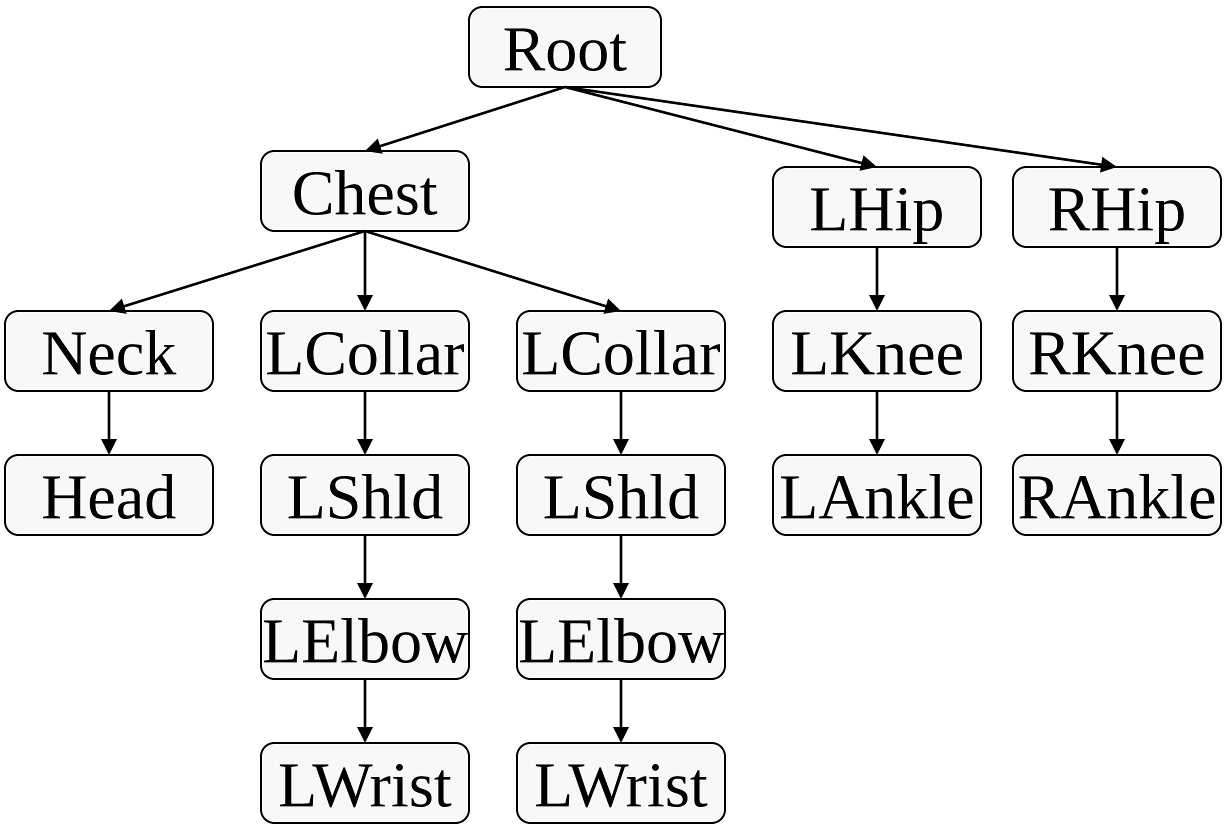
Paramount

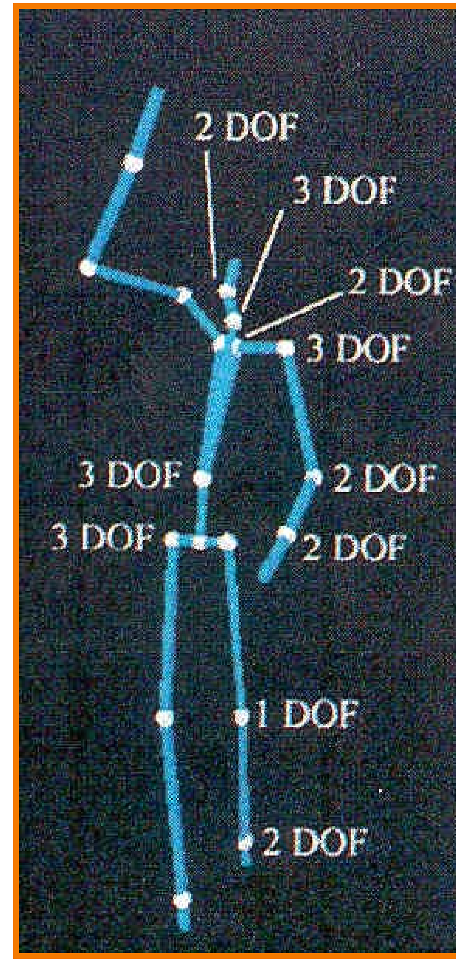
Computer Animation

Pixar

Character Animation

- Articulated figure:

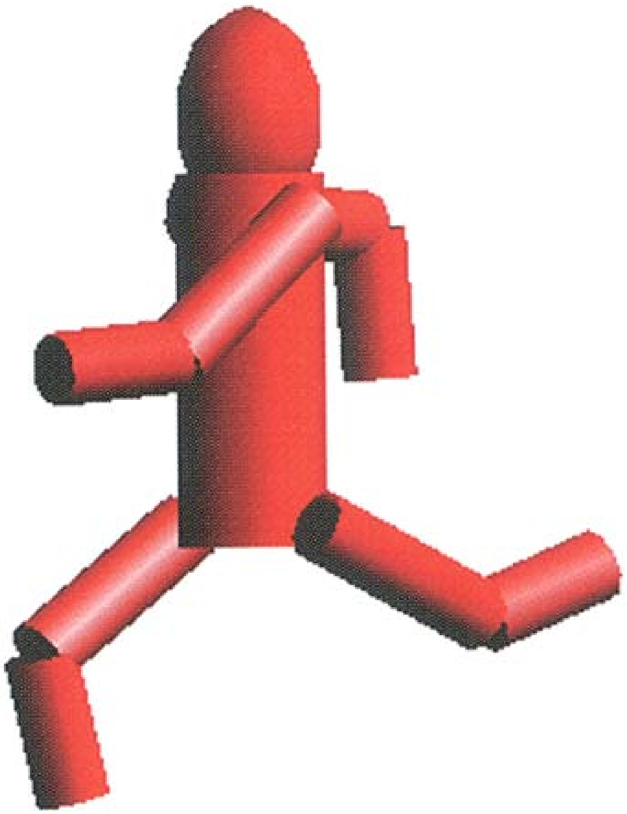




Rose et al. `96

Character Animation Methods

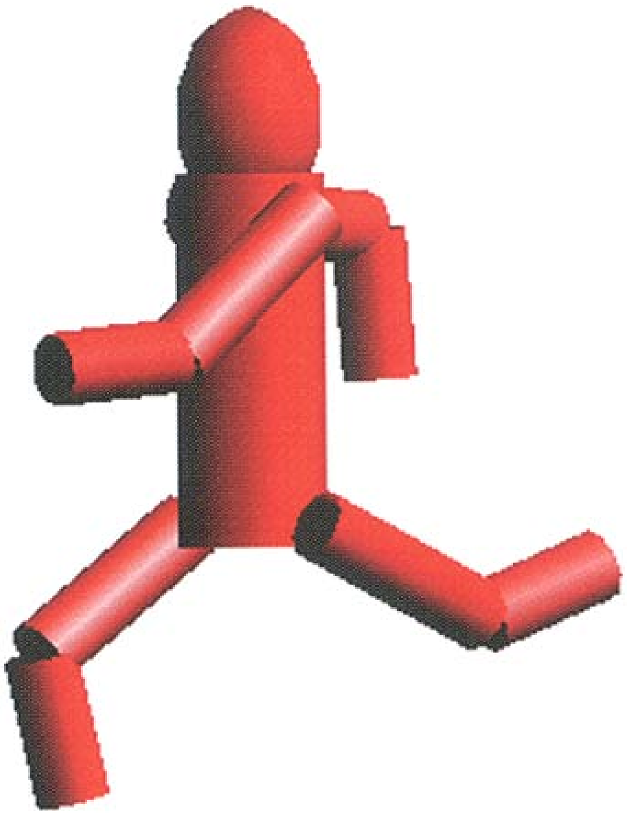
- Kinematics
- Dynamics
- Motion capture



Angel Plate 1

Character Animation Methods

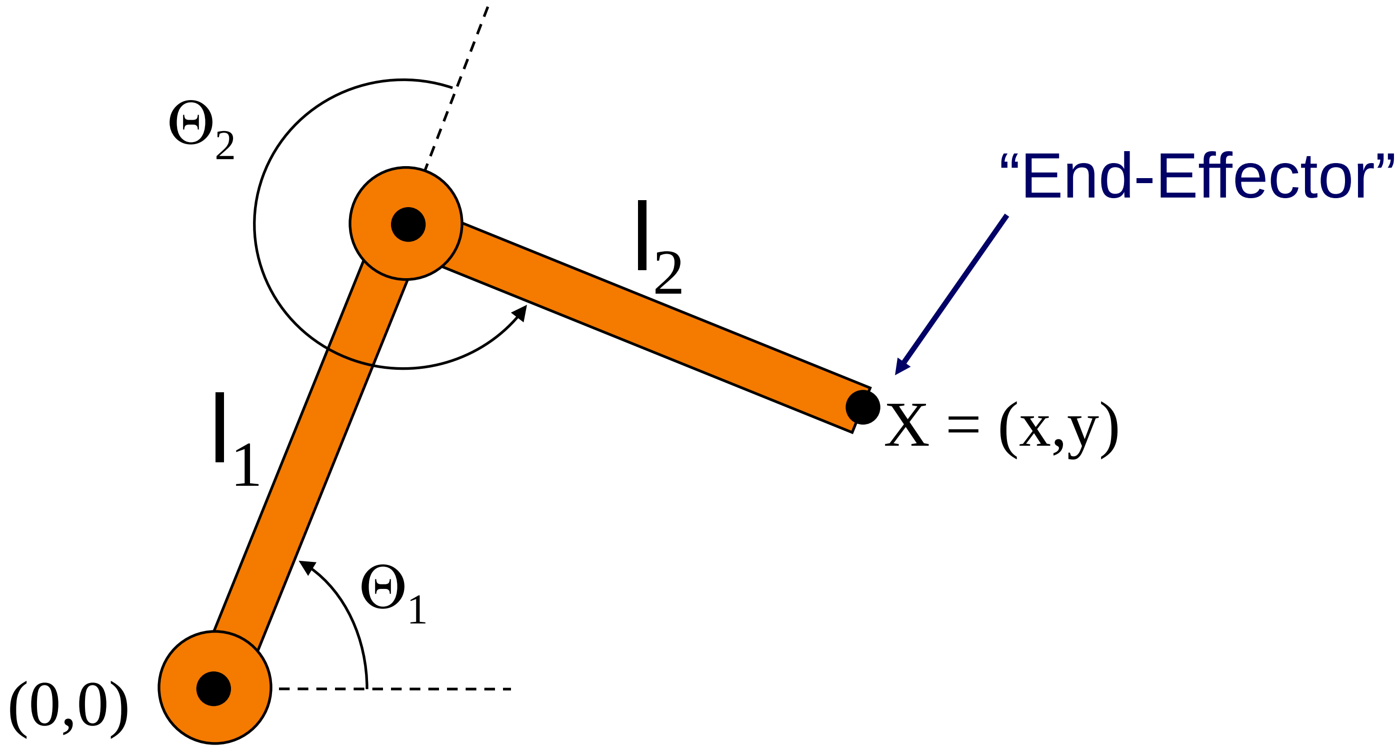
- Kinematics
- Dynamics
- Motion capture



