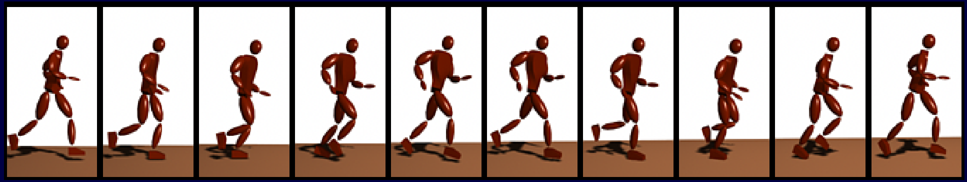
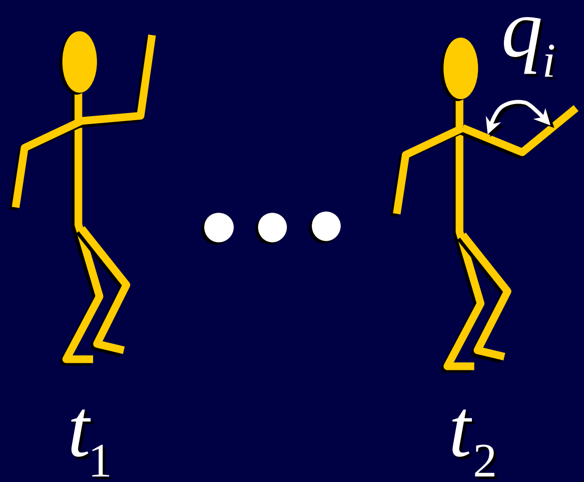
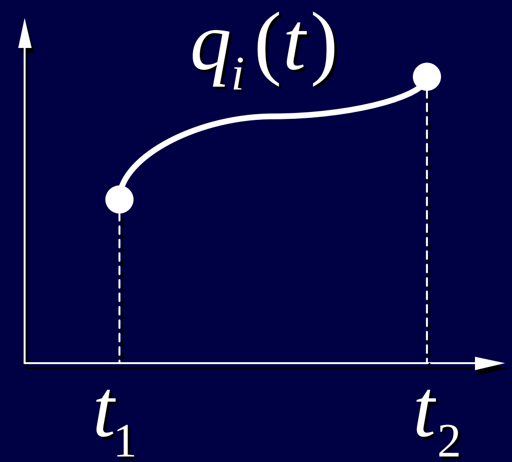
Physically Based Motion Transformation



Digital Character Animation

- ***Character*** – animated object with a number of degrees of freedom (DOFs)
- ***Motion*** – set of functions **q**(t) depicting how each DOF changes through time





The Animation Problem

Automatic generation of expressive/realistic motion that achieves a given set of tasks

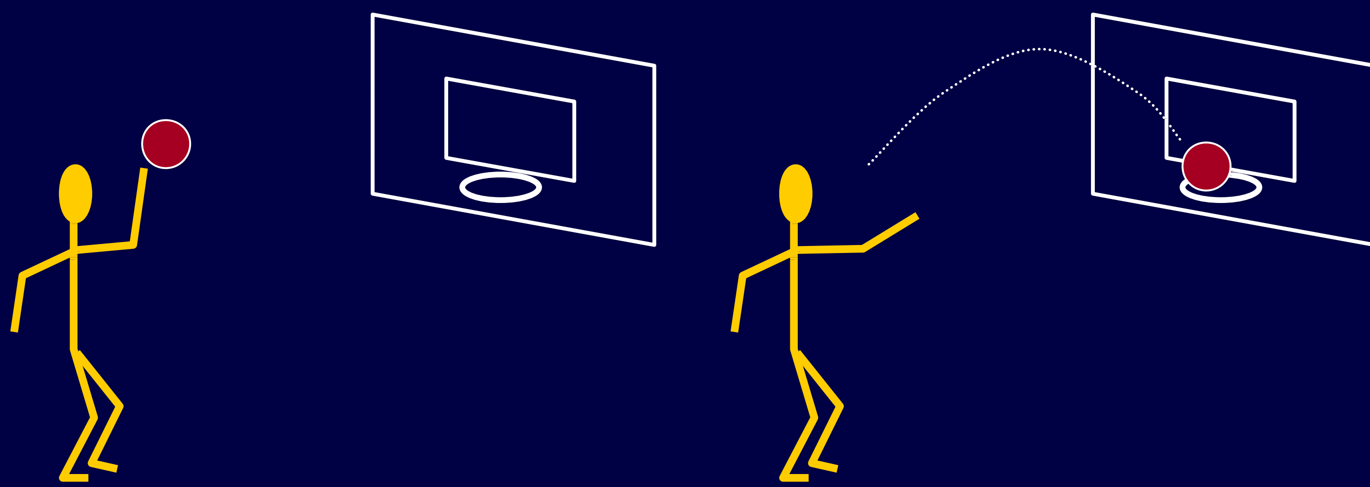
- An open problem
- Realism vs. control tradeoff

Physically-based Methods

- Forward simulation [Baraff]
  - Highly realistic
  - Simulated character very hard to control
- Controllers [Raibert, Hodgins, Ngo, van de Pane]
  - Fast motion generation once controllers are computed
  - No set rules on controller generation

Spacetime Constraints

- Provide both realism and control
- Downside
  - Methods do not scale up
  - Sensitivity to the initial position



Captured Motion

- Sampled DOFs through time gathered from the real world
- Rich and realistic
- Hard to edit

Motion Warping

- Set poses which the warped motion should interpolate
- Set time constraints and solve for the minimum curve deviation Interpolate the space constructed by a few sample motion capture sequences

