

OptiMo: Optimization-Guided Motion Editing for Keyframe Character Animation

Yuki Koyama & Masataka Goto



- Powerful
- Used in computer science and engineering

- Powerful
- Used in computer science and engineering

WIKIPEDIA
The Free Encyclopedia

- Main page
- Contents
- Featured content
- Current events
- Random article
- Donate to Wikipedia
- Wikipedia store
- Interaction
- Help
- About Wikipedia
- Community portal
- Recent changes
- Contact page
- Tools
- What links here
- Related changes

Yuki Koyama

[Talk](#)
[Sandbox](#)
[Preferences](#)
[Beta](#)
[Watchlist](#)
[Contributions](#)
[Log out](#)

Article
Talk
Read
Edit source
View history

Mathematical optimization

From Wikipedia, the free encyclopedia

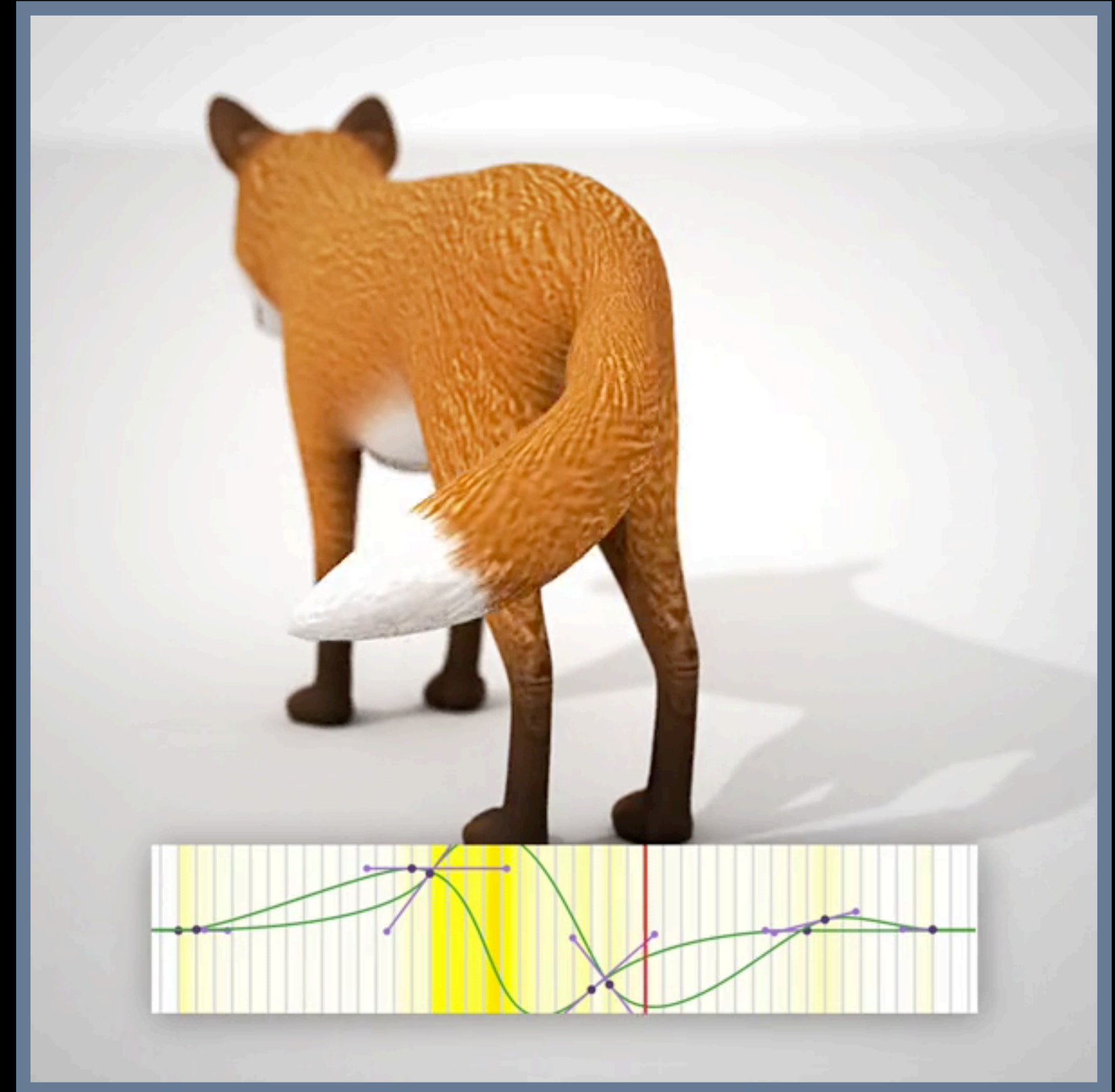
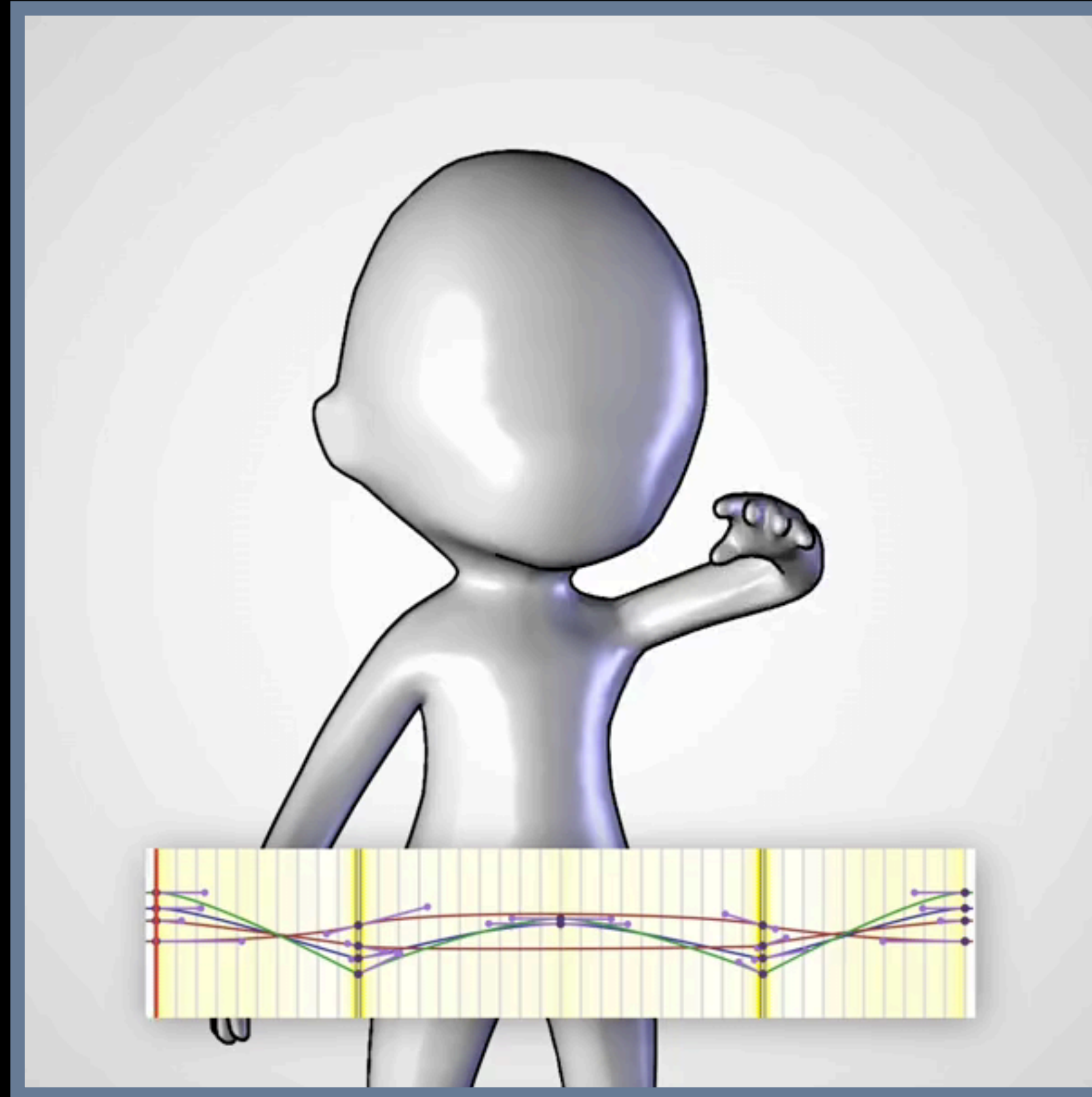
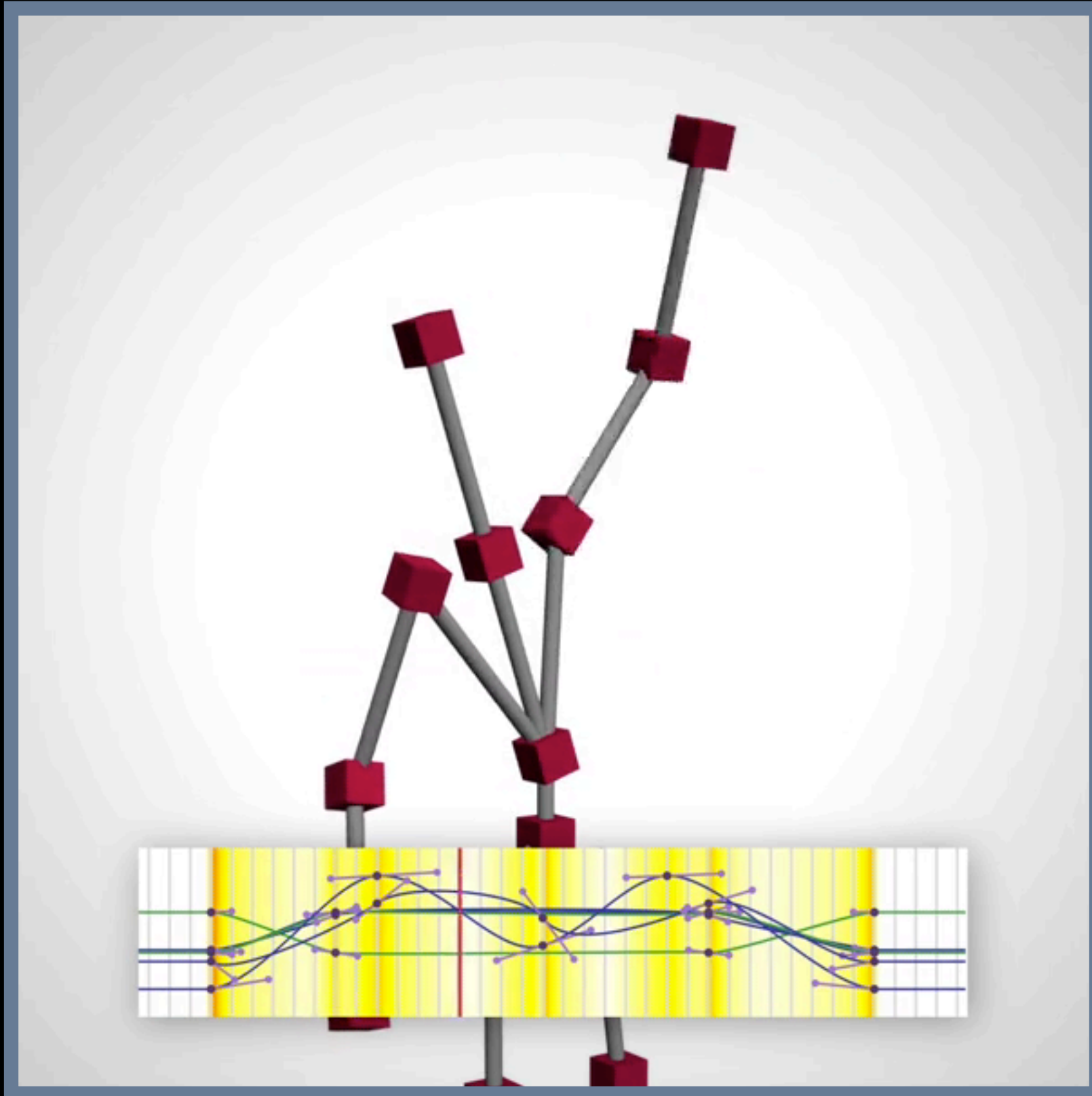
"*Mathematical programming*" redirects here. For the peer-reviewed journal, see [Mathematical Programming](#).
"Optimization" and *"Optimum"* redirect here. For other uses, see [Optimization \(disambiguation\)](#) and [Optimum \(disambiguation\)](#).