Angel Plate 1

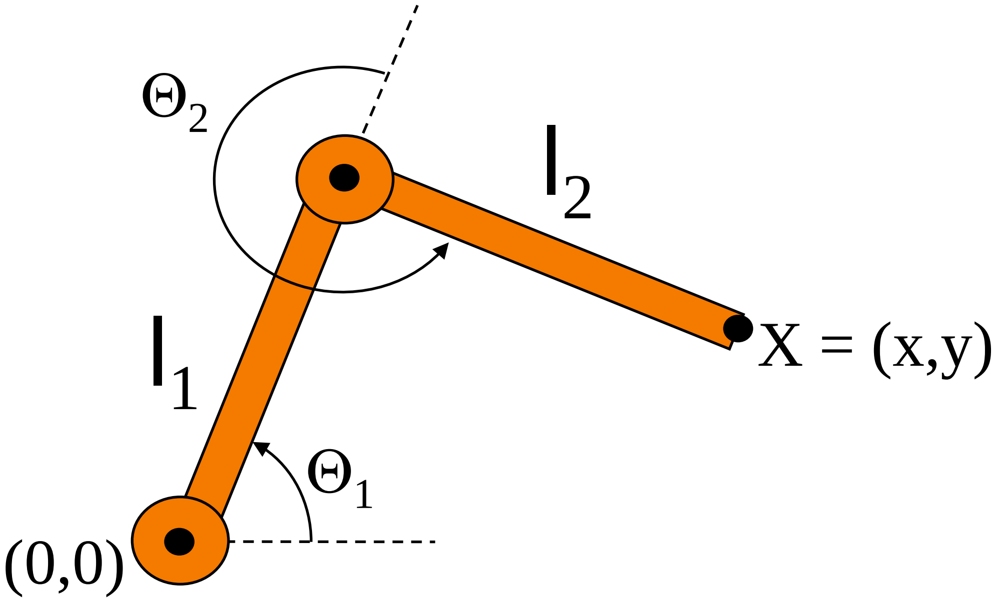
Kinematics

- Describe motion of articulated character



Forward Kinematics

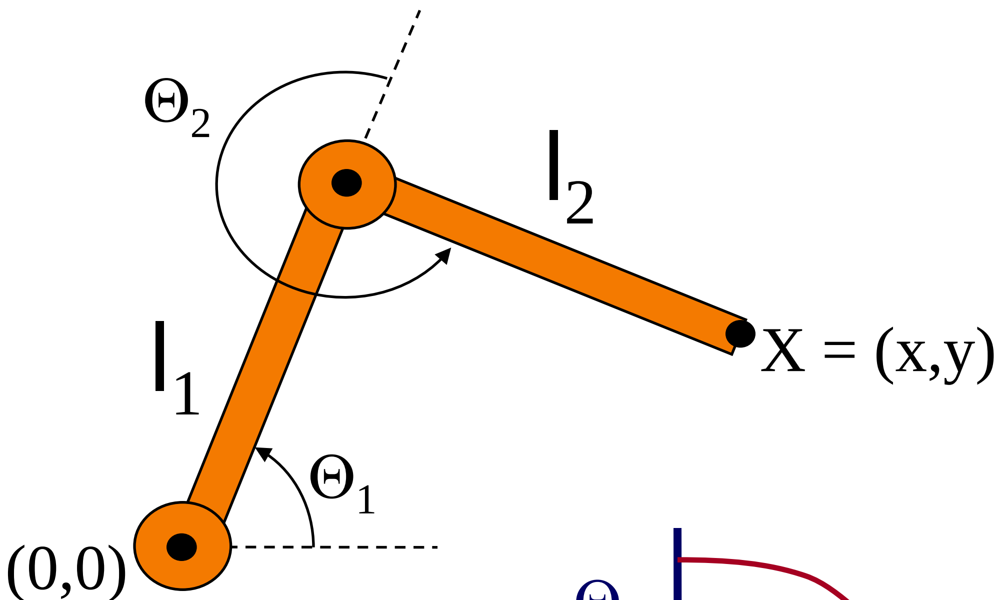
- Animator specifies joint angles: Θ_1 and Θ_2
- Computer finds positions of end-effector: X

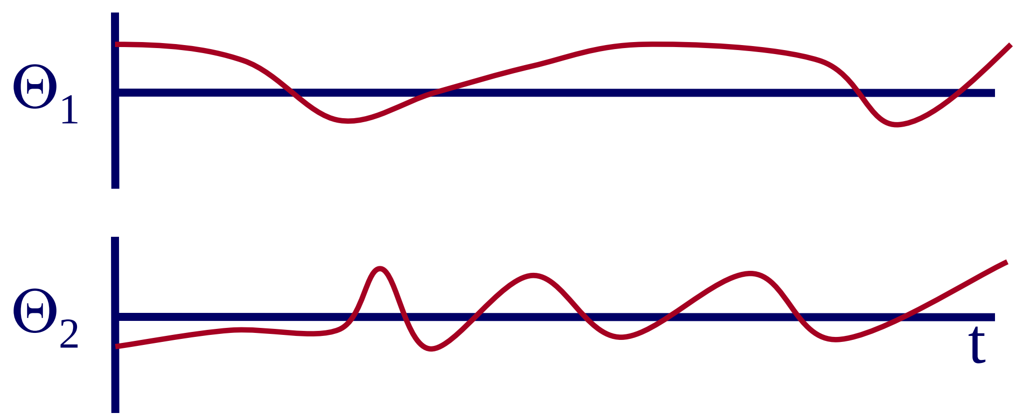


$$X = (l_1 \cos\Theta_1 + l_2 \cos(\Theta_1 + \Theta_2), l_1 \sin\Theta_1 + l_2 \sin(\Theta_1 + \Theta_2))$$

Forward Kinematics

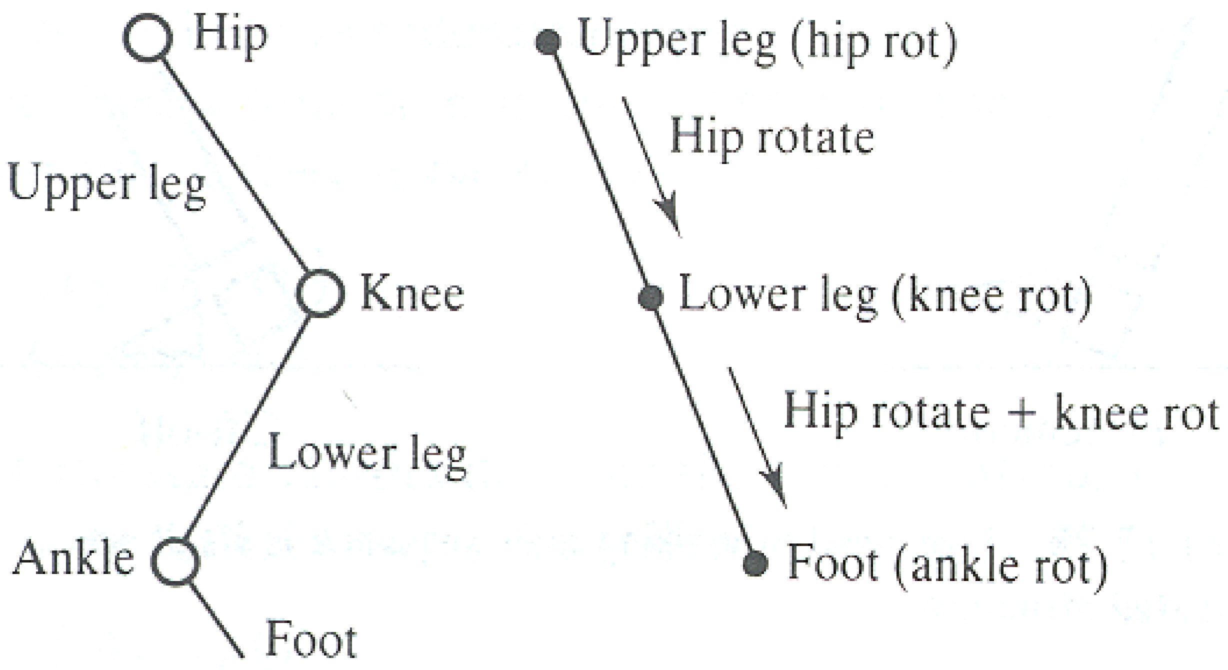
- Joint motions can be specified by spline curves





Example: Walk Cycle

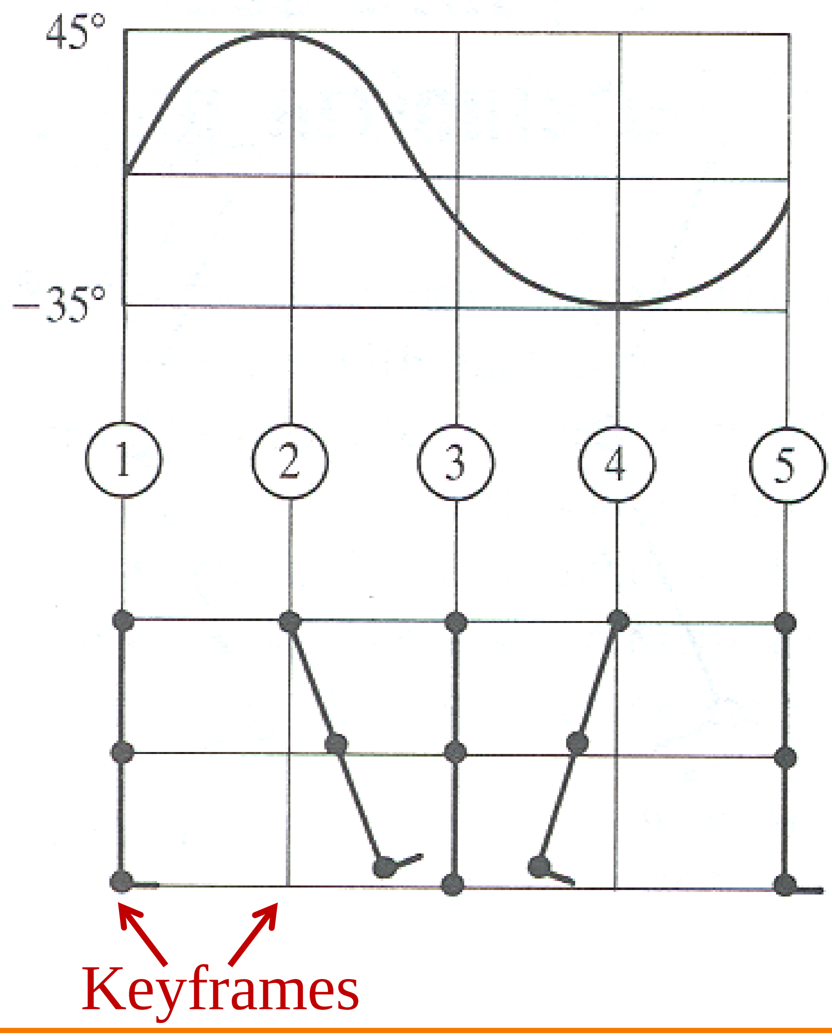
- Articulated figure:



Watt & Watt

Example: Walk Cycle

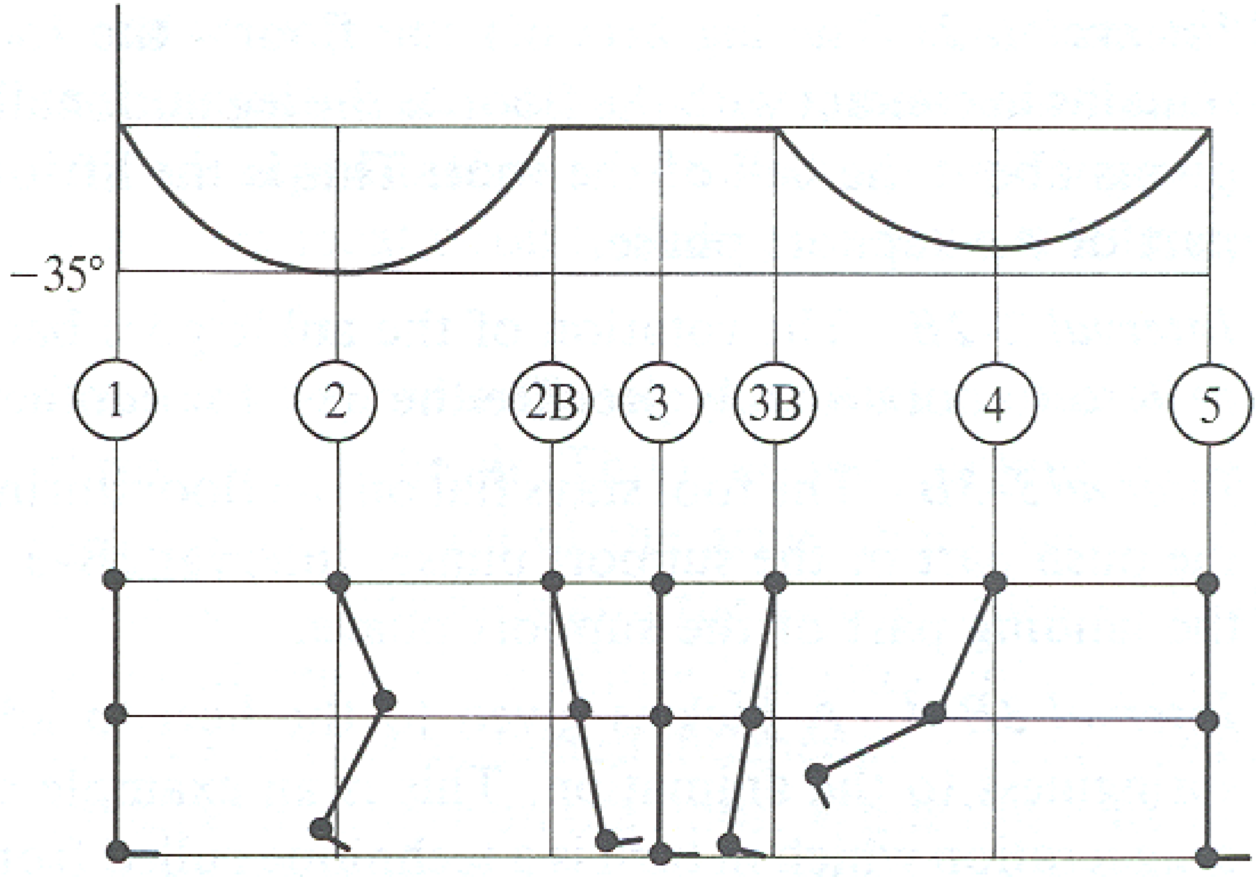
- Hip joint orientation:



Watt & Watt

Example: Walk Cycle

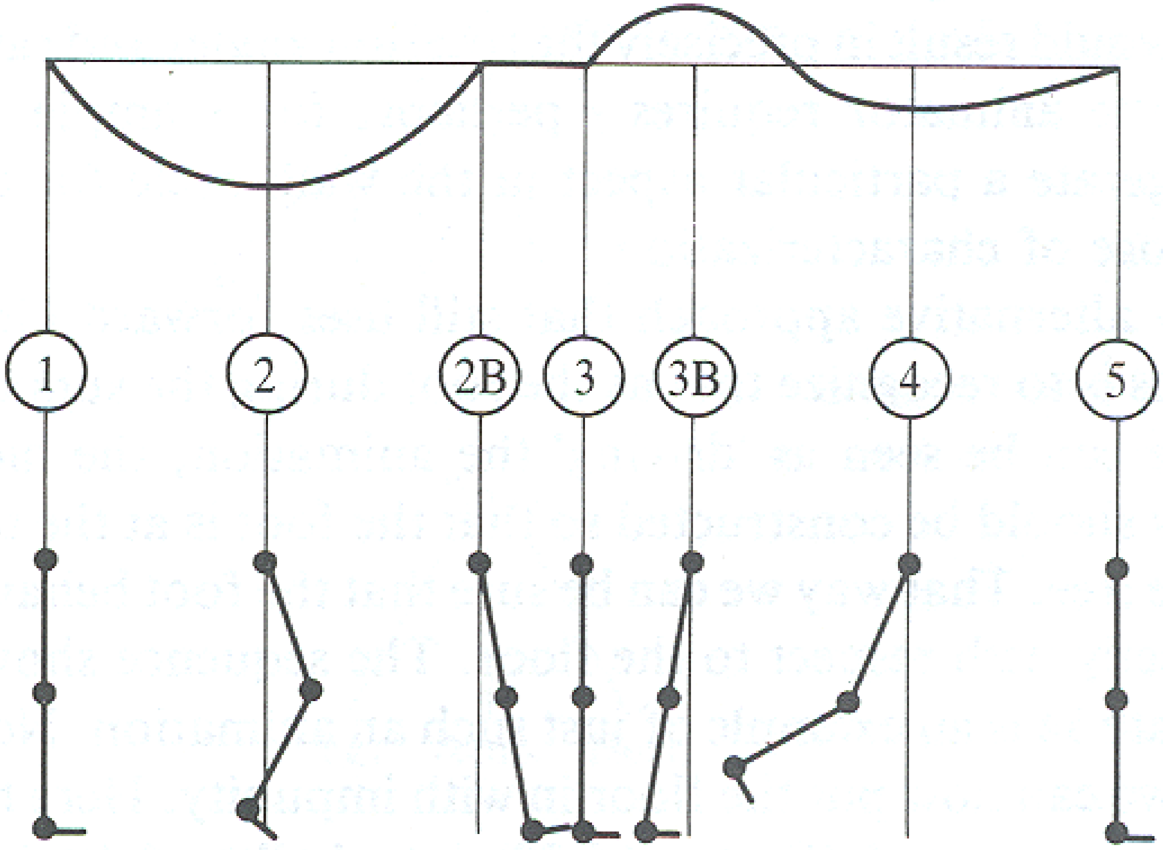
- Knee joint orientation:



Watt & Watt

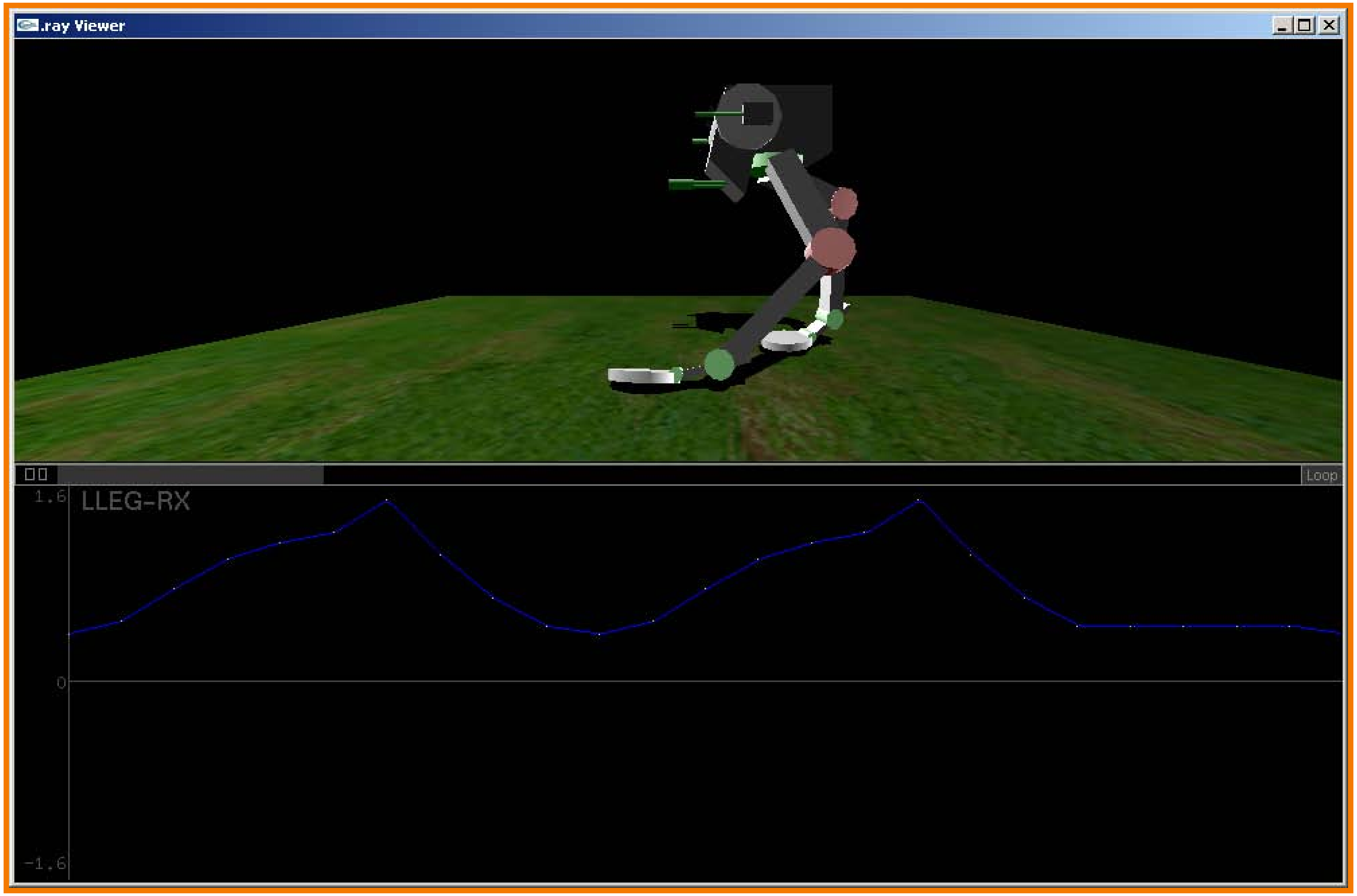
Example: Walk Cycle

- Ankle joint orientation:



Watt & Watt

Example: Robot



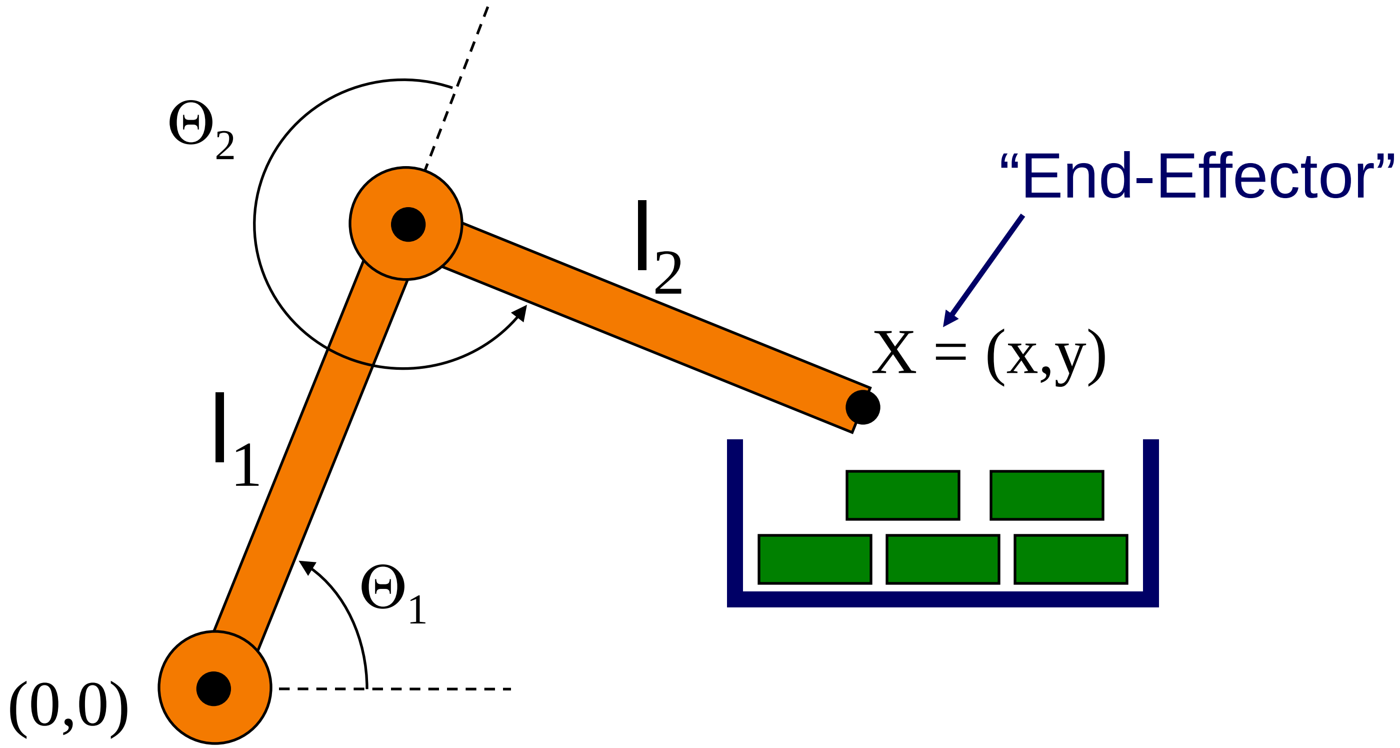
Mihai Parparita, COS 426, Princeton University, 2003