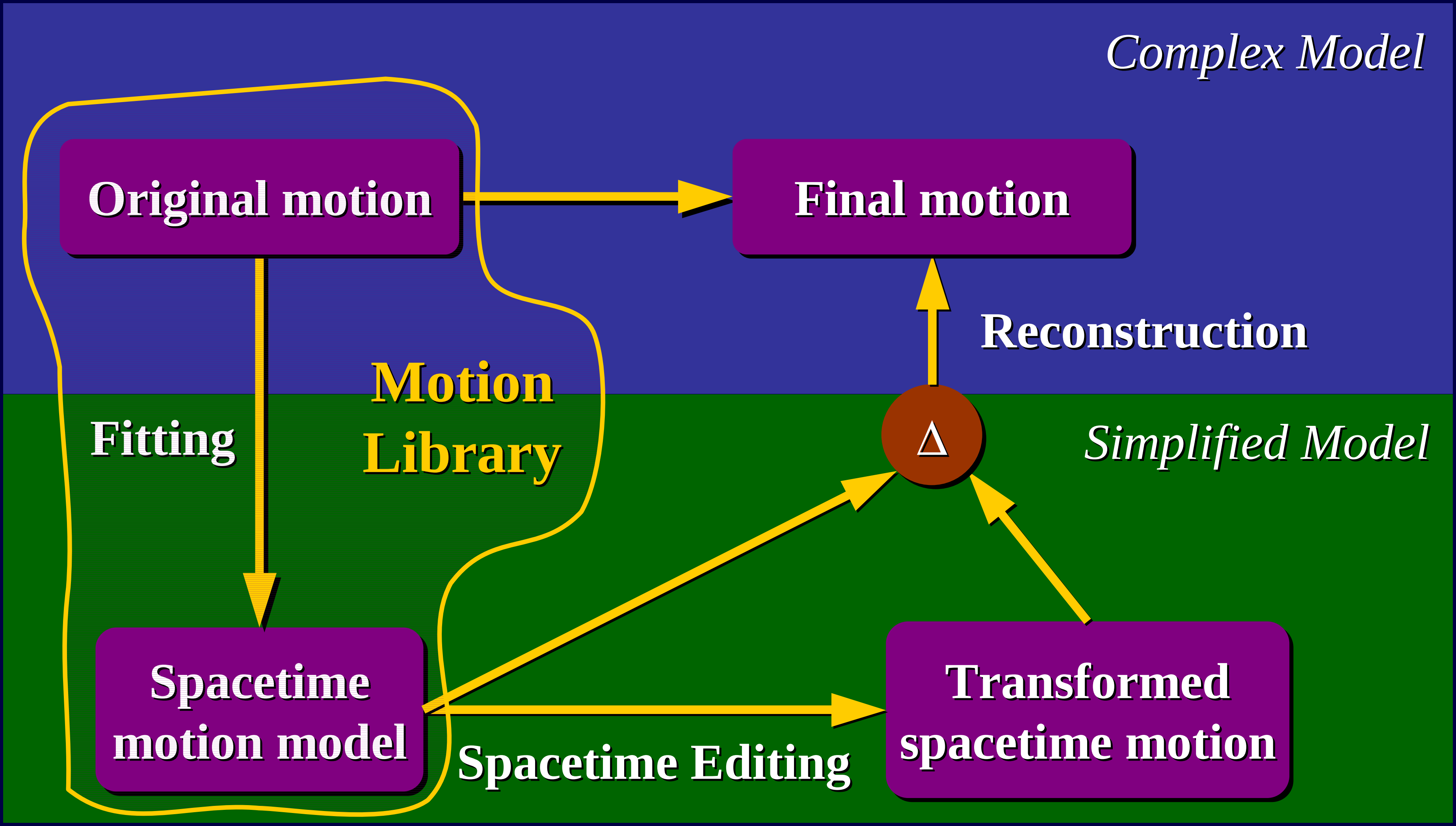
High Level Control

- Get a limp walk by making one leg stiff
- Reduce gravity to get a “moon walk”
- Change the position and timing of foot placements
- Make a “quiet” run by reducing the floor impact forces

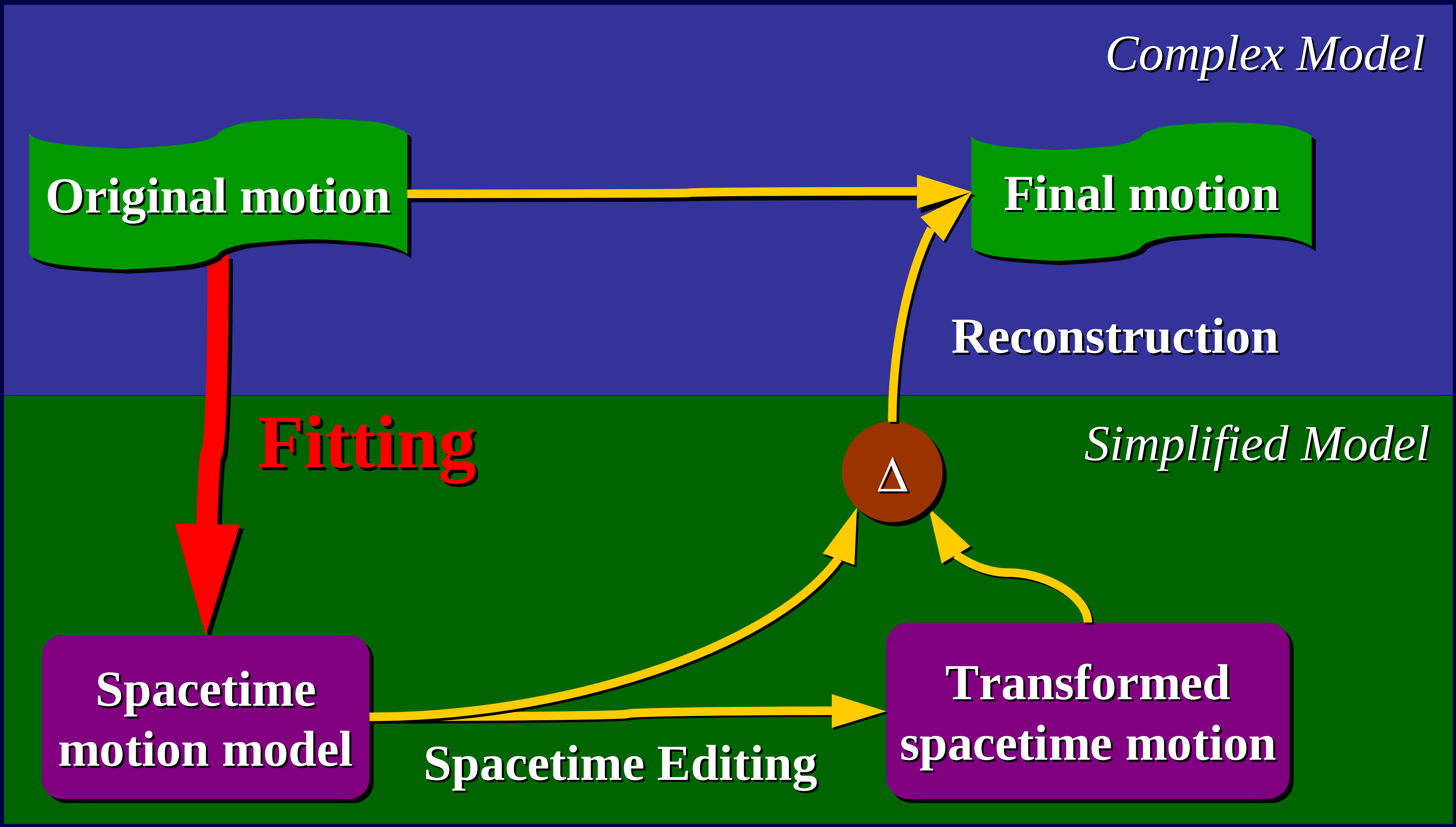
The New Approach

- Transform existing motion
- Spacetime constraints formulation
- Simplified character representation
- Get the best of both worlds:
  - Expressiveness of captured data
  - Controllability of the spacetime model