In [mathematics](#), [computer science](#) and [operations research](#), **mathematical optimization** or **mathematical programming**, alternatively spelled *optimisation*, is the selection of a best element (with regard to some criterion) from some set of available alternatives.^[1]

In the simplest case, an [optimization problem](#) consists of [maximizing or minimizing a real function](#) by systematically choosing [input values](#) from within an allowed set and computing the [value](#) of the function. The generalization of optimization theory and techniques to other formulations constitutes a large area of [applied mathematics](#). More generally, optimization includes finding "best available" values of some objective function given a defined [domain](#) (or input), including a variety of different types of objective functions and different types of domains.

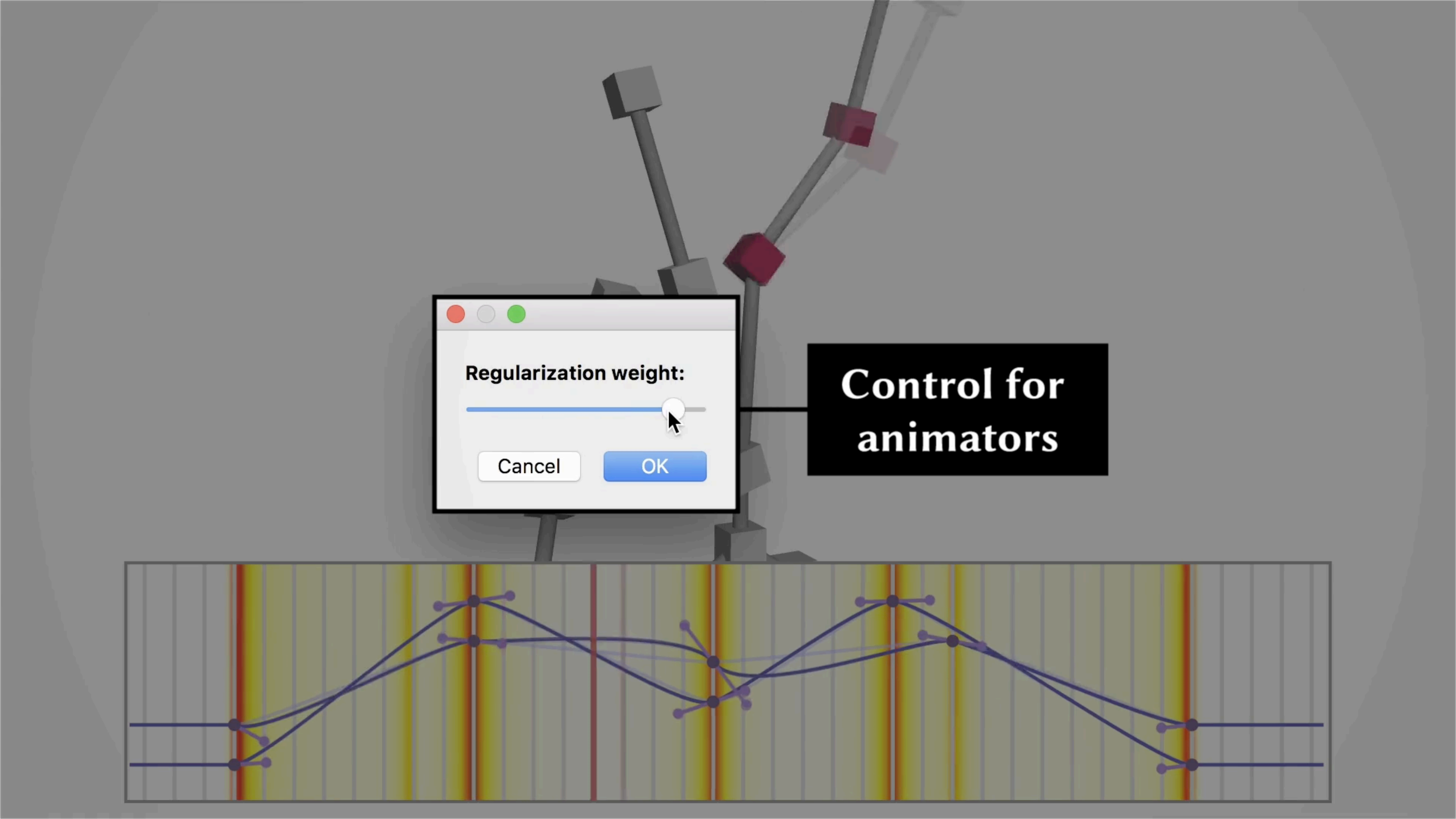
Graph of a [paraboloid](#) given by $z = -(x^2 + y^2) + 4$. The global maximum at $(x, y, z) = (0, 0, 4)$ is indicated by a blue dot.