Example: Ice Skating

(Mao Chen, Zaijin Guan, Zhiyan Liu, Xiaohu Qie,
CS426, Fall98, Princeton University)

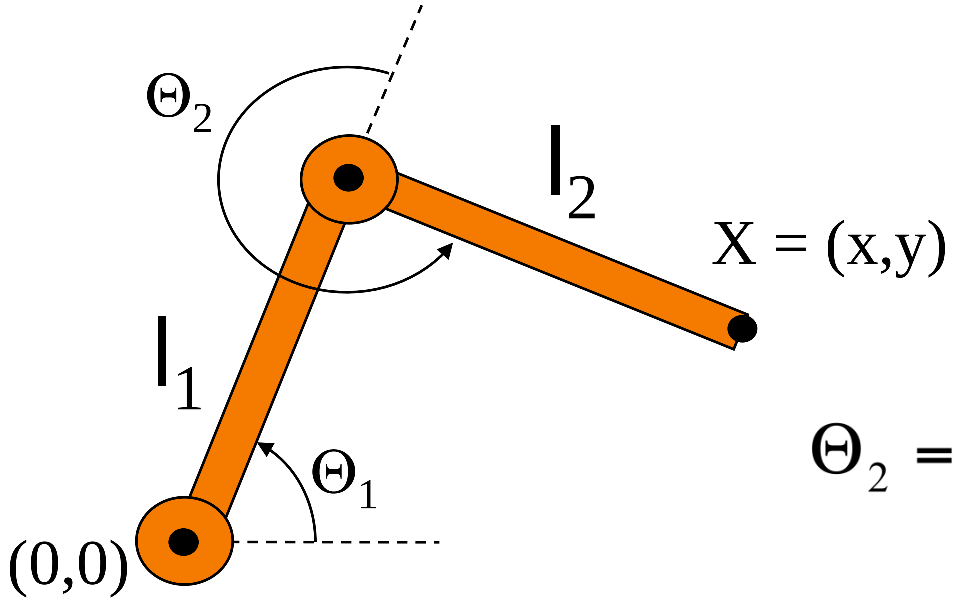
Inverse Kinematics

- What if animator knows position of “end-effector”



Inverse Kinematics

- Animator specifies end-effector positions: X
- Computer finds joint angles: Θ_1 and Θ_2:

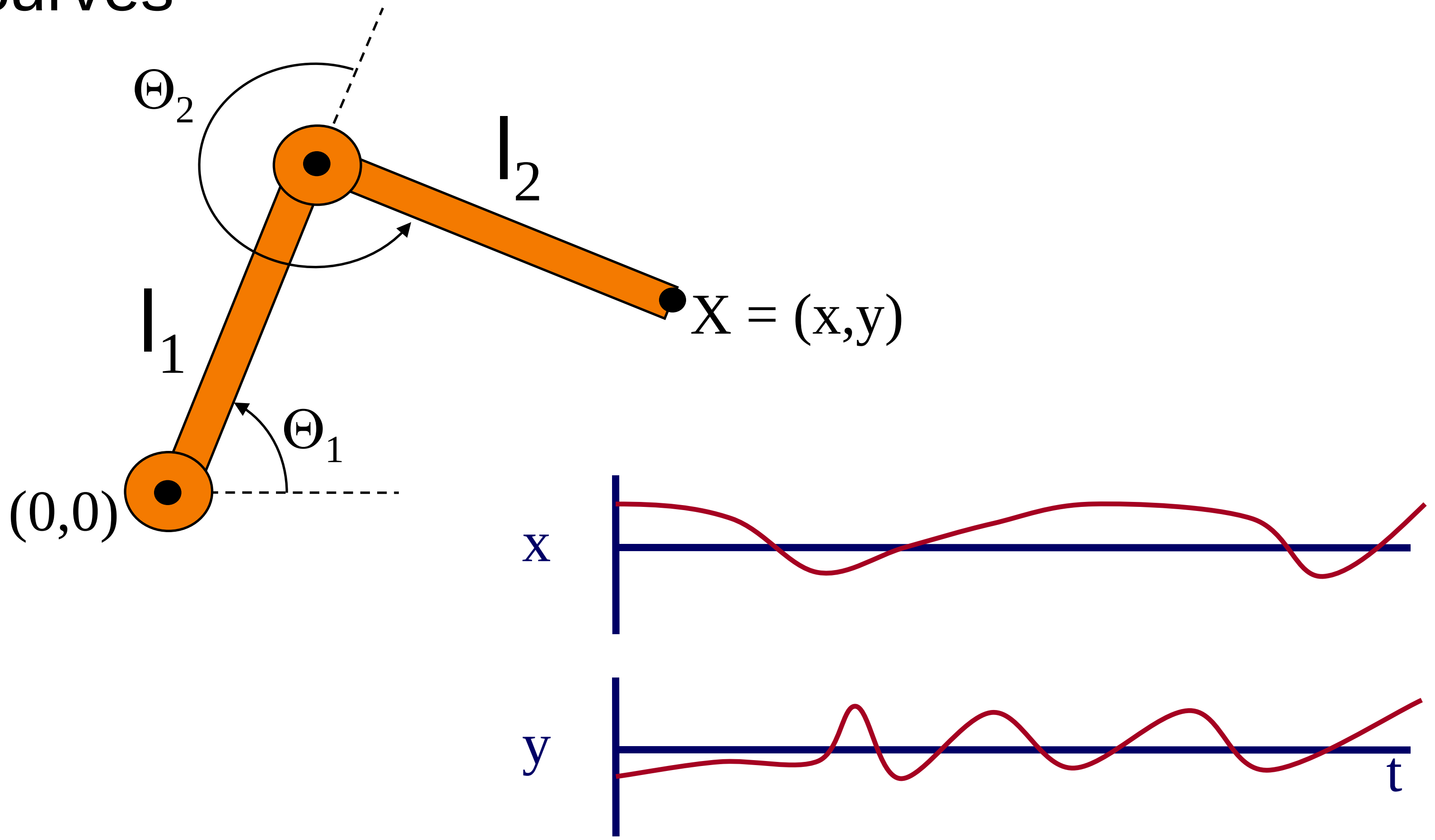


$$\Theta_2 = \cos^{-1}\left(\frac{x^2 + x^2 - l_1^{\,2} - l_2^{\,2}}{2 l_1 l_2}\right)$$

$$\Theta_1 = \frac{-(l_2 \sin(\Theta_2)x + (l_1 + l_2 \cos(\Theta_2))y}{(l_2 \sin(\Theta_2))y + (l_1 + l_2 \cos(\Theta_2))x}$$

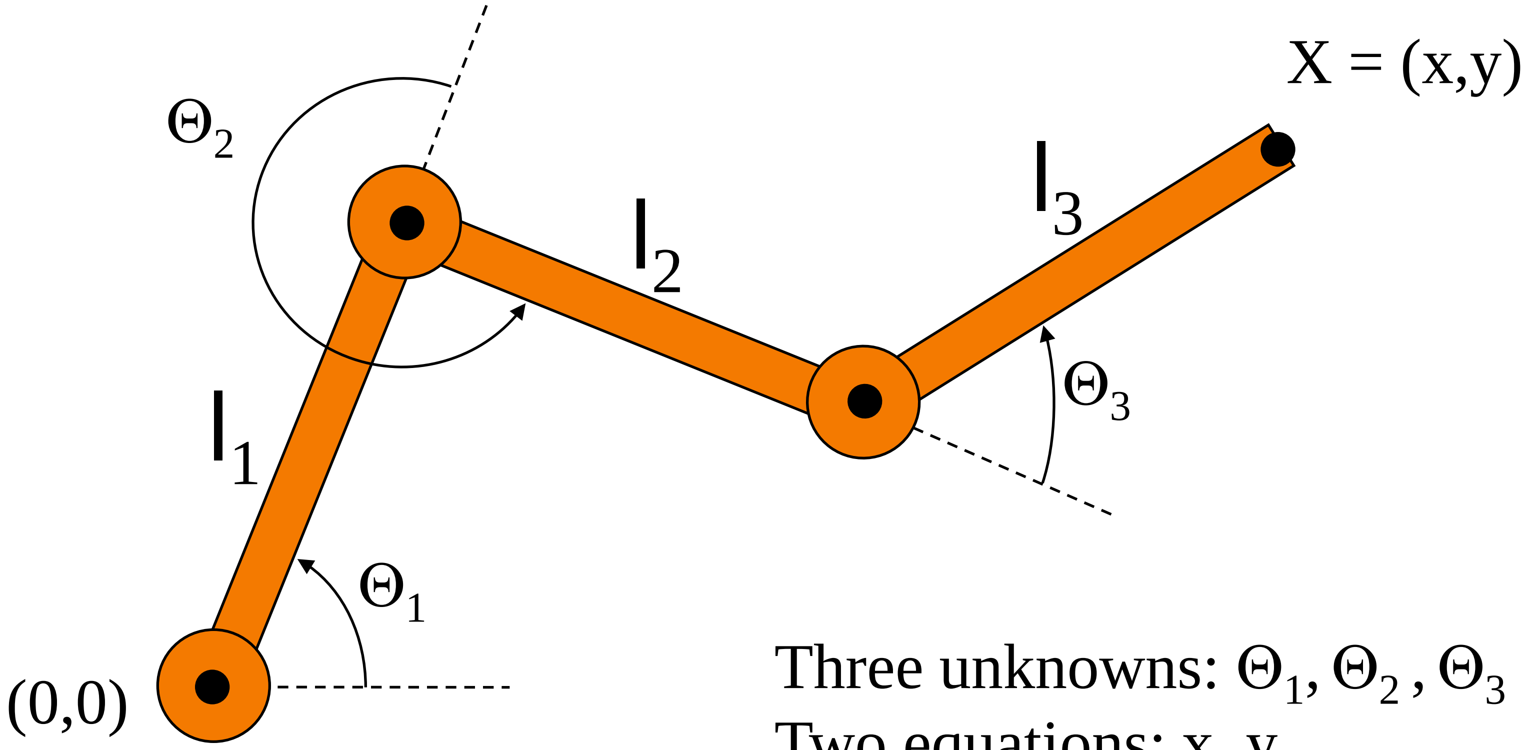
Inverse Kinematics

- End-effector postions can be specified by spline curves



Inverse Kinematics

- Problem for more complex structures
  - System of equations is usually under-defined
  - Multiple solutions