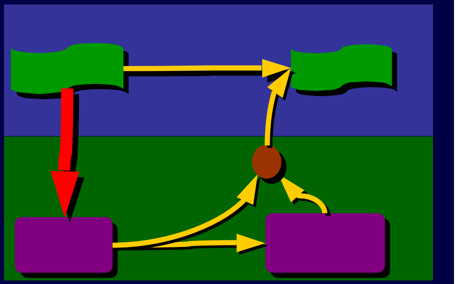
Outline


Complex Model
Original motion
Final motion
Reconstruction
Motion Library
Fitting
Δ
Simplified Model
Spacetime motion model
Transformed spacetime motion
Spacetime Editing

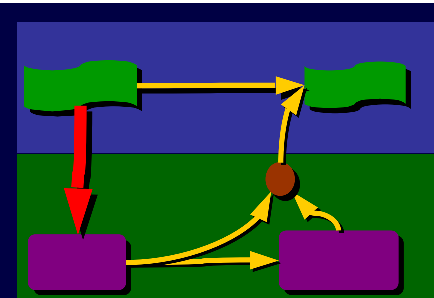
Outline


Complex Model
Original motion
Final motion
Reconstruction
Fitting
Simplified Model
Δ
Spacetime motion model
Spacetime Editing
Transformed spacetime motion

Model Fitting

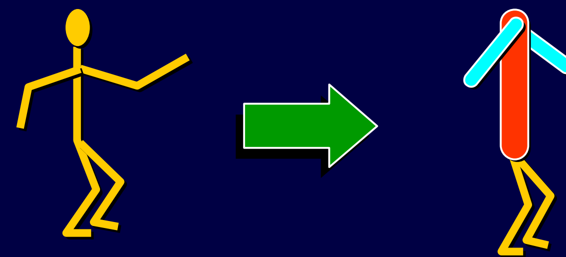


- Two phases:
  - Simplify character kinematics
  - Use input motion to construct a spacetime motion model

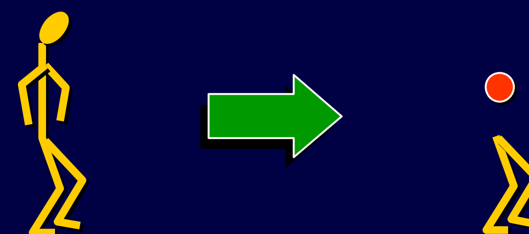


Simplified Kinematics

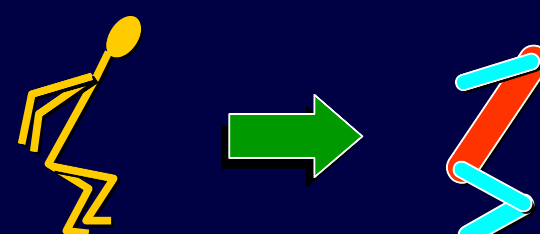
- Remove irrelevant DOFs



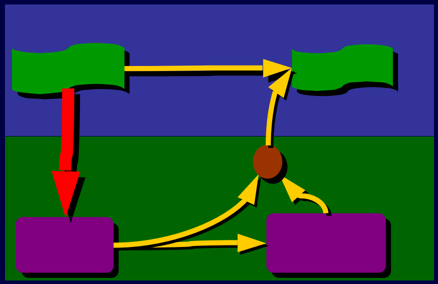
- Reduce ***passive*** body structure to mass points