https://en.wikipedia.org/wiki/Mathematical_optimization



How can animators utilize **optimization techniques** for motion editing effectively?

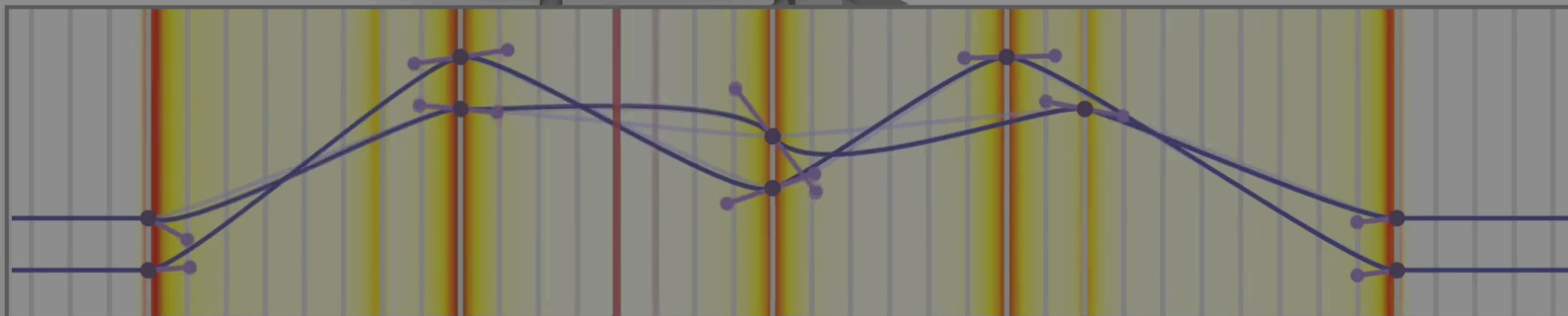
We propose
“Optimization-Guided Motion Editing”



Regularization weight:

Cancel OK

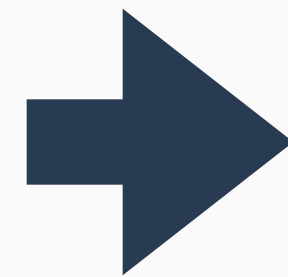
**Control for
animators**



Background: How Do Animators Create Motions?

How Do Animators Create Motions?

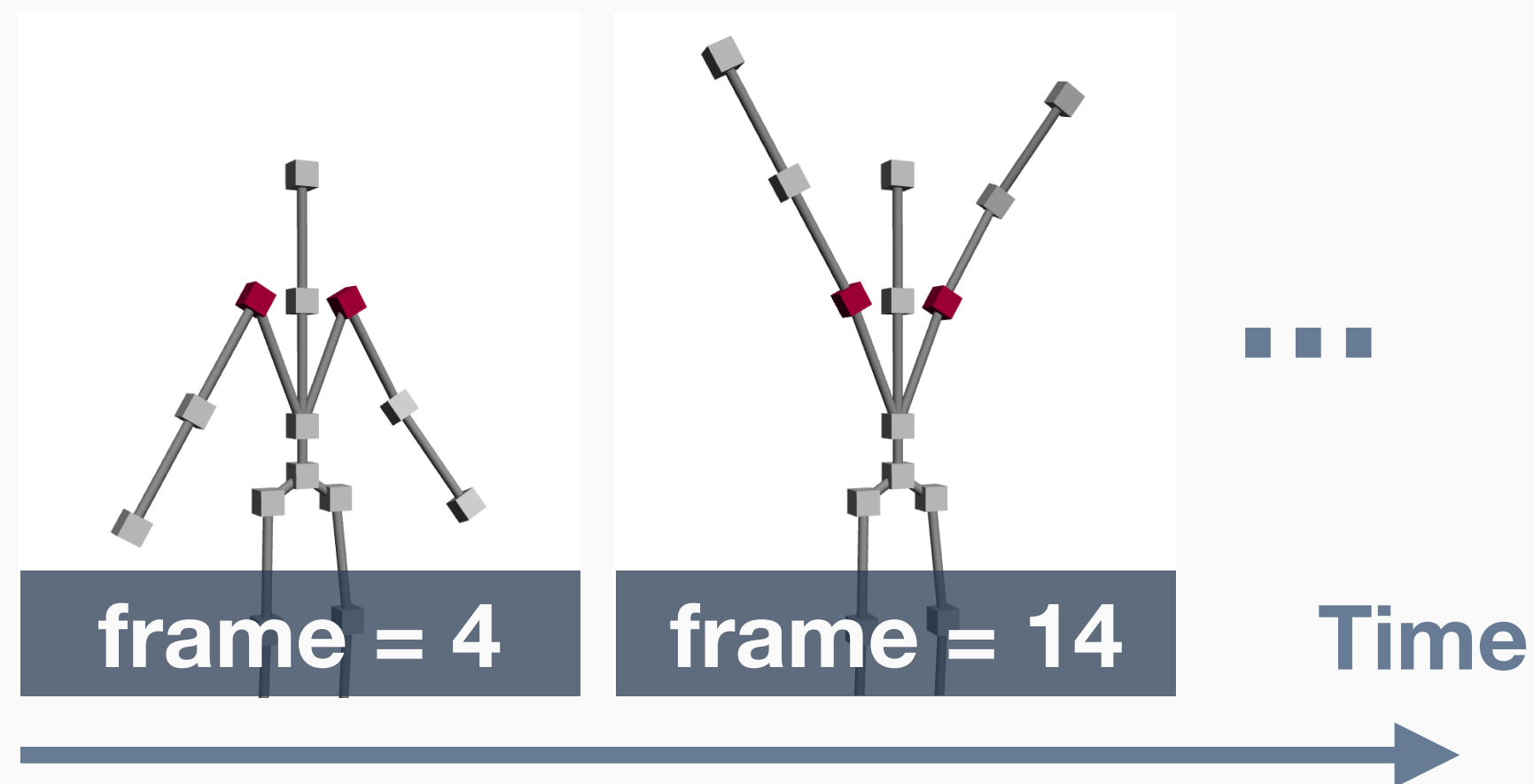
1. Keyframing



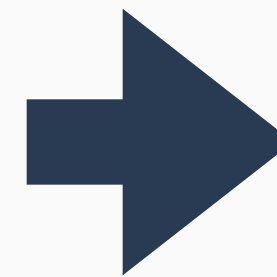
2. Curve Editing

How Do Animators Create Motions?