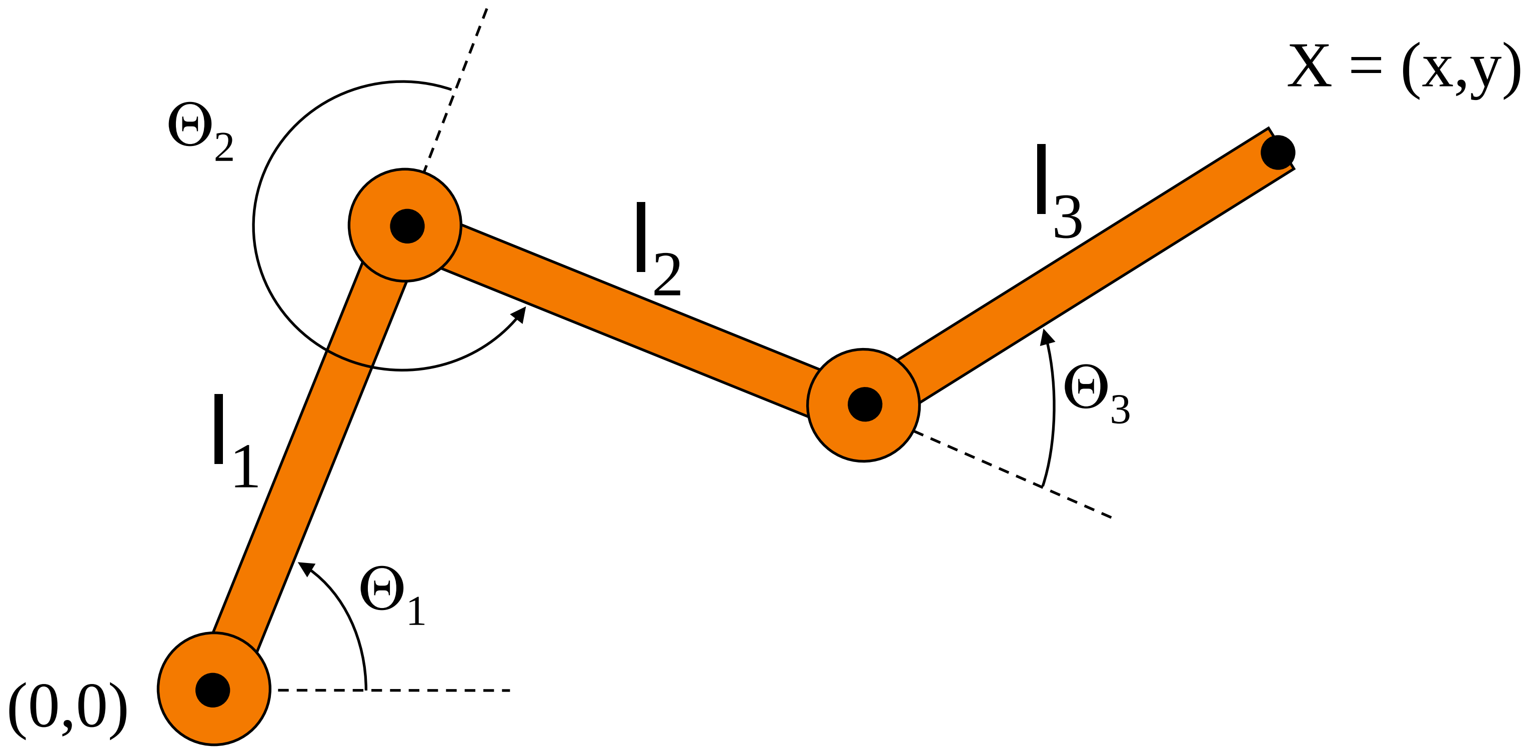


Three unknowns: Θ_1, Θ_2 , Θ_3

Two equations: x, y

Inverse Kinematics

- Solution for more complex structures:
  - Find best solution (e.g., minimize energy in motion)
  - Non-linear optimization



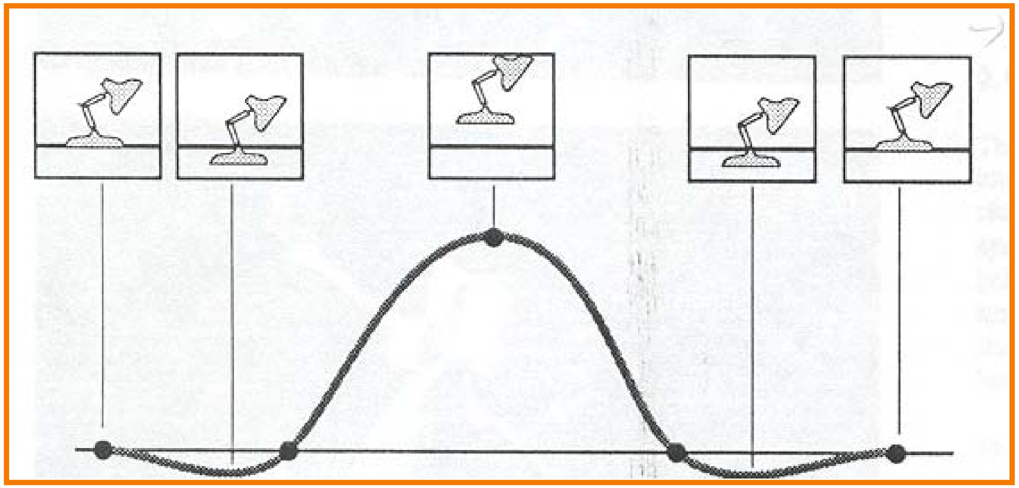
Example: Ball Boy

"Ballboy"

Fujito, Milliron, Ngan, & Sanocki
Princeton University

Kinematics

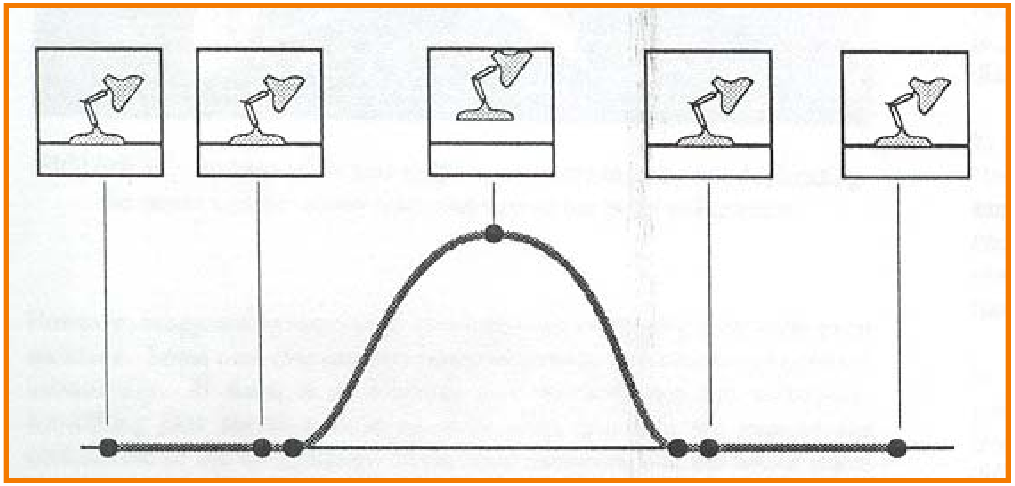
- Advantages
  - Simple to implement
  - Complete animator control
- Disadvantages
  - Motions may not follow physical laws
  - Tedious for animator



Lasseter `87

Keyframe Animation

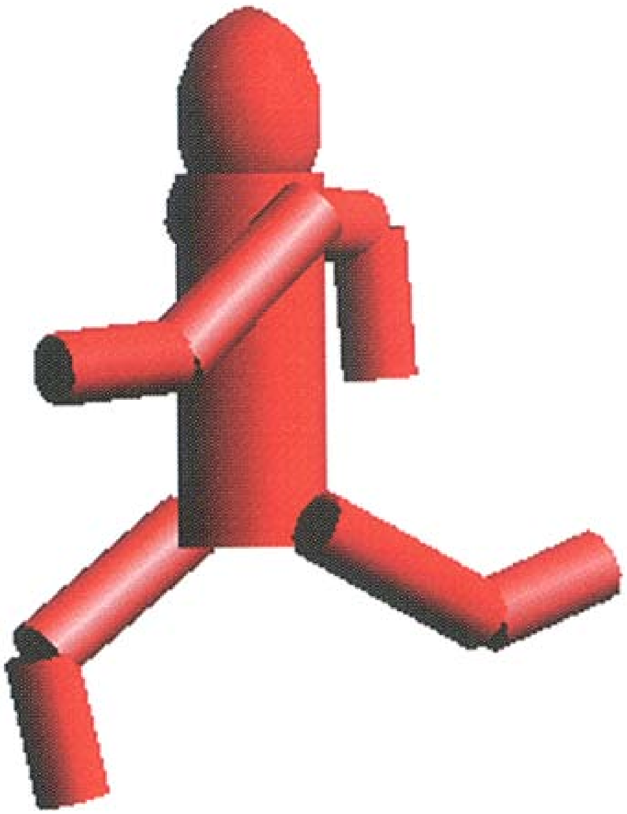
- Advantages
  - Simple to implement
  - Complete animator control
- Disadvantages
  - Motions may not follow physical laws
  - Tedious for animator



Lasseter `87

Outline

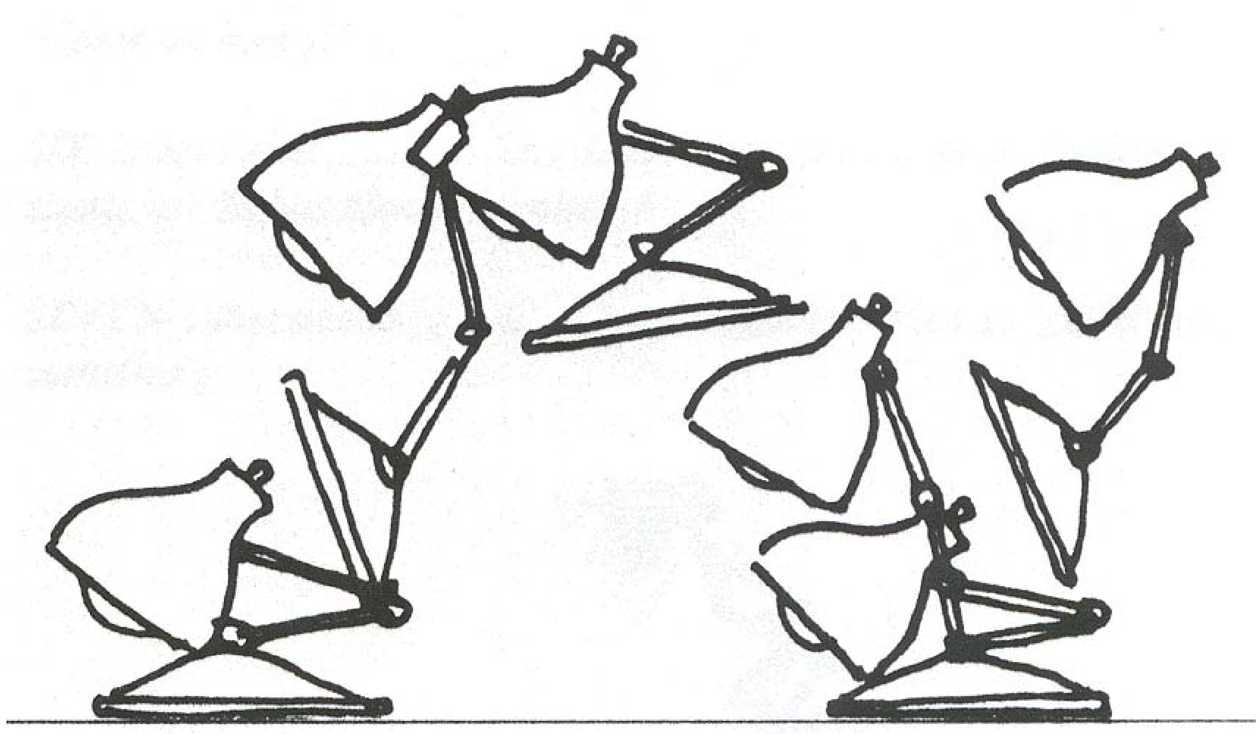
- Kinematics
- ➢ Dynamics
- Motion capture



Angel Plate 1

Dynamics

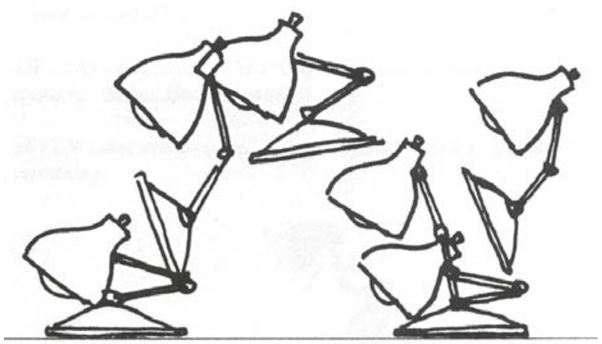
- Simulation of physics insures realism of motion