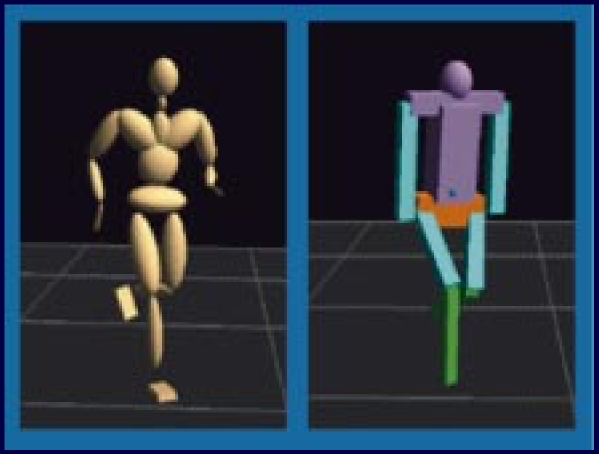


- Exploit symmetric movement of limbs

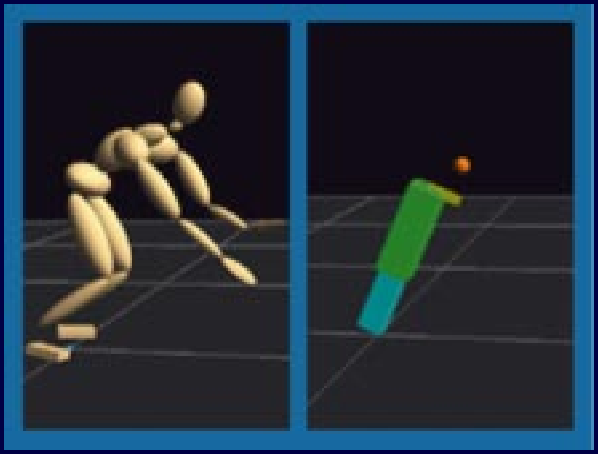


Simplified Kinematics



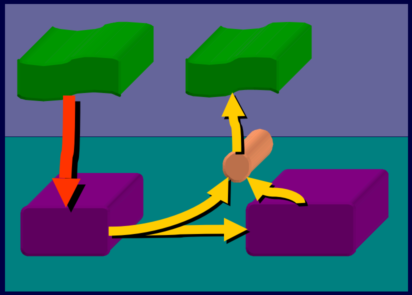


Human Run

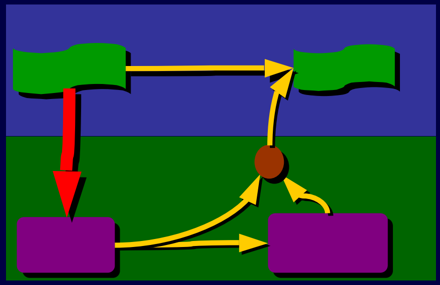


Human Jump

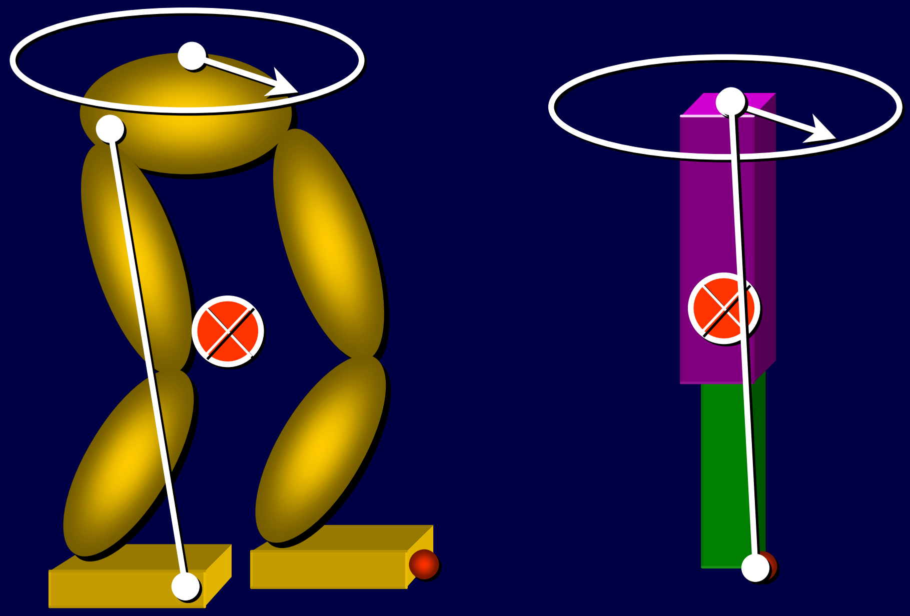
Motion Fitting



- Handle – a property that correlates the original and simplified model
- Must have enough handles to fully determine simplified model configuration



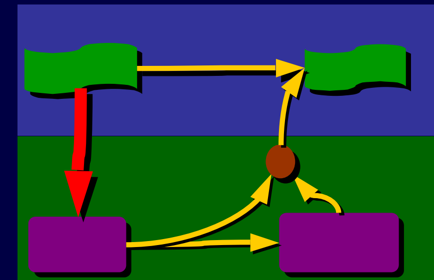
Handle Examples



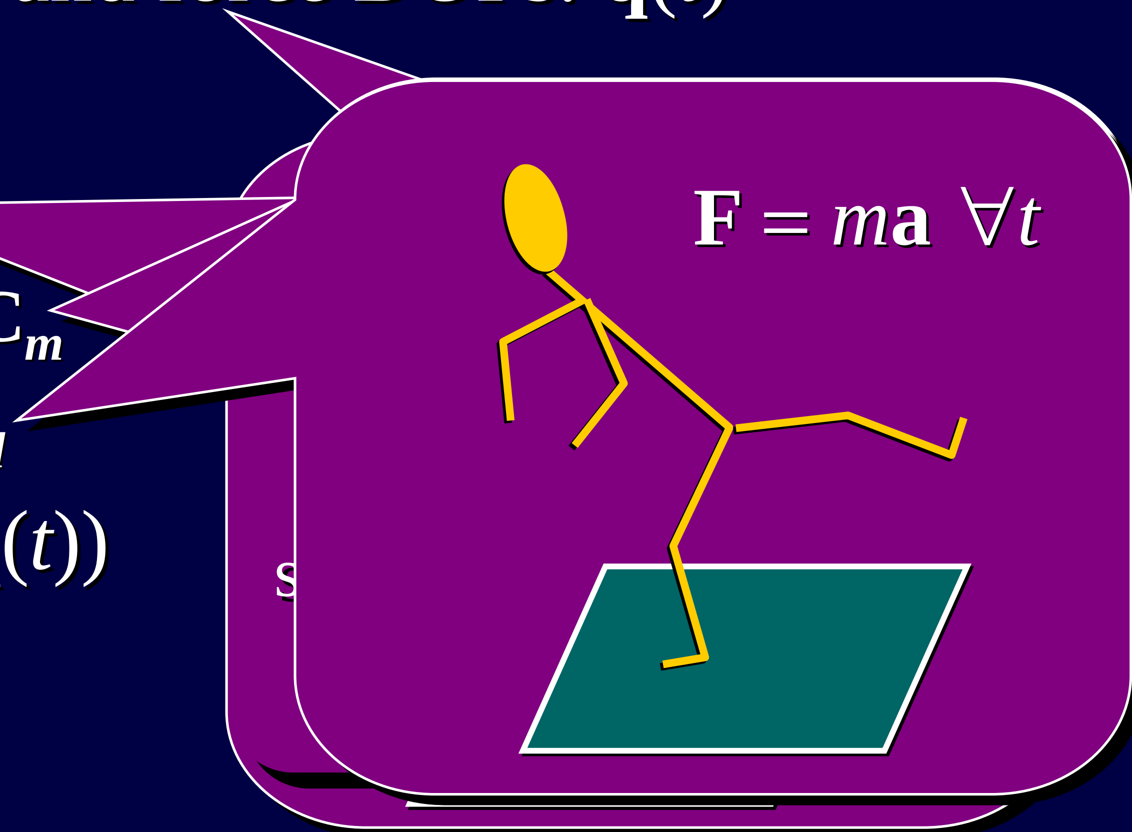
Distance Handle

Body Point Handle

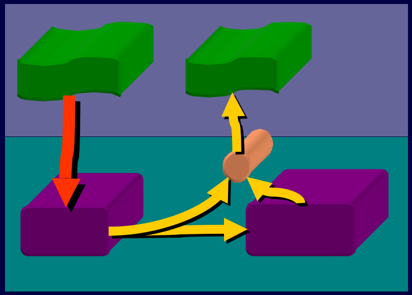
Motion Synthesis As Constrained Optimization



- Body, muscle and force DOFs: $\mathbf{q}(t)$
- Constraints:
  - Pose $\mathbf{C}_p$
  - Mechanical $\mathbf{C}_m$
  - Dynamics $\mathbf{C}_d$
- Objective $E(\mathbf{q}(t))$