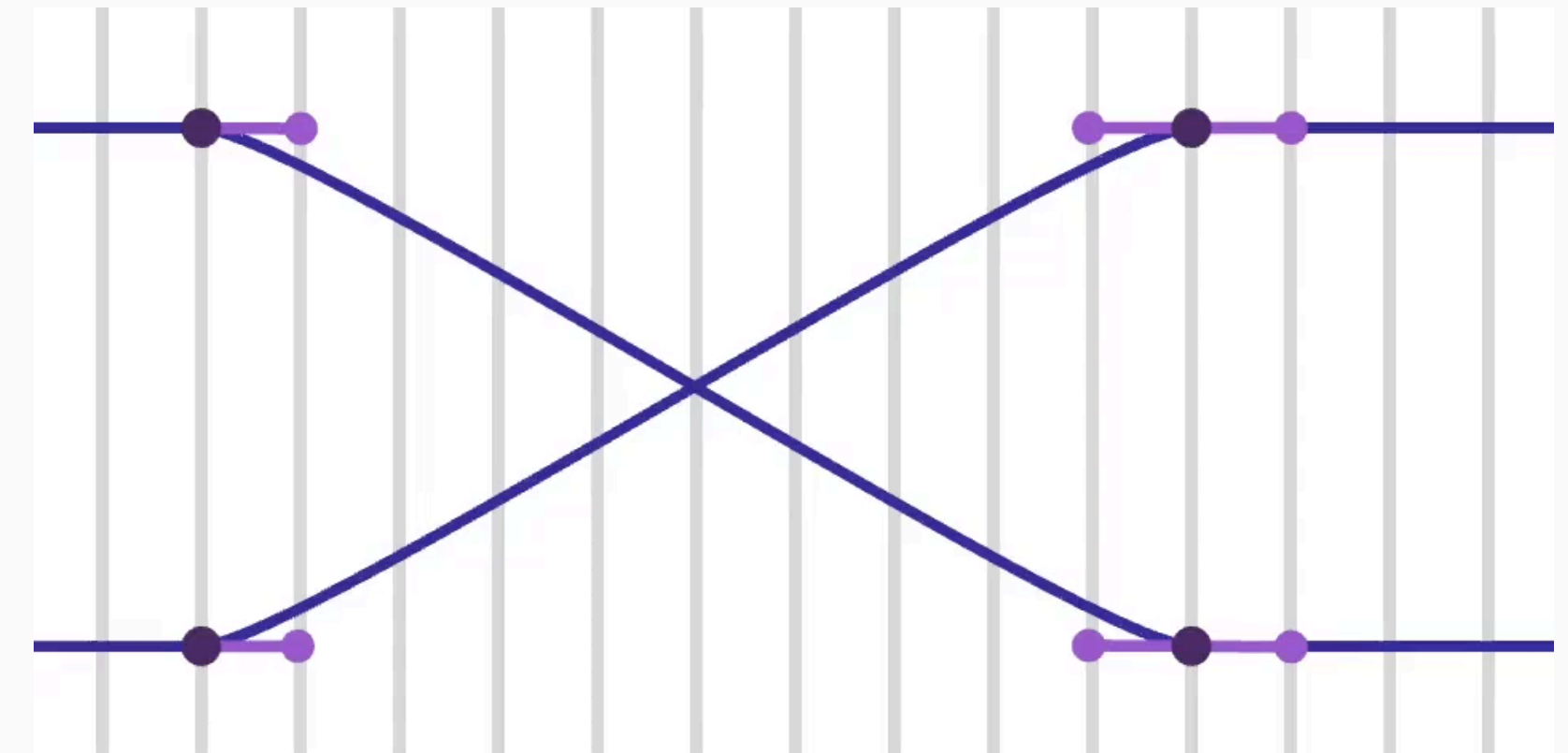
1. Keyframing



Define **key poses** at
several **keyframes**



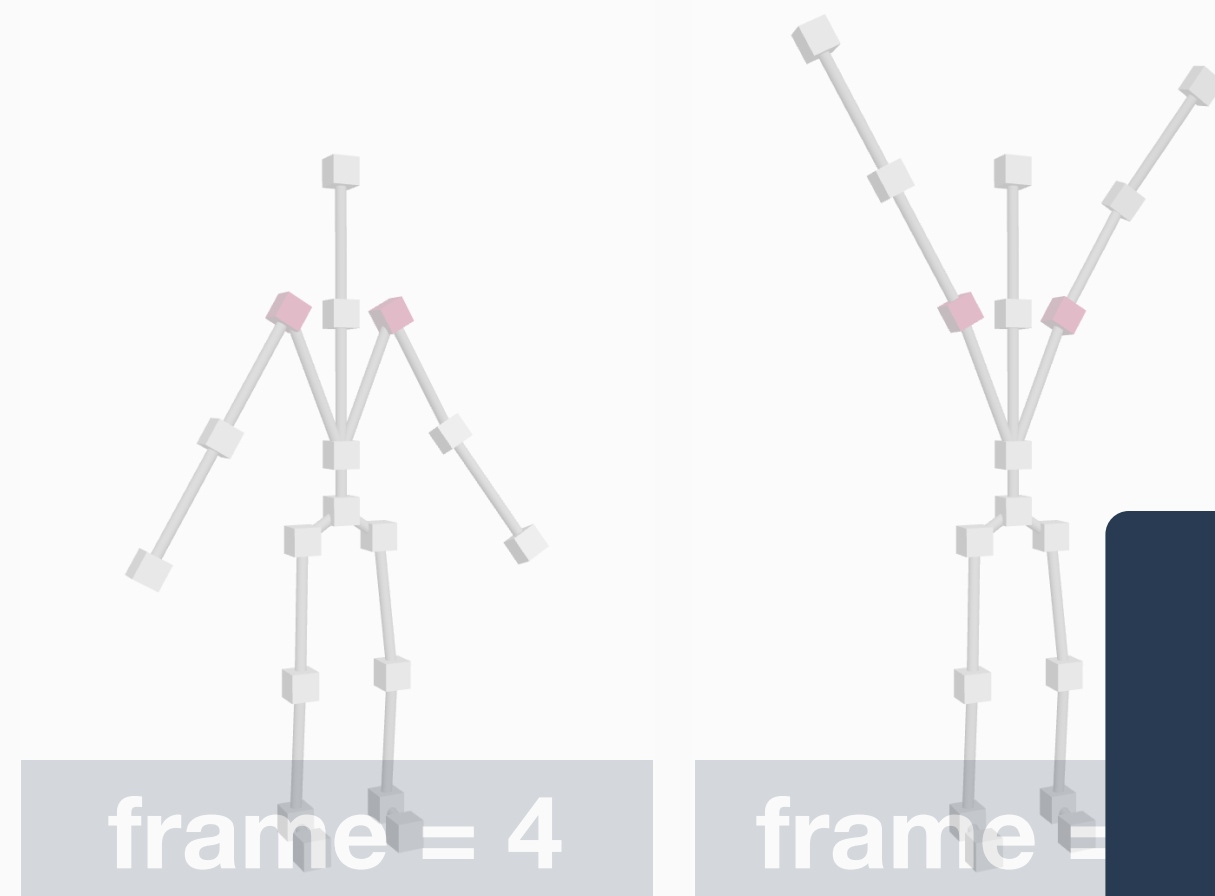
2. Curve Editing



Adjust **interpolation curves**
(how key poses are interpolated
between keyframes)

How Do Animators Create Motions?