Lasseter `87

Spacetime Constraints

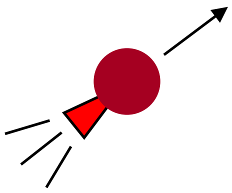
- Animator specifies constraints:
  - What the character's physical structure is
    - » e.g., articulated figure
  - What the character has to do (keyframes)
    - » e.g., jump from here to there within time t
  - What other physical structures are present
    - » e.g., floor to push off and land
  - How the motion should be performed
    - » e.g., minimize energy



Spacetime Constraints

- Computer finds the "best" physical motion satisfying constraints

- Example: particle with jet propulsion
  - **x**(t) is position of particle at time t
  - **f**(t) is force of jet propulsion at time t
  - Particle's equation of motion is:



$$mx'' - f - mg = 0$$

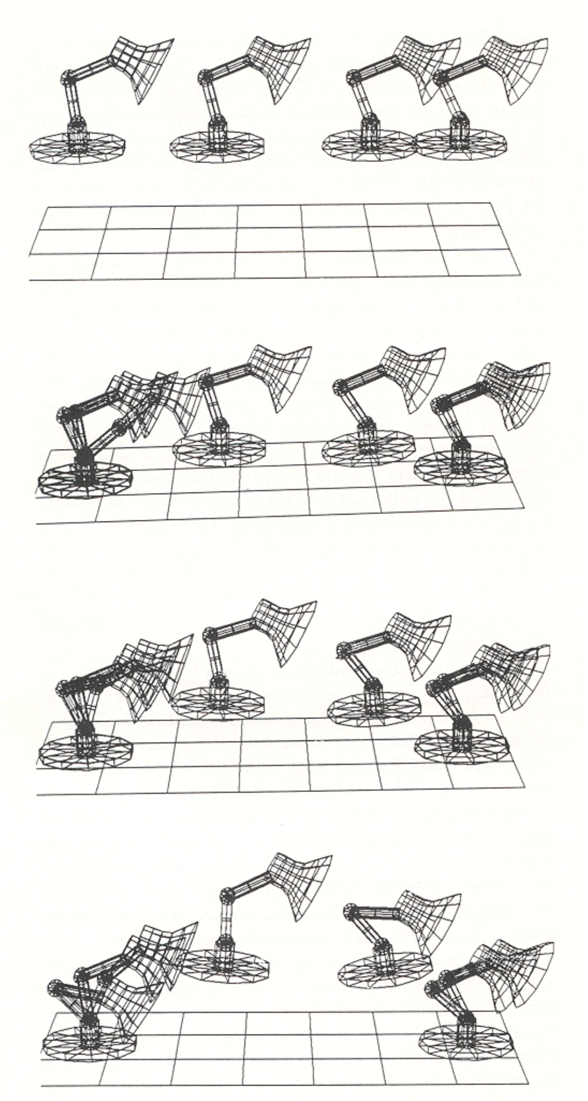
  - Suppose we want to move from a to b within t_0 to t_1 with minimum jet fuel:

$$\text{Minimize} \int_{t_0}^{t_1} \left| f(t)^2 \right| dt \text{ subject to } x(t_0)=a \text{ and } x(t_1)=b$$

Witkin & Kass `88

Spacetime Constraints

- Solve with iterative optimization methods



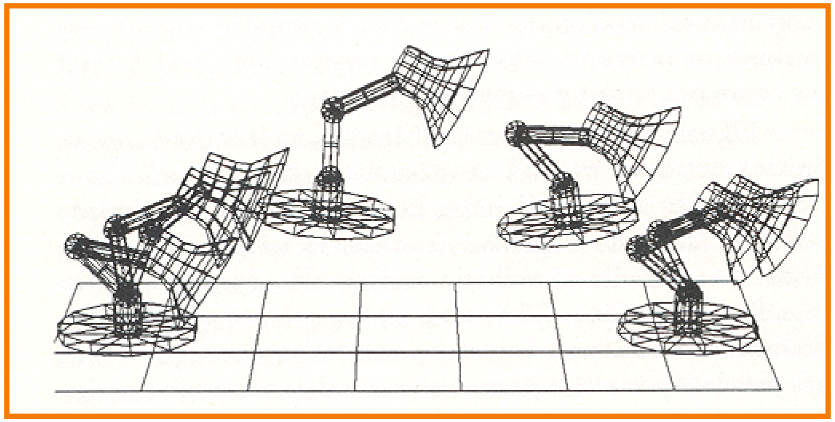
Witkin & Kass `88

Spacetime Constraints

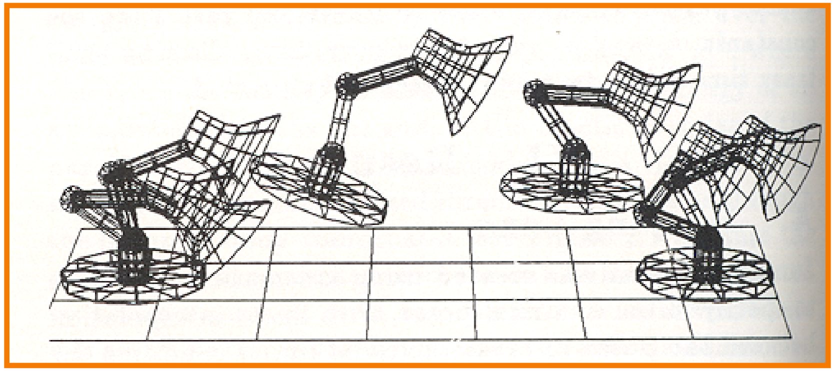
- Advantages:
  - Free animator from having to specify details of physically realistic motion with spline curves
  - Easy to vary motions due to new parameters and/or new constraints
- Challenges:
  - Specifying constraints and objective functions
  - Avoiding local minima during optimization

Spacetime Constraints

- Adapting motion:



Original Jump

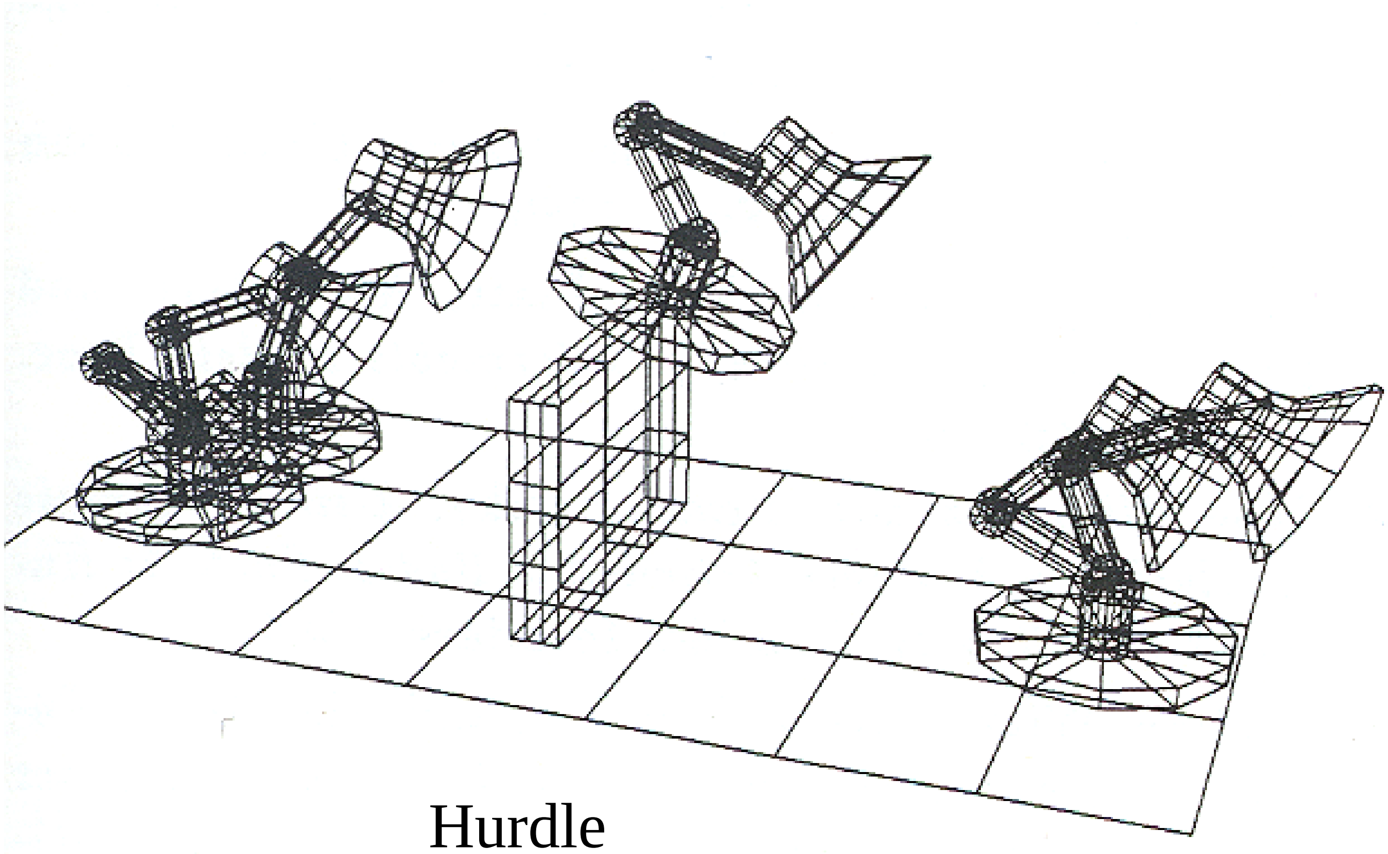


Heavier Base

Witkin & Kass `88

Spacetime Constraints

- Adapting motion:

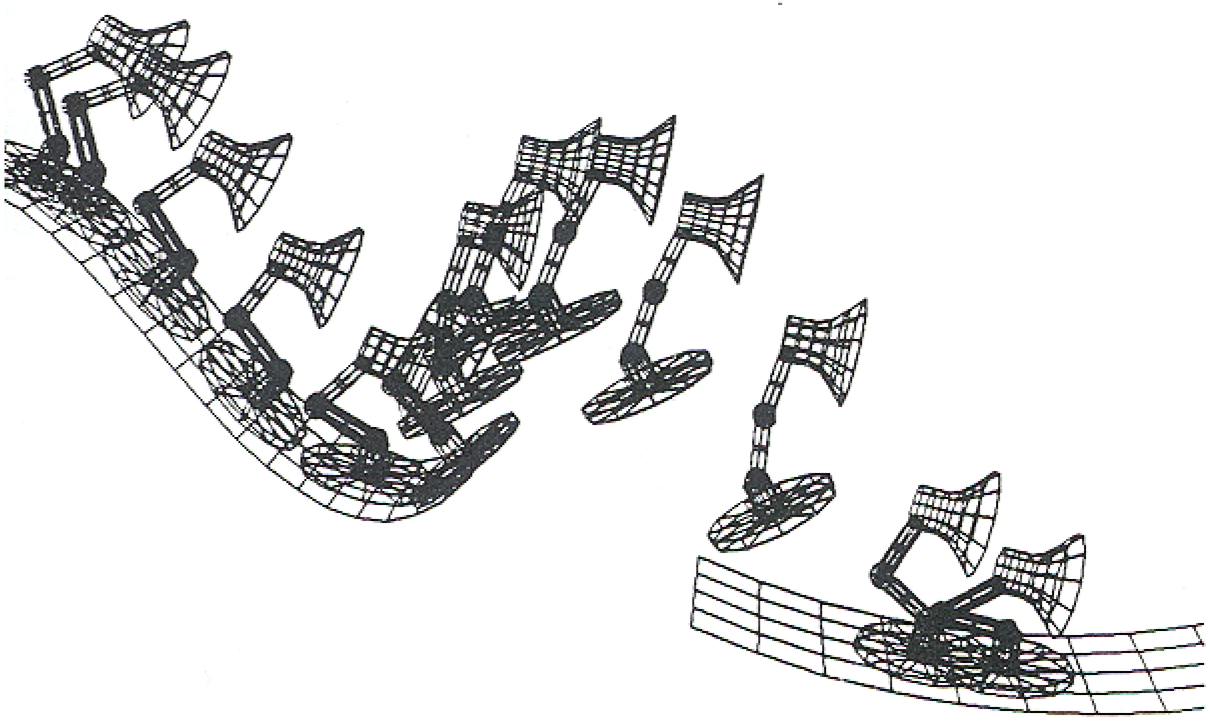


Hurdle

Witkin & Kass `88

Spacetime Constraints

- Adapting motion:



Ski Jump

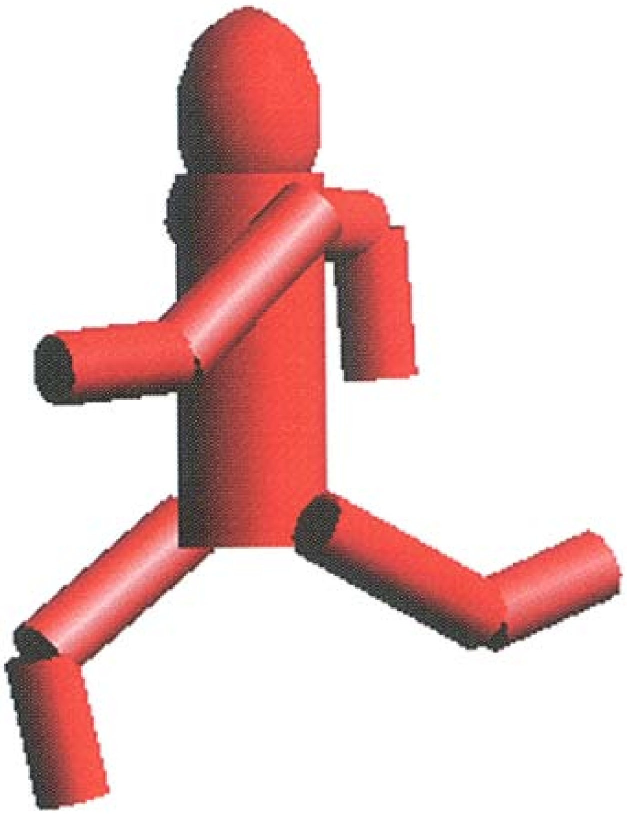
Witkin & Kass `88

Spacetime Constraints

- Advantages:
  - Free animator from having to specify details of physically realistic motion with spline curves
  - Easy to vary motions due to new parameters and/or new constraints
- Challenges:
  - Specifying constraints and objective functions
  - Avoiding local minima during optimization

Outline

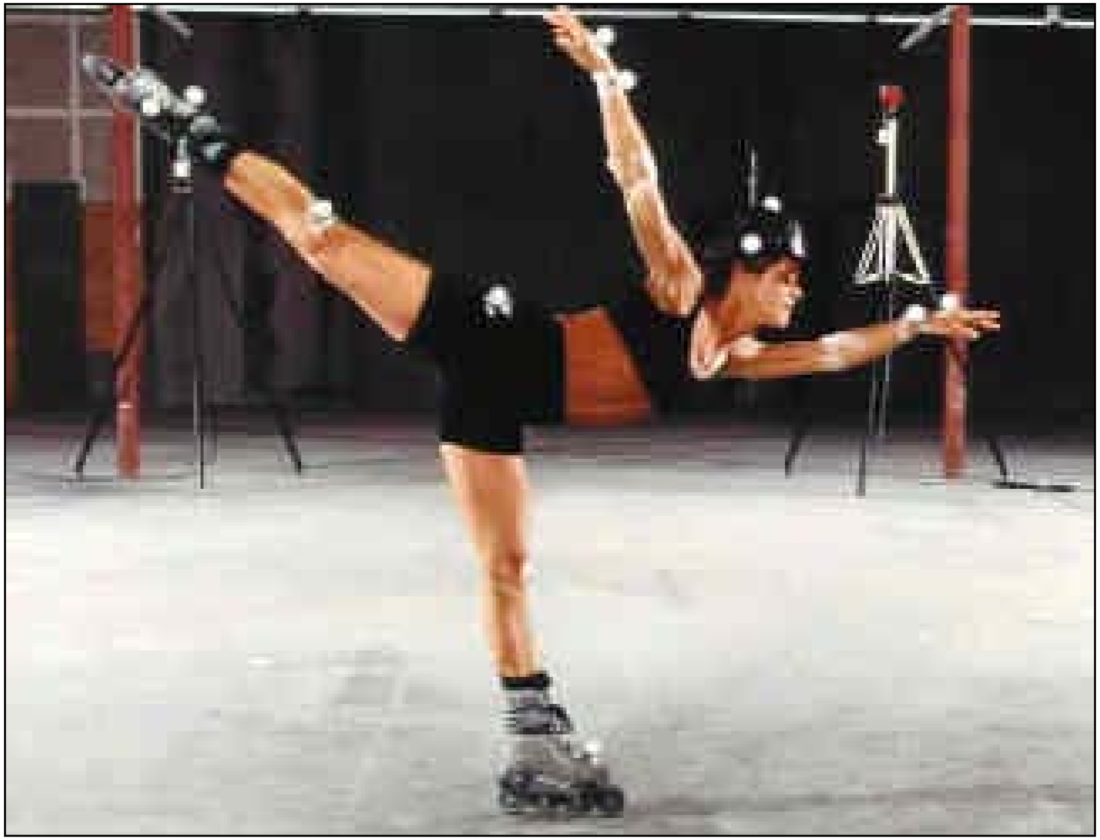
- Kinematics
- Dynamics
- ➢ Motion capture



Angel Plate 1

Motion Capture

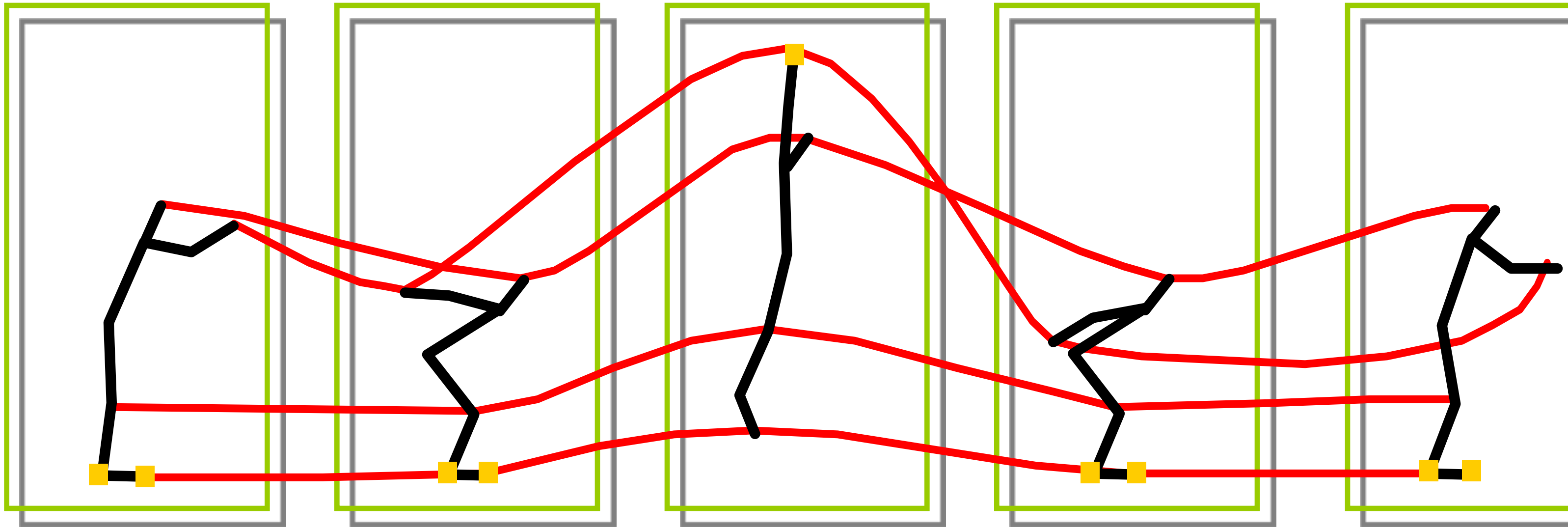
- Measure motion of real characters and then simply "play it back" with kinematics



Gleicher

Motion Capture

- Measure motion of real characters and then simply “play it back” with kinematics



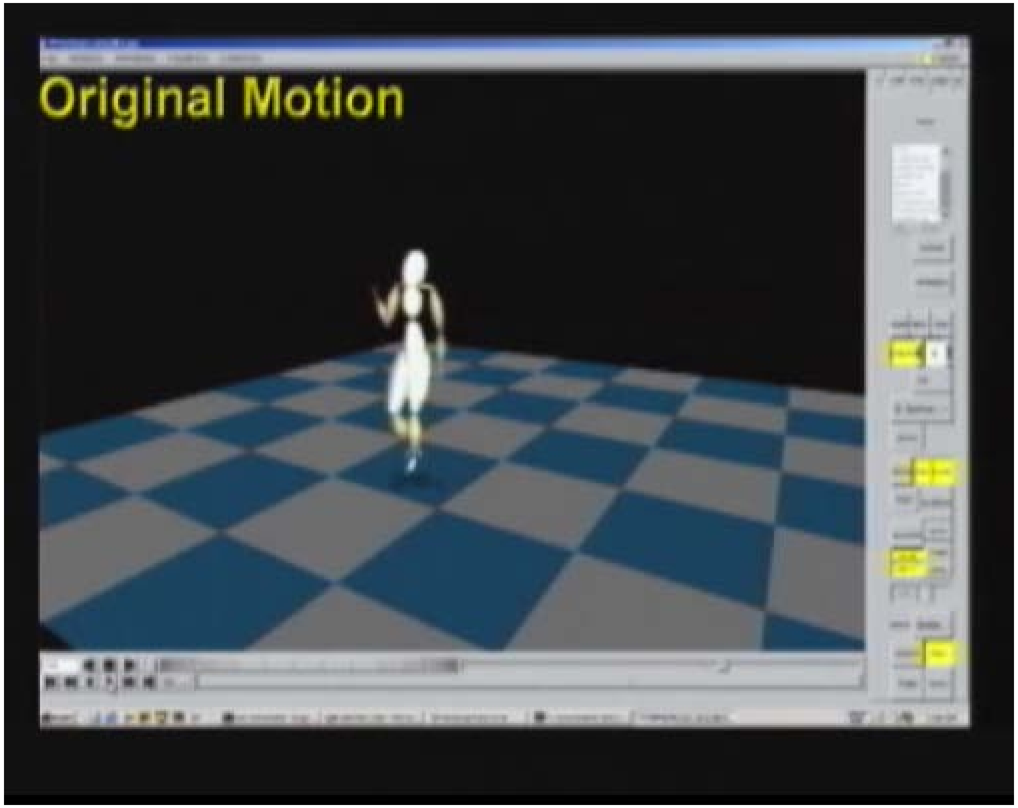
Captured Motion

Motion Capture

- Advantage:
  - Physical realism
- Challenge:
  - Animator control

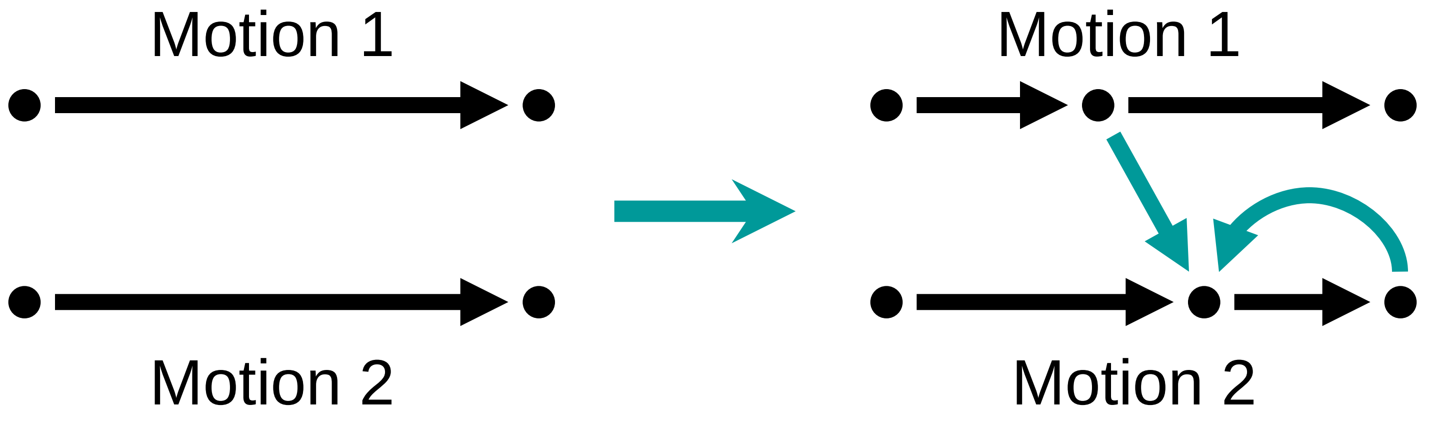
Motion Capture

- Editing motion:



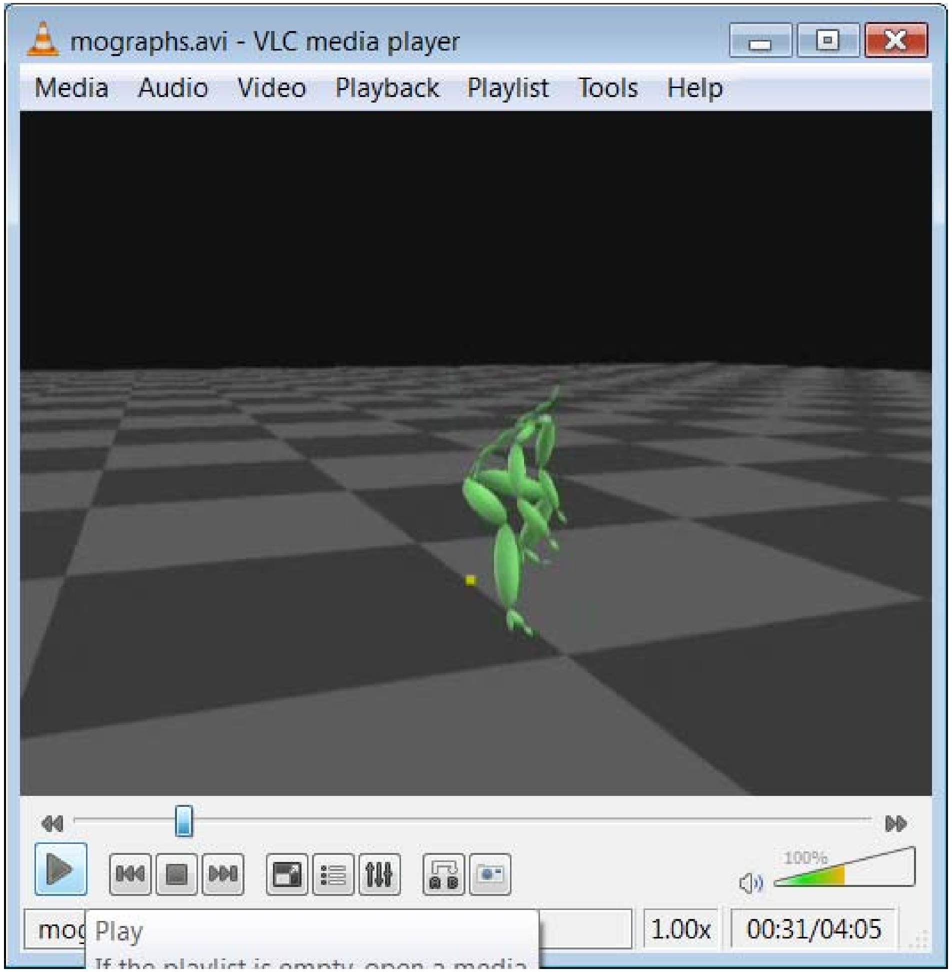
Motion Capture

- Motion graphs:



Motion Capture

- Motion graphs:

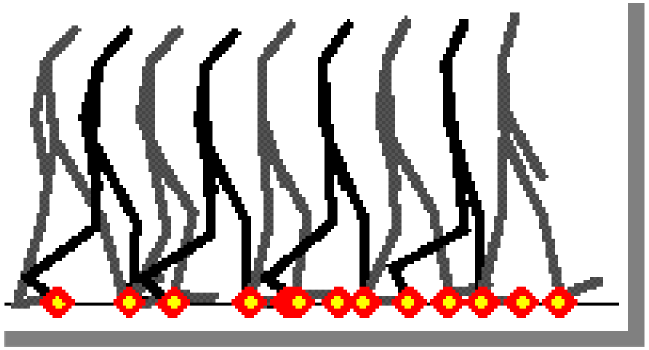


Kovacs & Gleicher

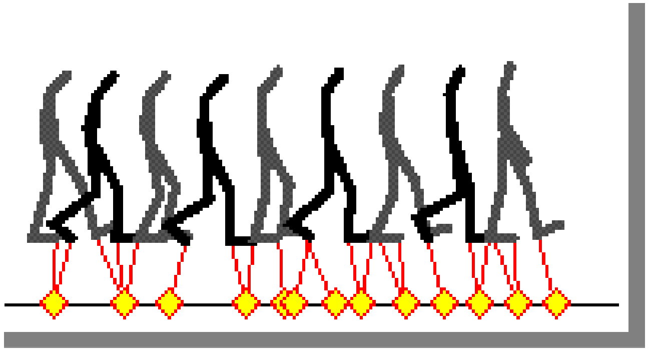
Motion Capture

- Retargeting motion:

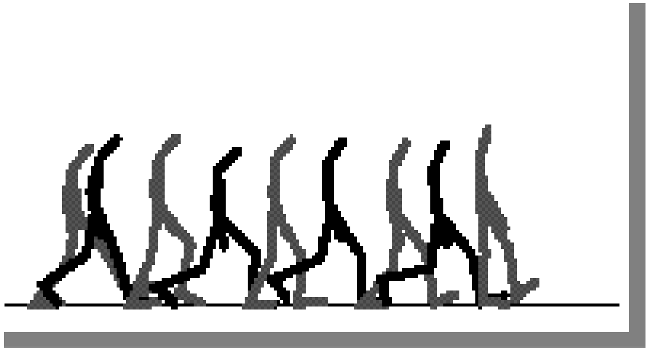
Original motion data + constraints:



New character:



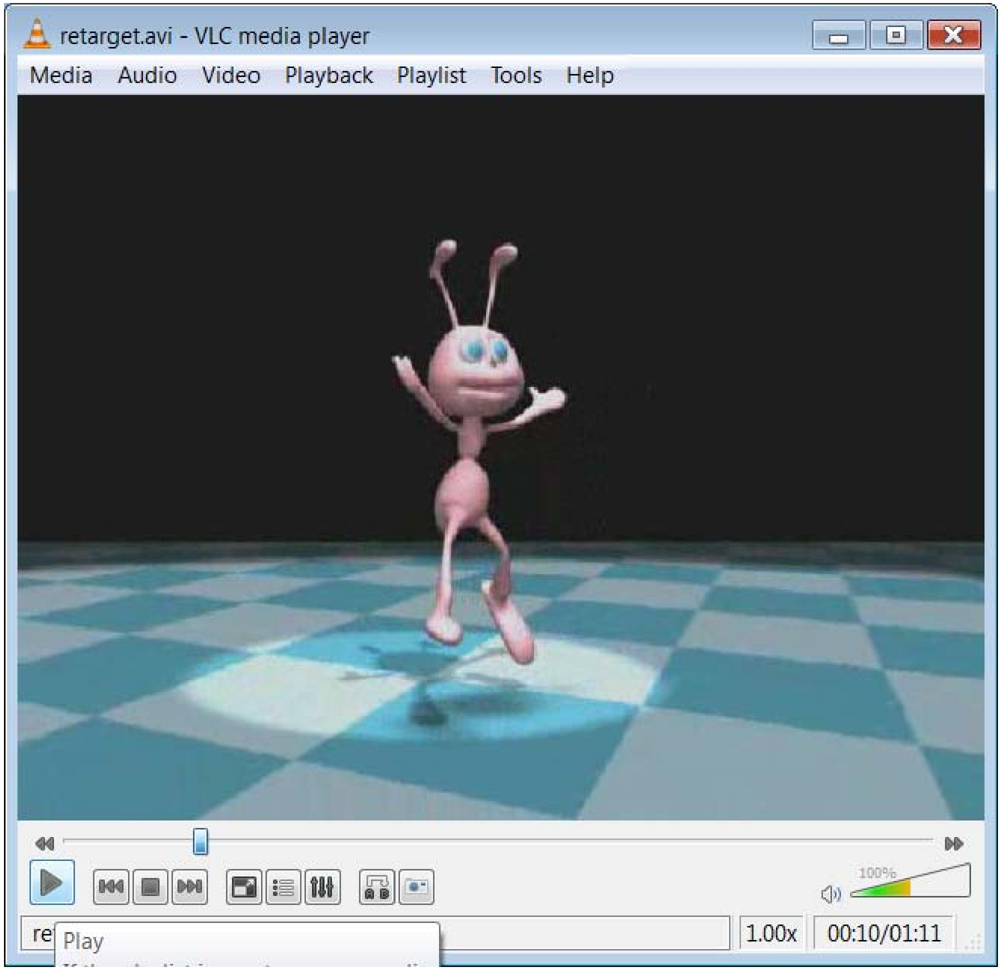
New motion data:



Gleicher

Motion Capture

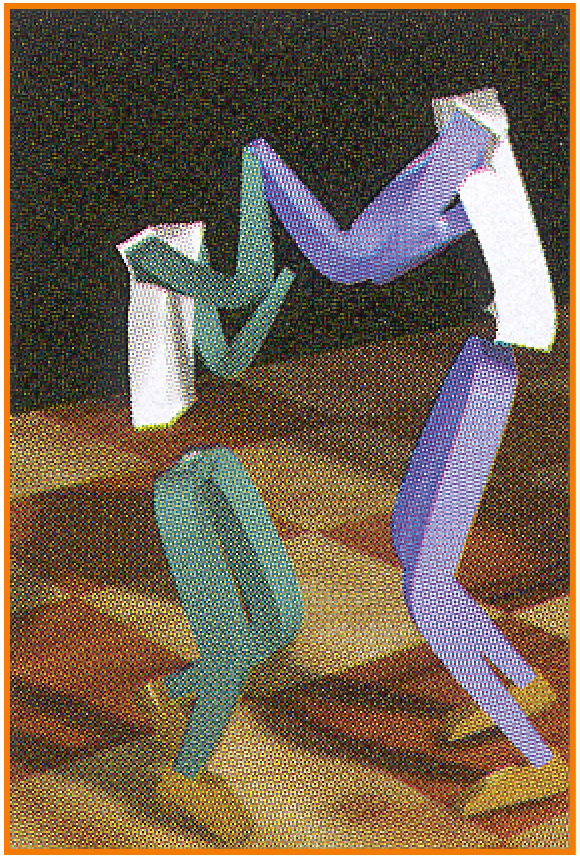
- Retargeting motion:



Gleicher

Motion Capture

- Morphing motion:



Summary

- Kinematics
  - Animator specifies poses (joint angles or positions) at keyframes and computer determines motion by kinematics and interpolation
- Dynamics
  - Animator specifies physical attributes, constraints, and starting conditions and computer determines motion by physical simulation
- Motion capture
  - Compute captures motion of real character and provides tools for animator to edit it