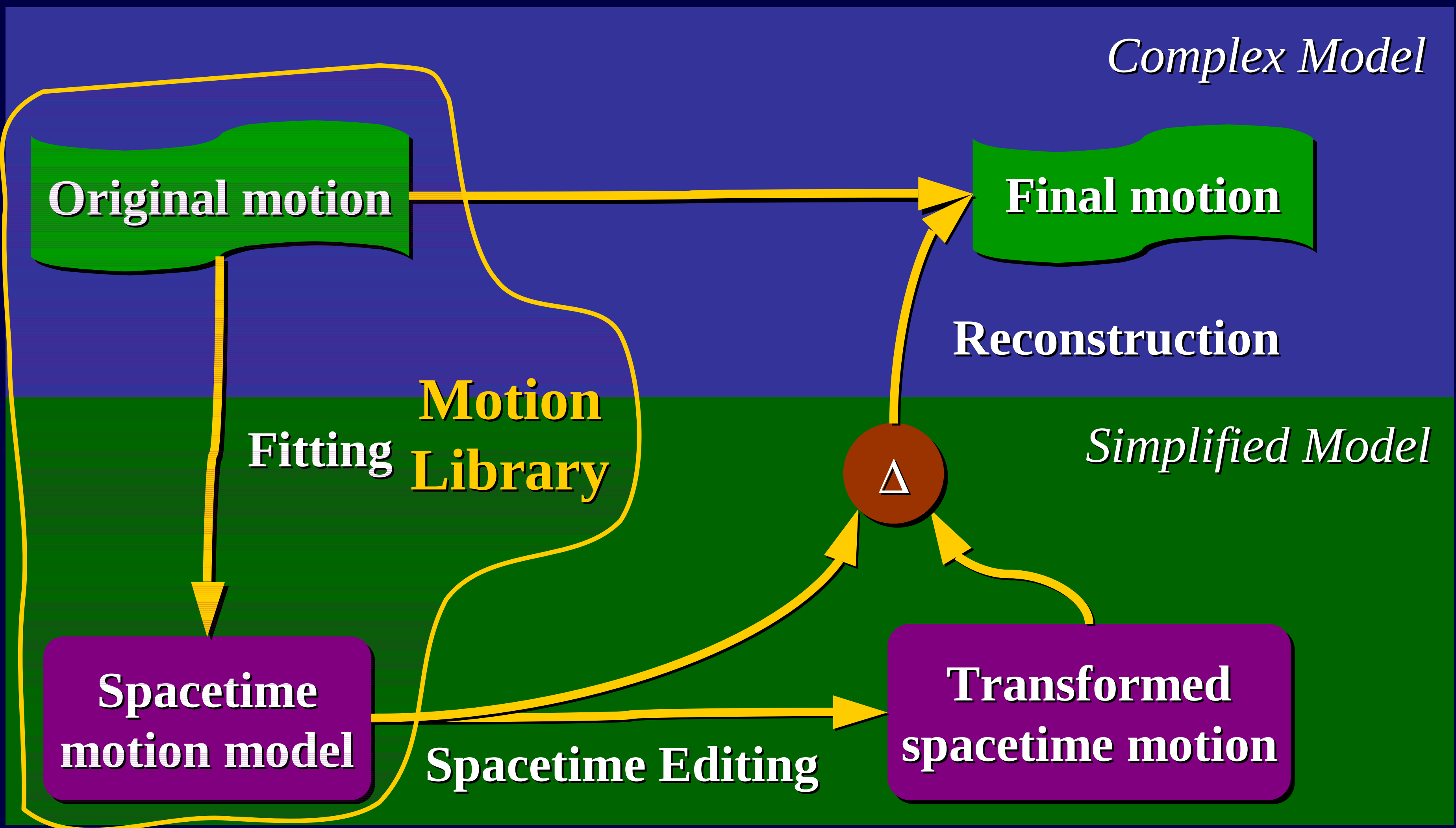
Spacetime Model Fitting



- Biological data: mass distribution, muscles
- Use ***handles*** to create "best-guess" motion
- Specify constraints essential for given motion (e.g. foot placements)
- Use simple objective: smooth muscles