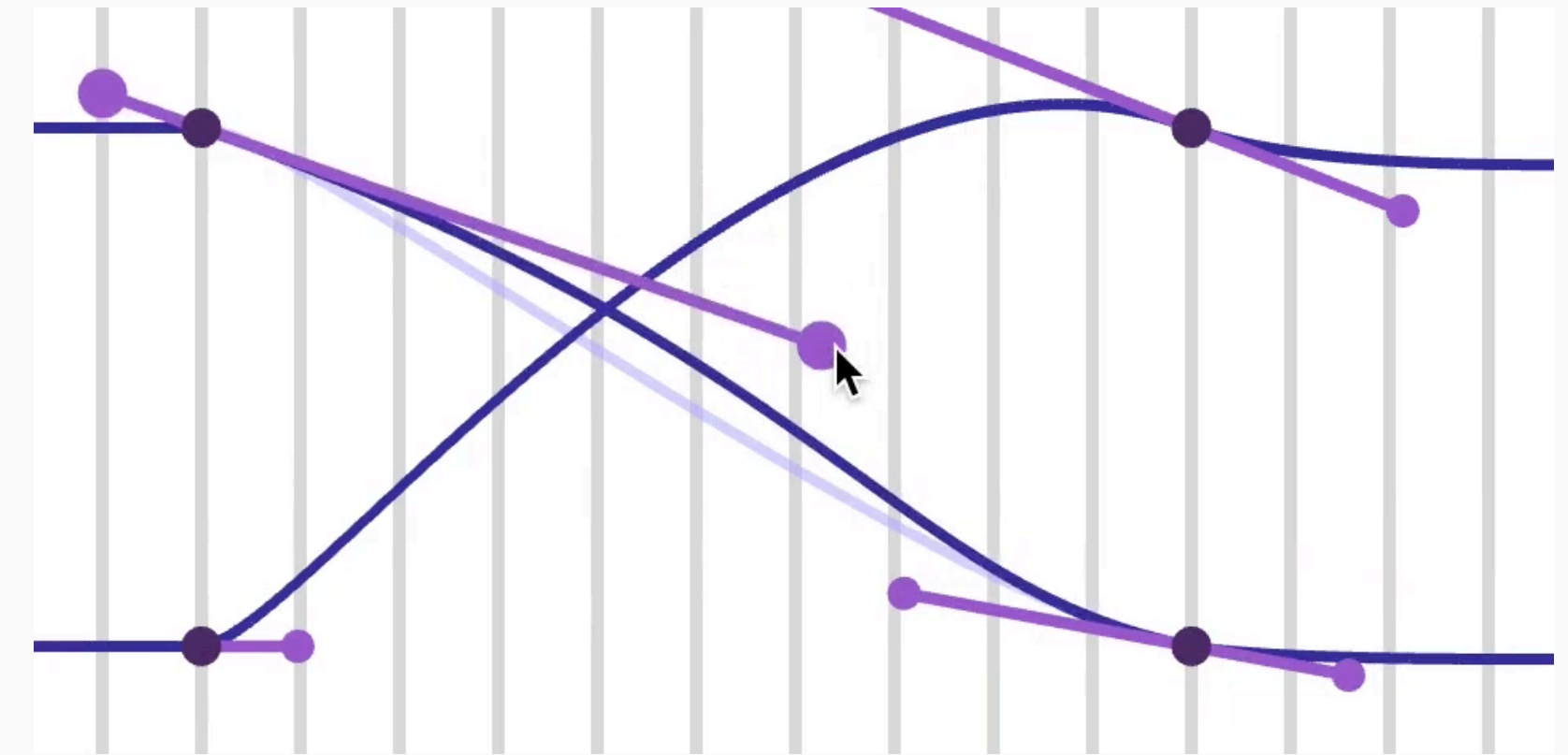
1. Keyframing



Our Target

Define **key poses** at
several **keyframes**

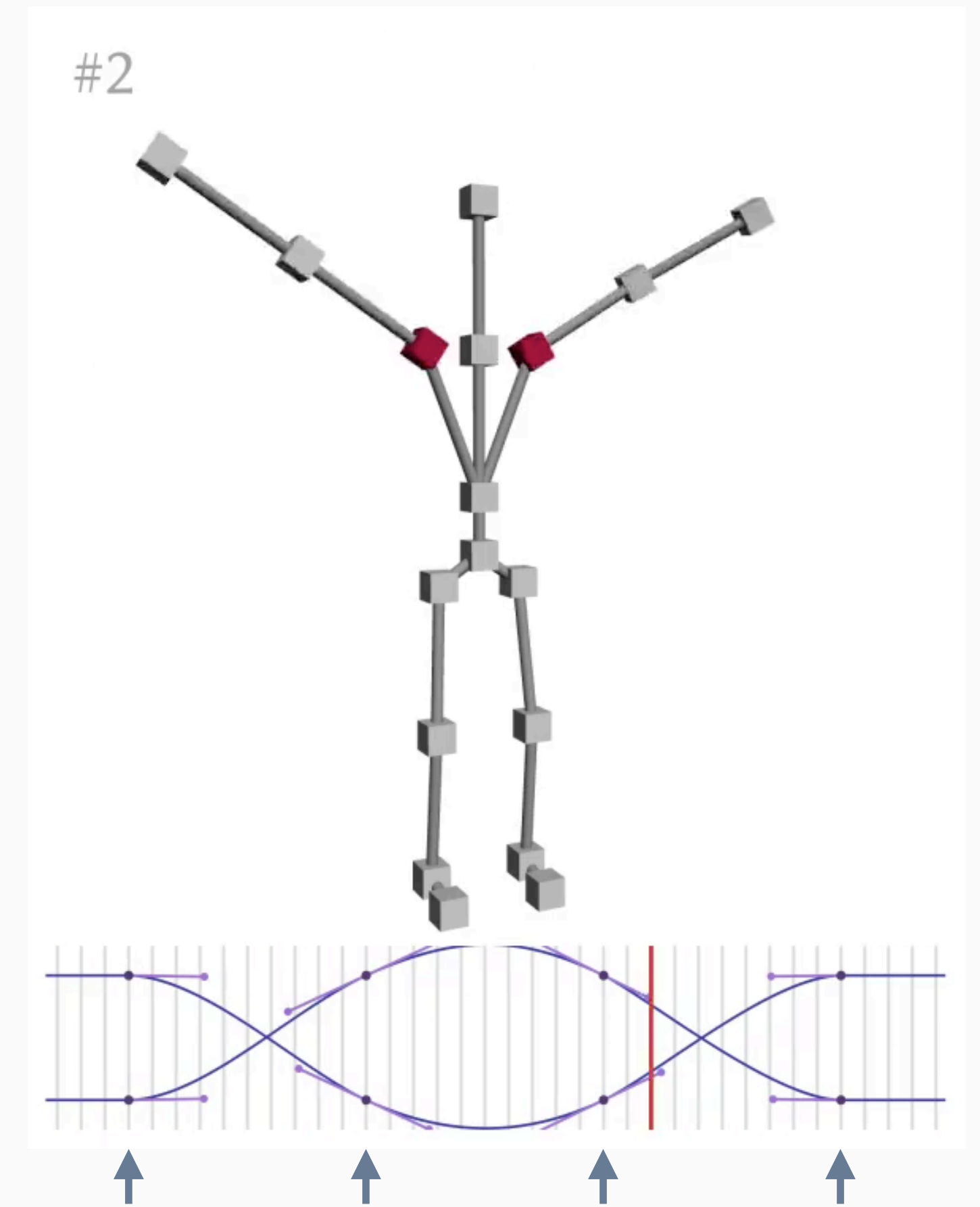
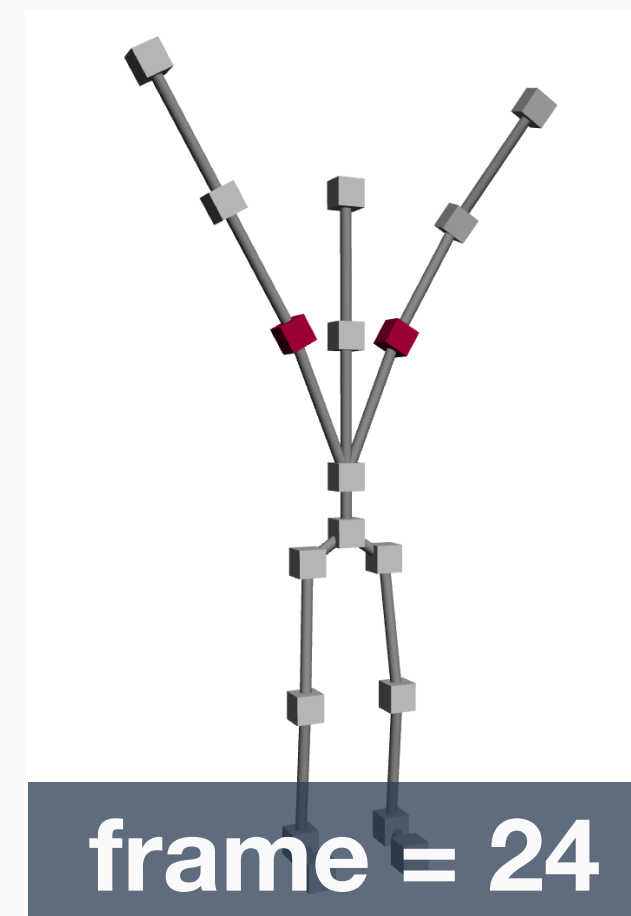
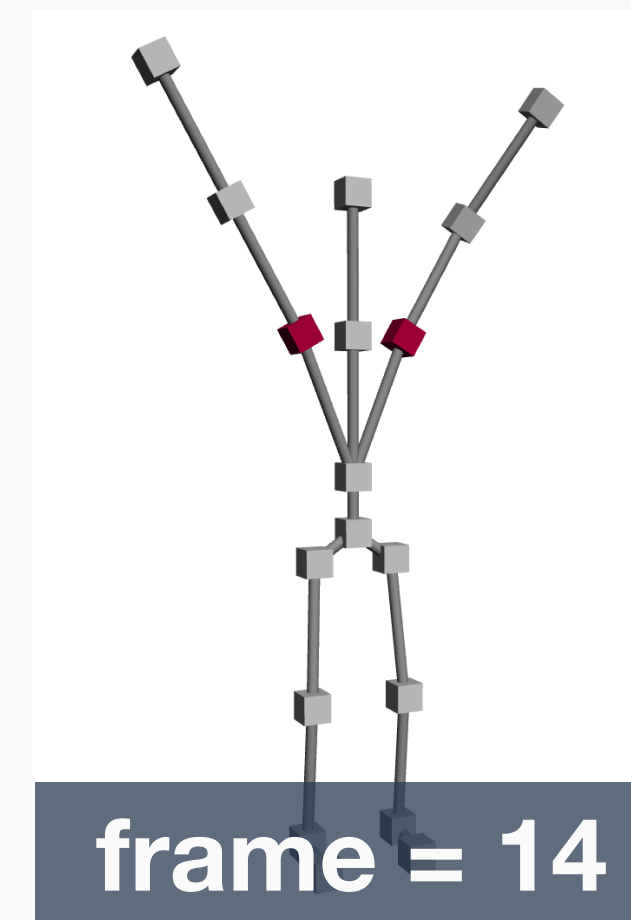
2. Curve Editing