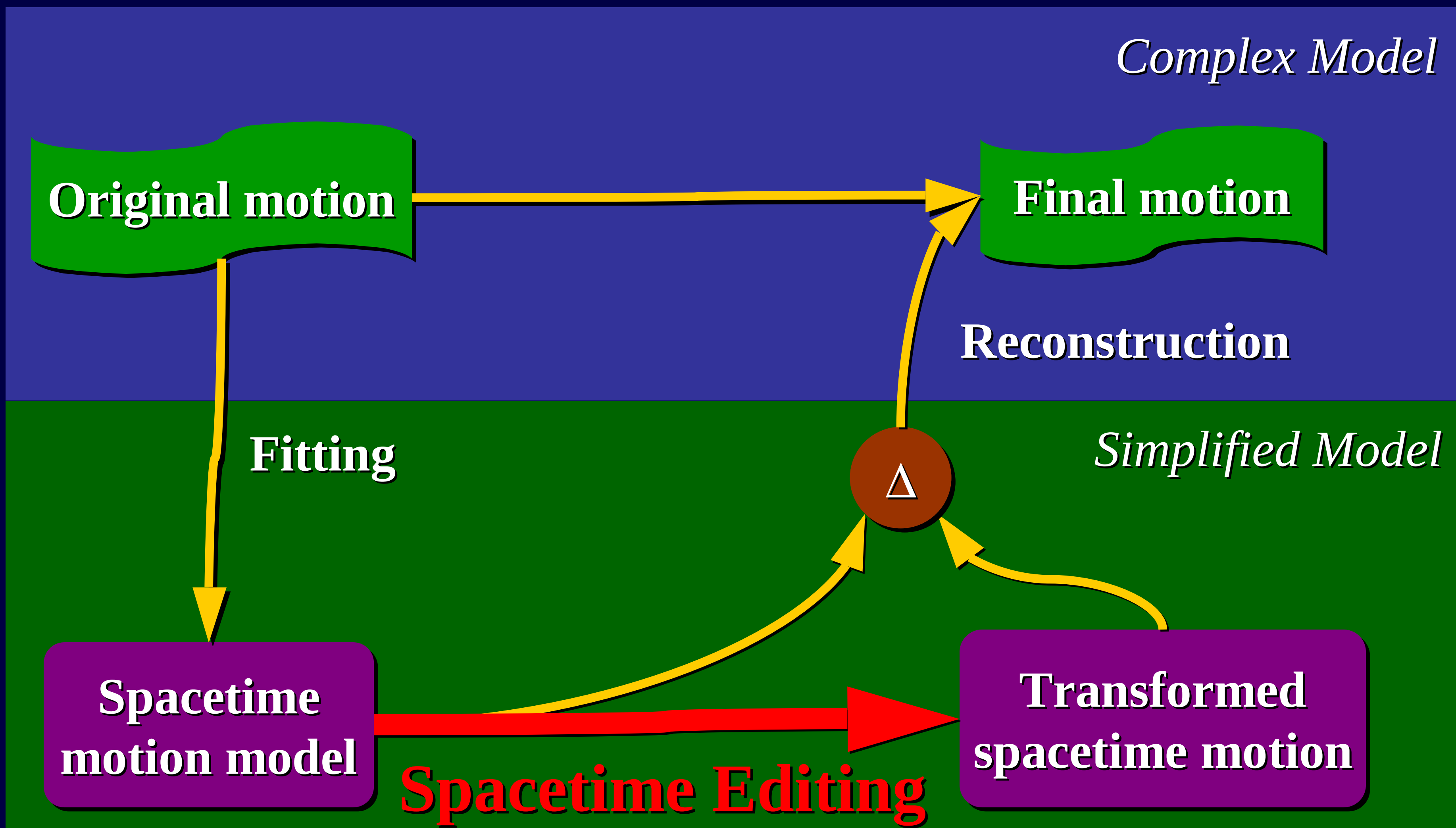
$$E(\mathbf{q}) = \ddot{\mathbf{q}}^2$$

Outline



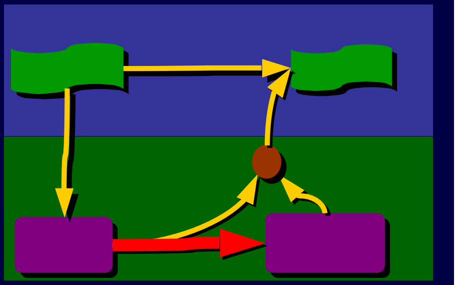
Complex Model
Original motion
Final motion
Reconstruction
Motion Library
Fitting
Simplified Model
Δ
Spacetime motion model
Transformed spacetime motion
Spacetime Editing

Outline



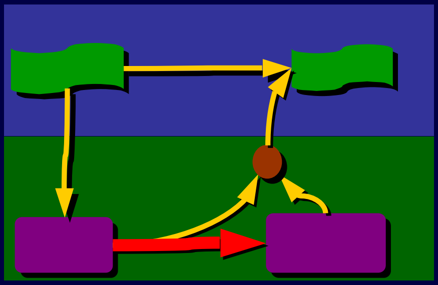
Complex Model
Original motion
Final motion
Reconstruction
Fitting
Simplified Model
Δ
Spacetime
motion model
Transformed
spacetime motion
Spacetime Editing

Spacetime Editing



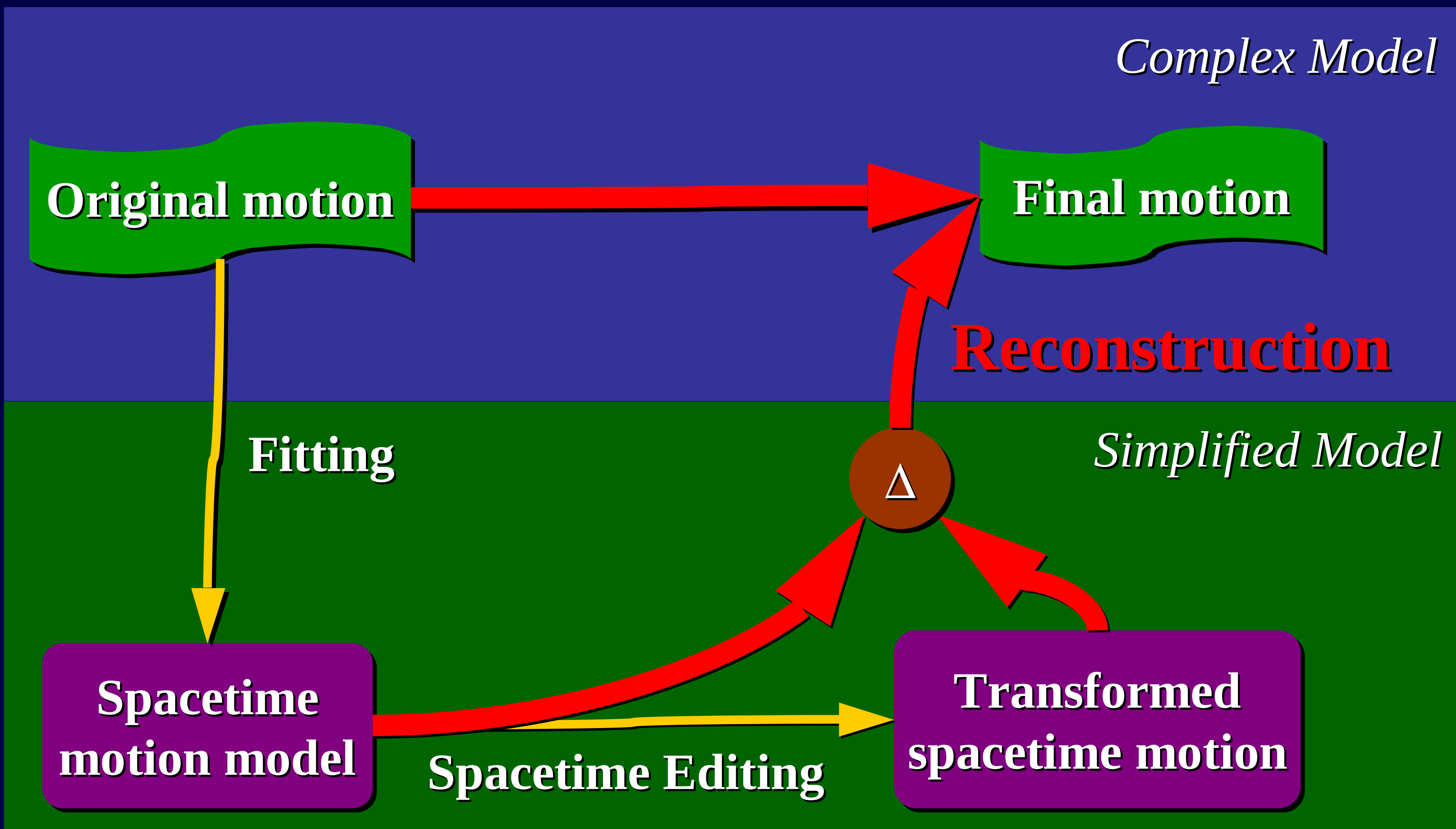
- Change pose and environment constraints
  - Foot placement and timing
  - Introduce a new obstacle
- Change the objective function
  - Minimize floor impact forces
  - Make dynamic balance more important