Adjust **interpolation curves**
(how key poses are interpolated
between keyframes)

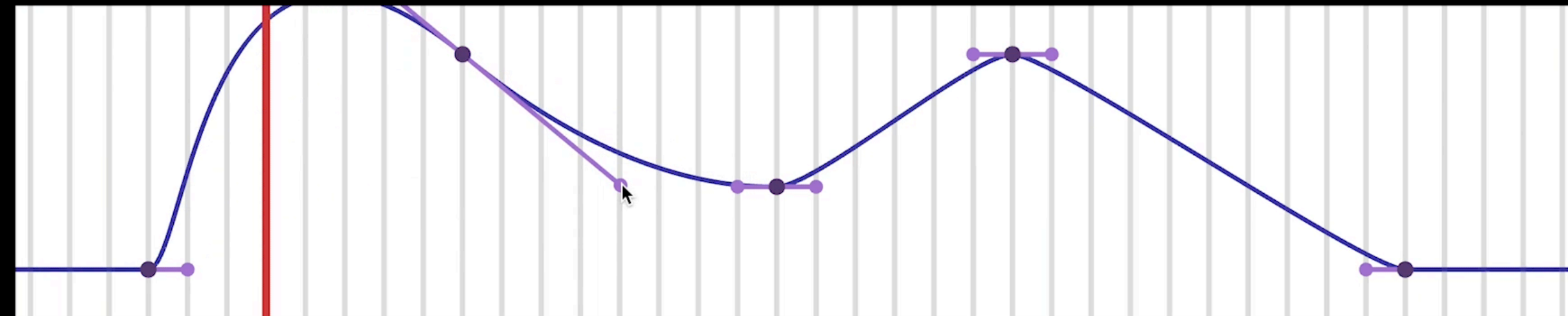
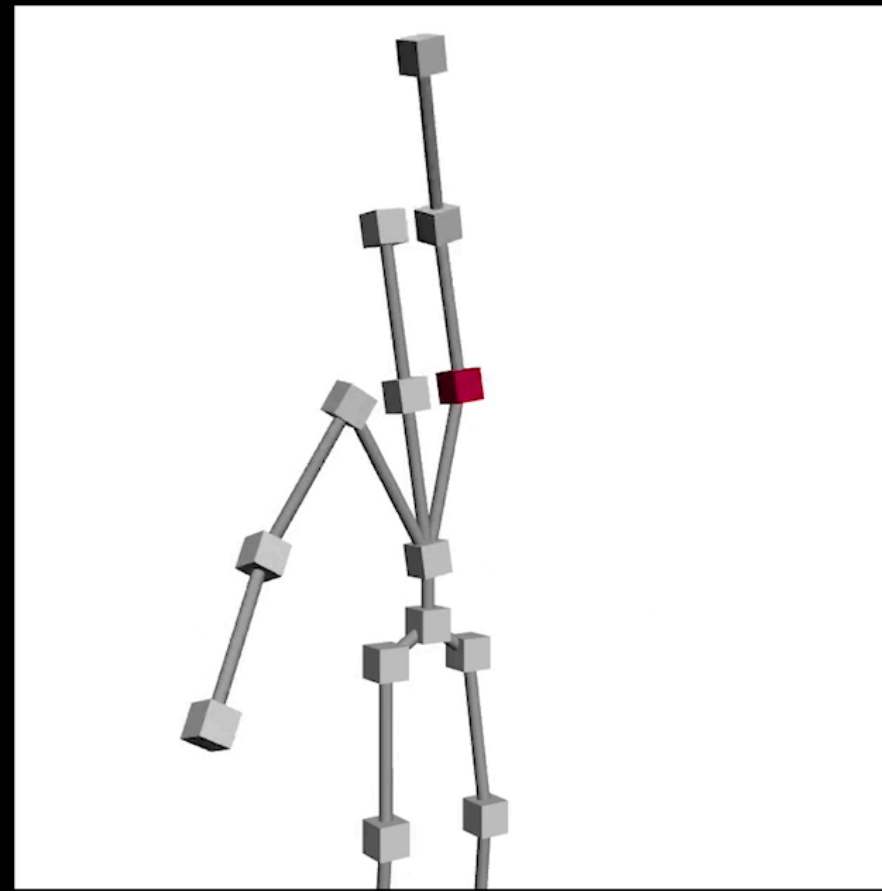
Curve Editing is Important for Nuances

Even with **the same keyframing**, motions could be very **different in nuances**