Spacetime Editing

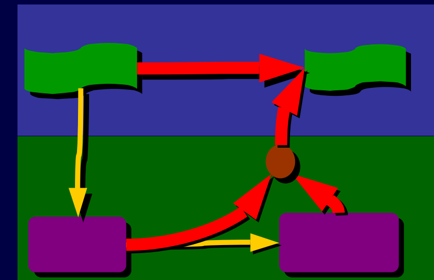


- Change explicit character parameters
  - Short leg
  - Redistribute mass
  - Modify muscle characteristic
  - Gravity

Outline


Complex Model
Original motion
Final motion
Reconstruction
Fitting
Δ
Simplified Model
Spacetime motion model
Spacetime Editing
Transformed spacetime motion

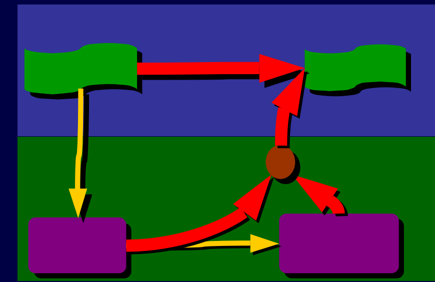
Motion Reconstruction



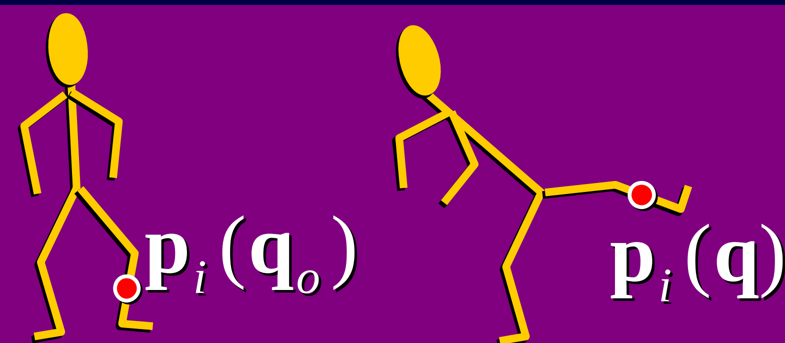
- Three different handle sets
  - Original motion handles $\mathbf{h}(\mathbf{q}_o)$
  - Spacetime fit handles $\mathbf{h}(\mathbf{q}_s)$
  - Transformed spacetime handles $\mathbf{h}(\mathbf{q}_t)$
- Compute final motion handles

$$\mathbf{h}(\mathbf{q}_f) = \mathbf{h}(\mathbf{q}_o) + \left(\mathbf{h}(\mathbf{q}_t) - \mathbf{h}(\mathbf{q}_s)\right)$$

Minimum Displaced Mass Objective

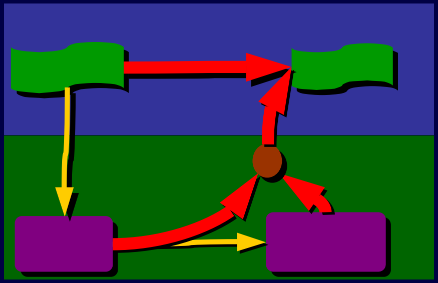


- $E_{dm}(\mathbf{q}_o, \mathbf{q})$ evaluates ***total displaced mass*** when moving a character from pose $\mathbf{q}_o$ to pose $\mathbf{q}$



$$E_{dm} = \iiint_i \mu_i \left(\mathbf{p}_i(\mathbf{q}_o) - \mathbf{p}_i(\mathbf{q})\right)^2 dx\, dy\, dz$$

Reconstruction Algorithm

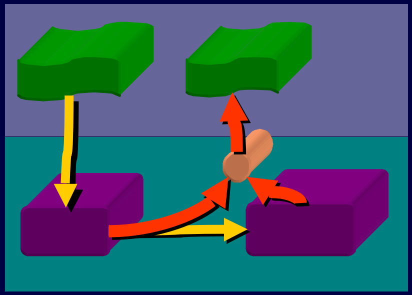


- For each time *t* solve

$$\underset{\mathbf{q}_f}{\text{minimize}} \quad E_{dm}\left(\mathbf{q}_o, \mathbf{q}_f\right)$$

$$\text{subject to} \quad \mathbf{h}(\mathbf{q}_f) = \mathbf{h}(\mathbf{q}_o) + \left(\mathbf{h}(\mathbf{q}_t) - \mathbf{h}(\mathbf{q}_s)\right)$$