Difficulties in Curve Editing

Difficulties in Curve Editing



Unintuitive (indirect) effects

Need to play motions every time manipulating handles

Many parameters

High-dimensional search task

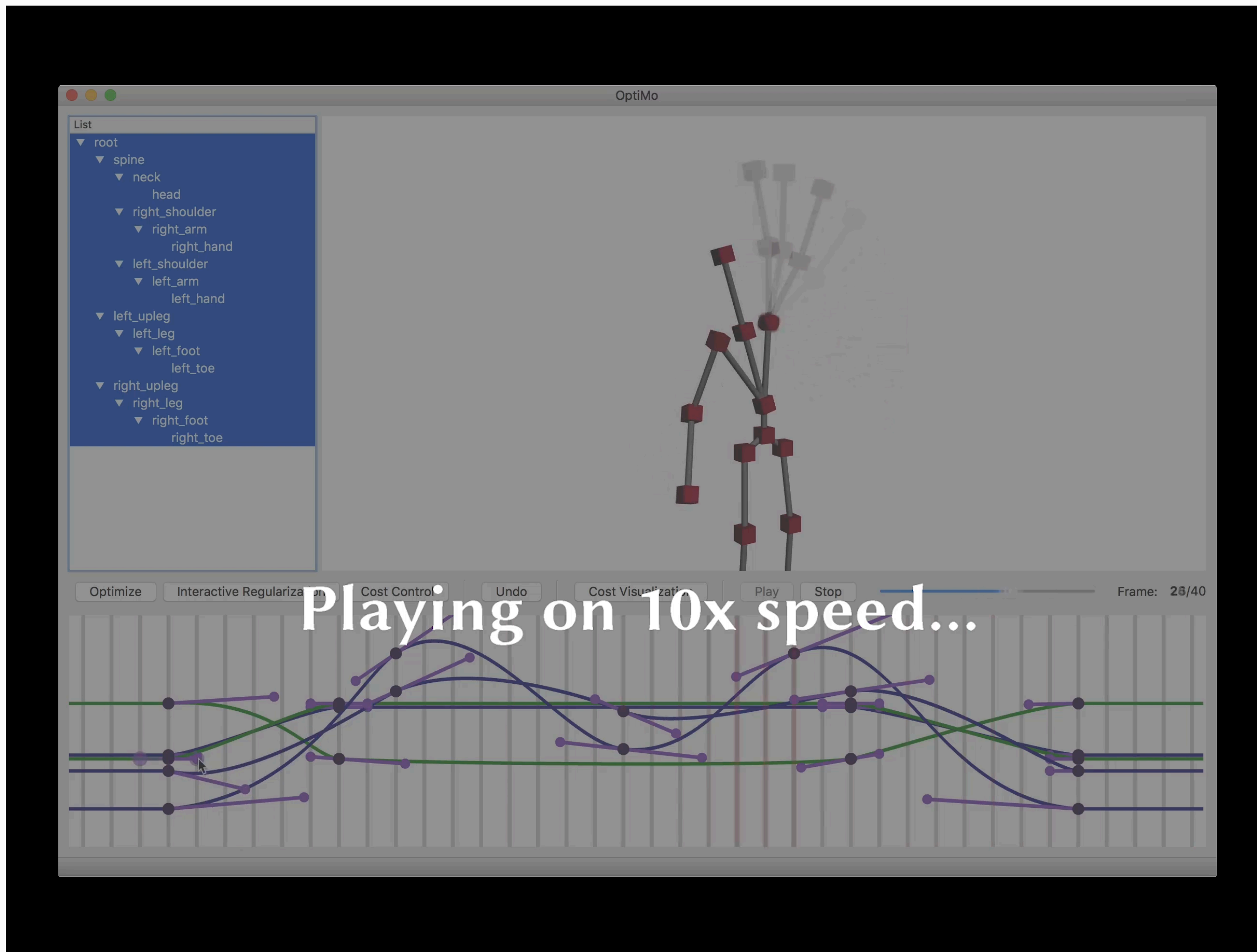
Difficulties in Curve Editing

Unintuitive (indirect) effects

Need to play motions every time manipulating handles

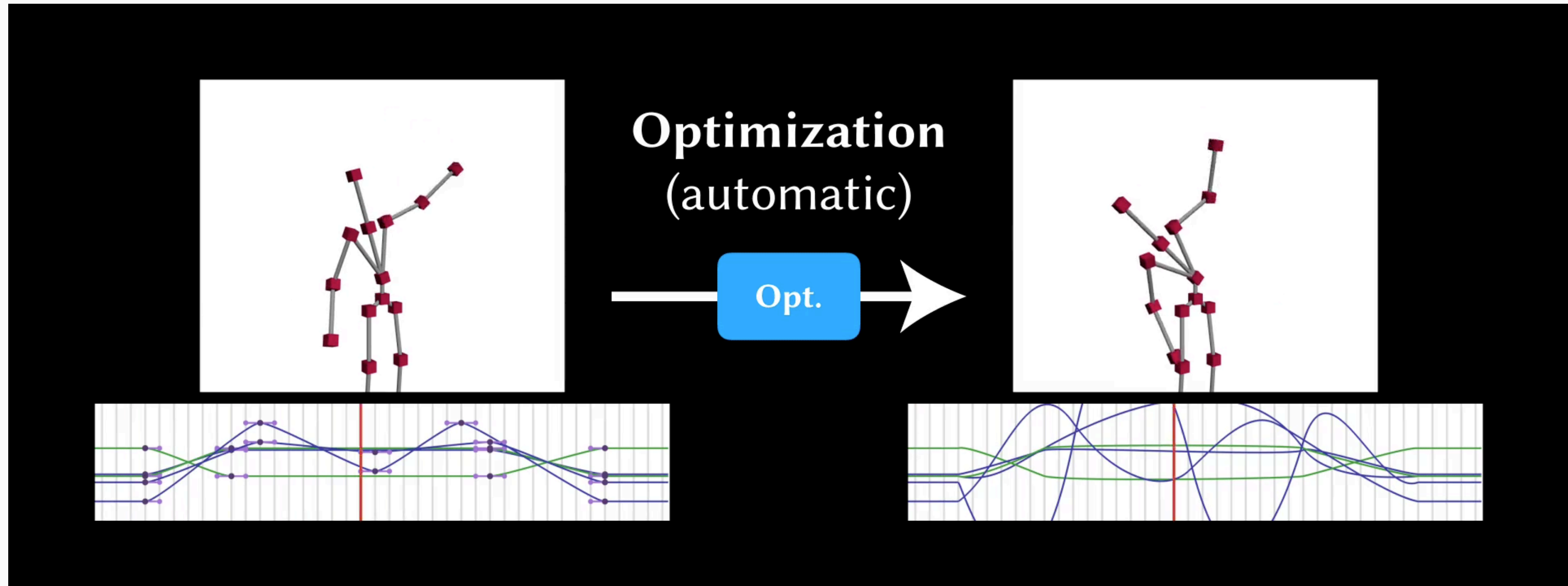
Many parameters

High-dimensional search task



A Possible (?) Solution: Naïve Optimization

Optimization for Full Automation?



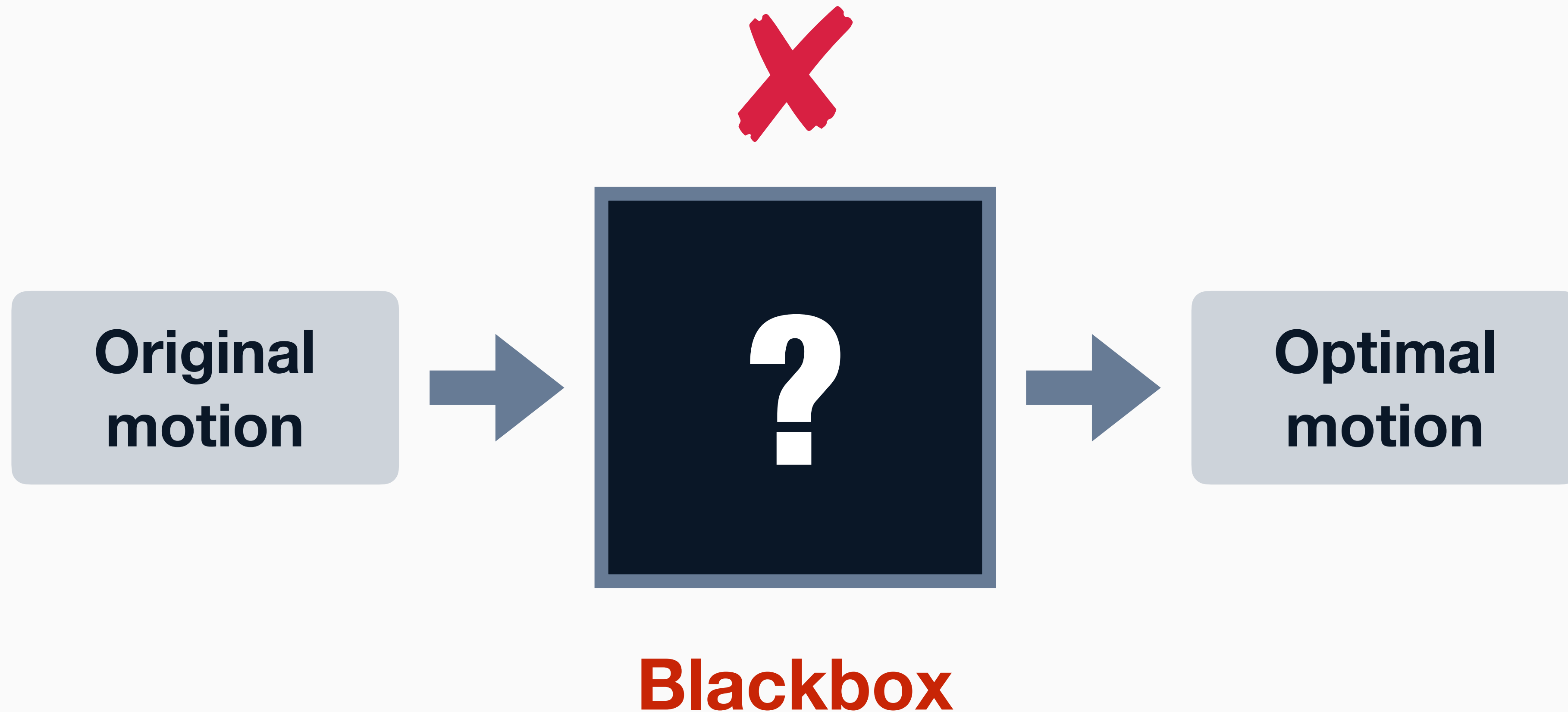
Optimization can fully automate the task,
but results are **not always satisfactory (no control...)**