Alternative Reconstruction Algorithm



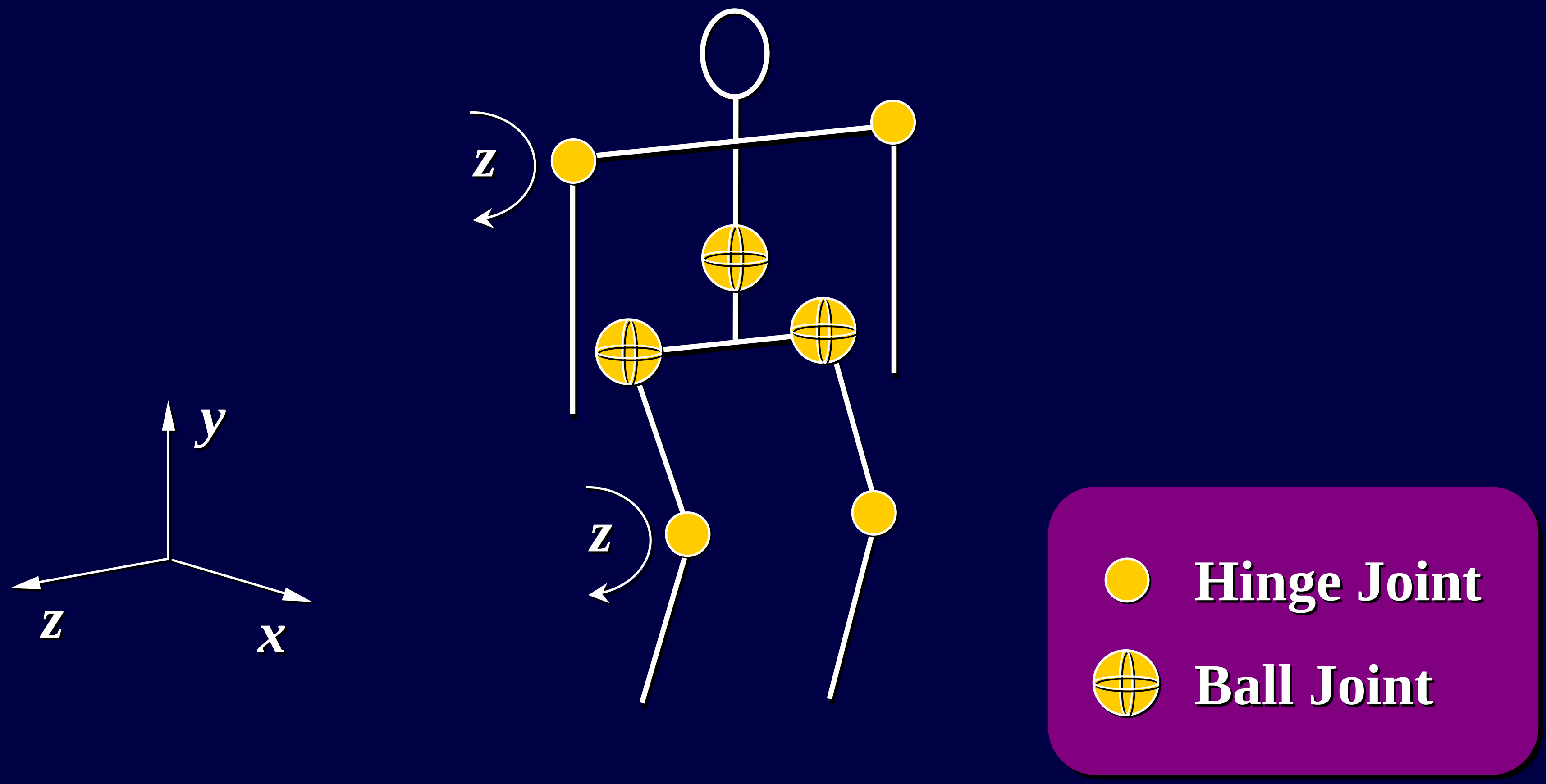
- For each time *t* solve

$$\underset{\mathbf{q}_f}{\text{minimize}} \quad \begin{array}{c} w_{dm} E_{dm}\left(\mathbf{q}_o, \mathbf{q}_f\right) + \\ w_h\left[\left(\mathbf{h}(\mathbf{q}_f) - \mathbf{h}(\mathbf{q}_o)\right) - \left(\mathbf{h}(\mathbf{q}_t) - \mathbf{h}(\mathbf{q}_s)\right)\right]^2 \end{array}$$

Example: Human Run

- Original model has 59 DOFs
- Simplified model has ***19*** DOFs
- Optimizations are done on one gait cycle
- Each optimization completes within 2 minutes

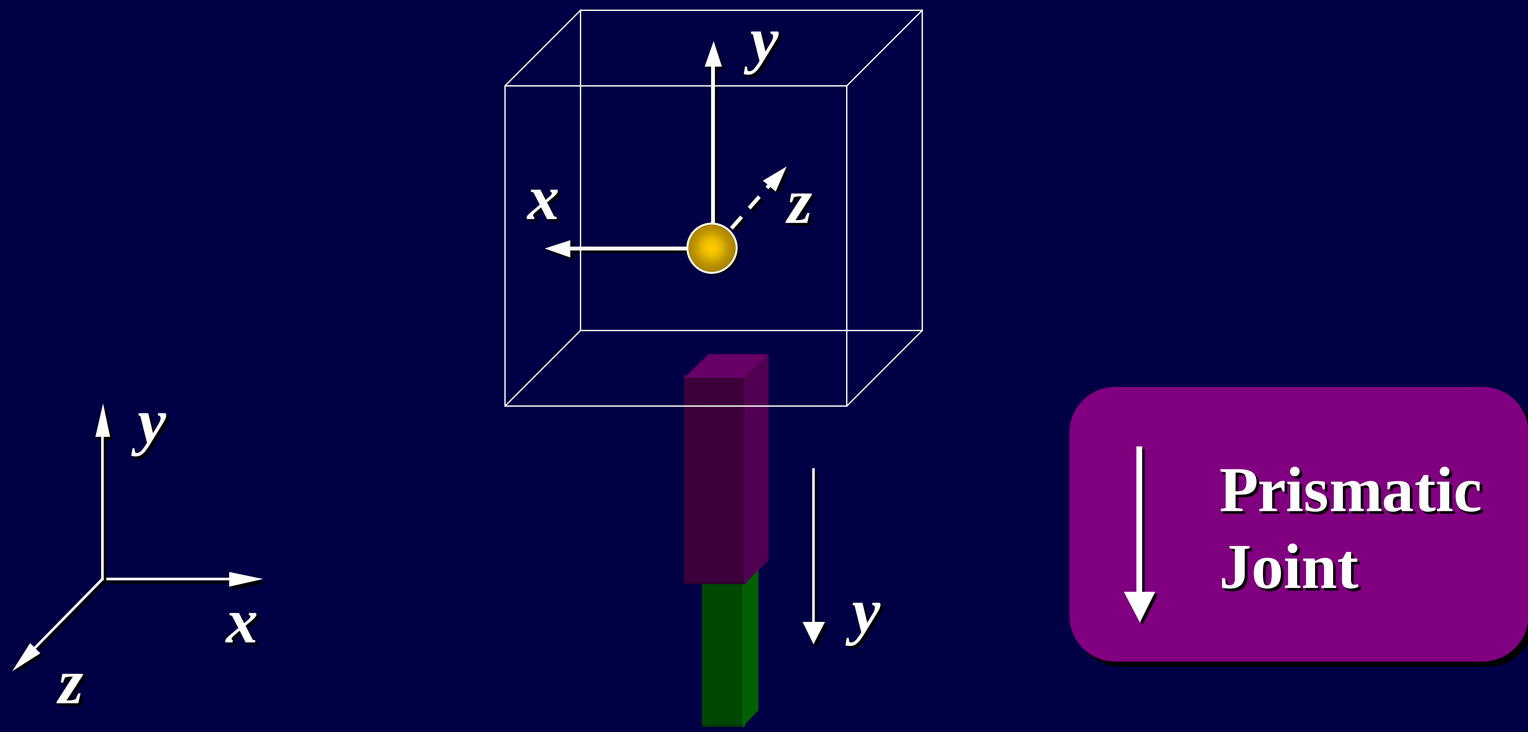
Biped



Example: Human Broad Jump

- Original model has 59 DOFs
- Simplified model has ***11*** DOFs
- Entire upper body reduced to a mass point
- No joint angle DOFs

Hopper



Future Work

- Optimal robots
- Extracting style
- Motion retargeting
- Building motion libraries
- Digital actors