Optimization-Guided Motion Editing

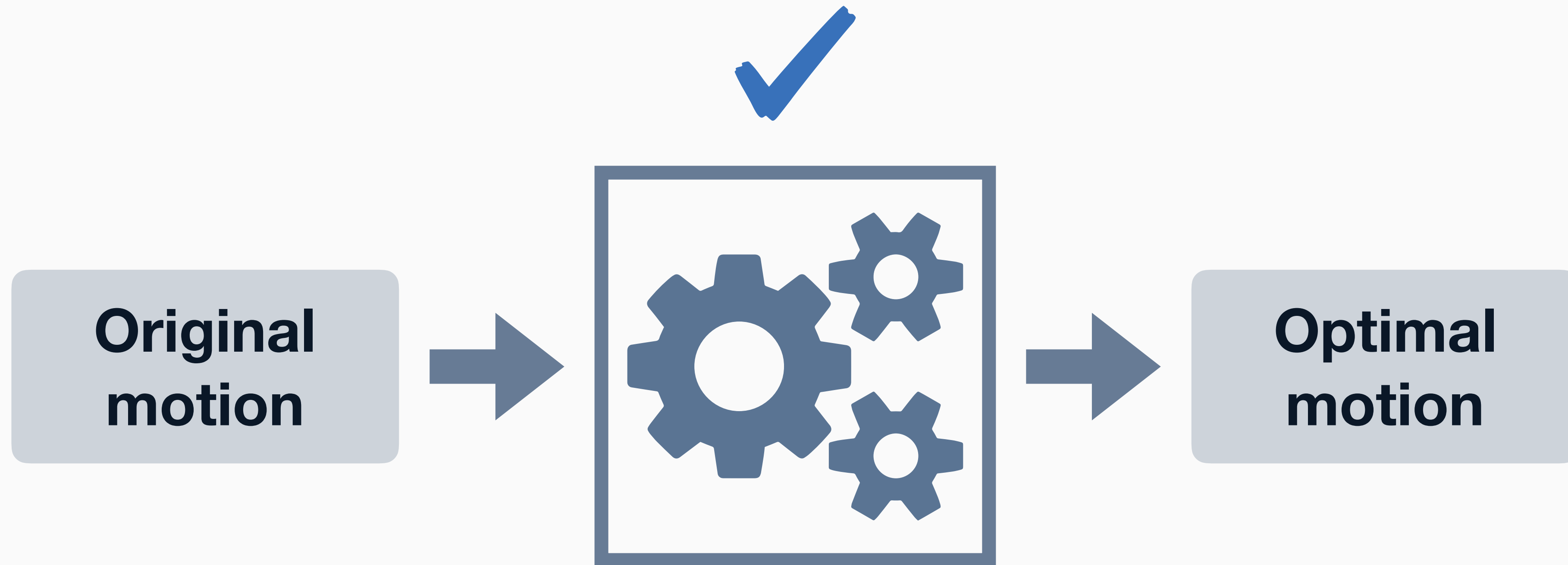
Live Demo!

Design Goals: What Should be Considered for Effective Interaction?

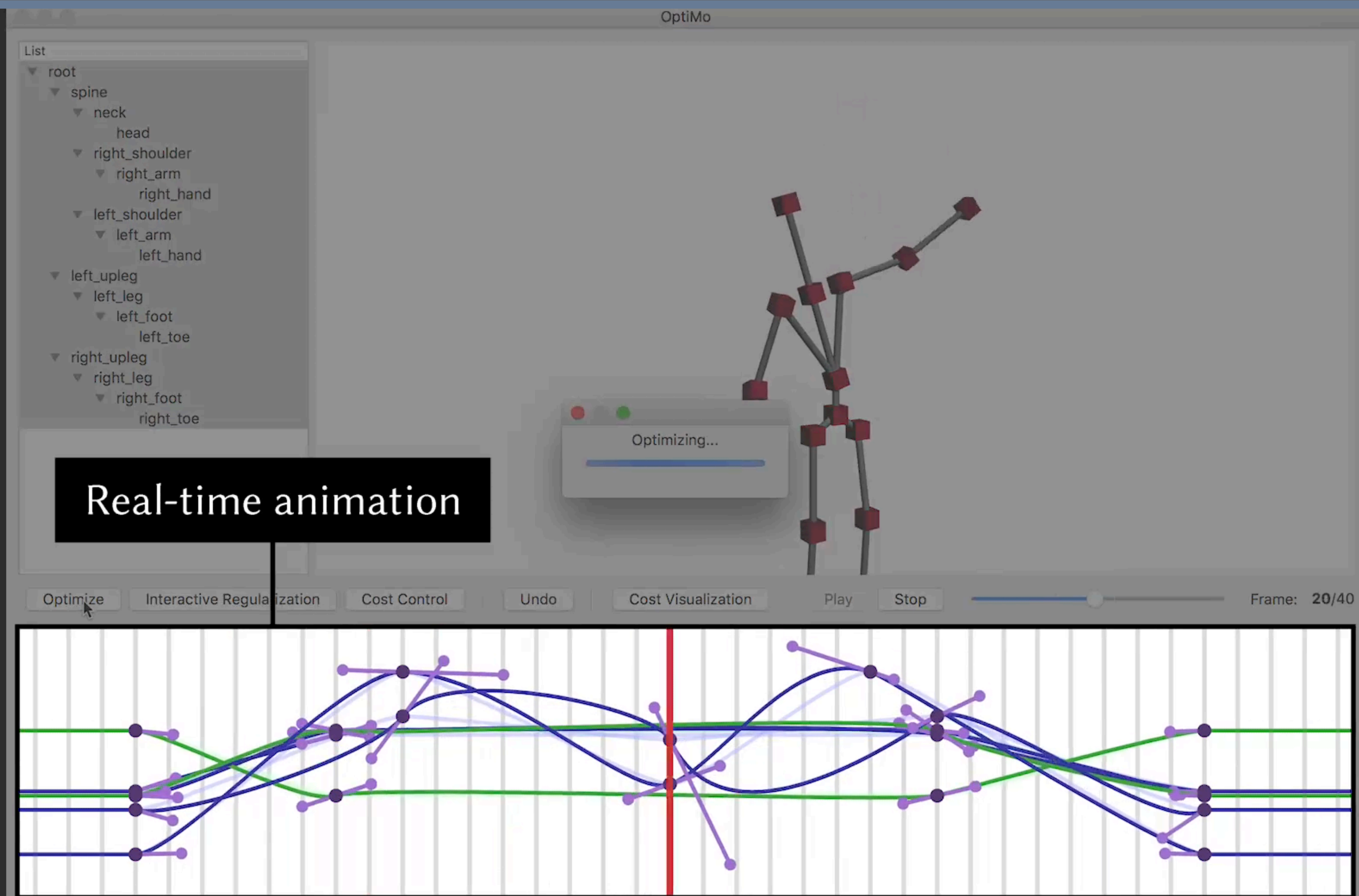
Three Design Goals



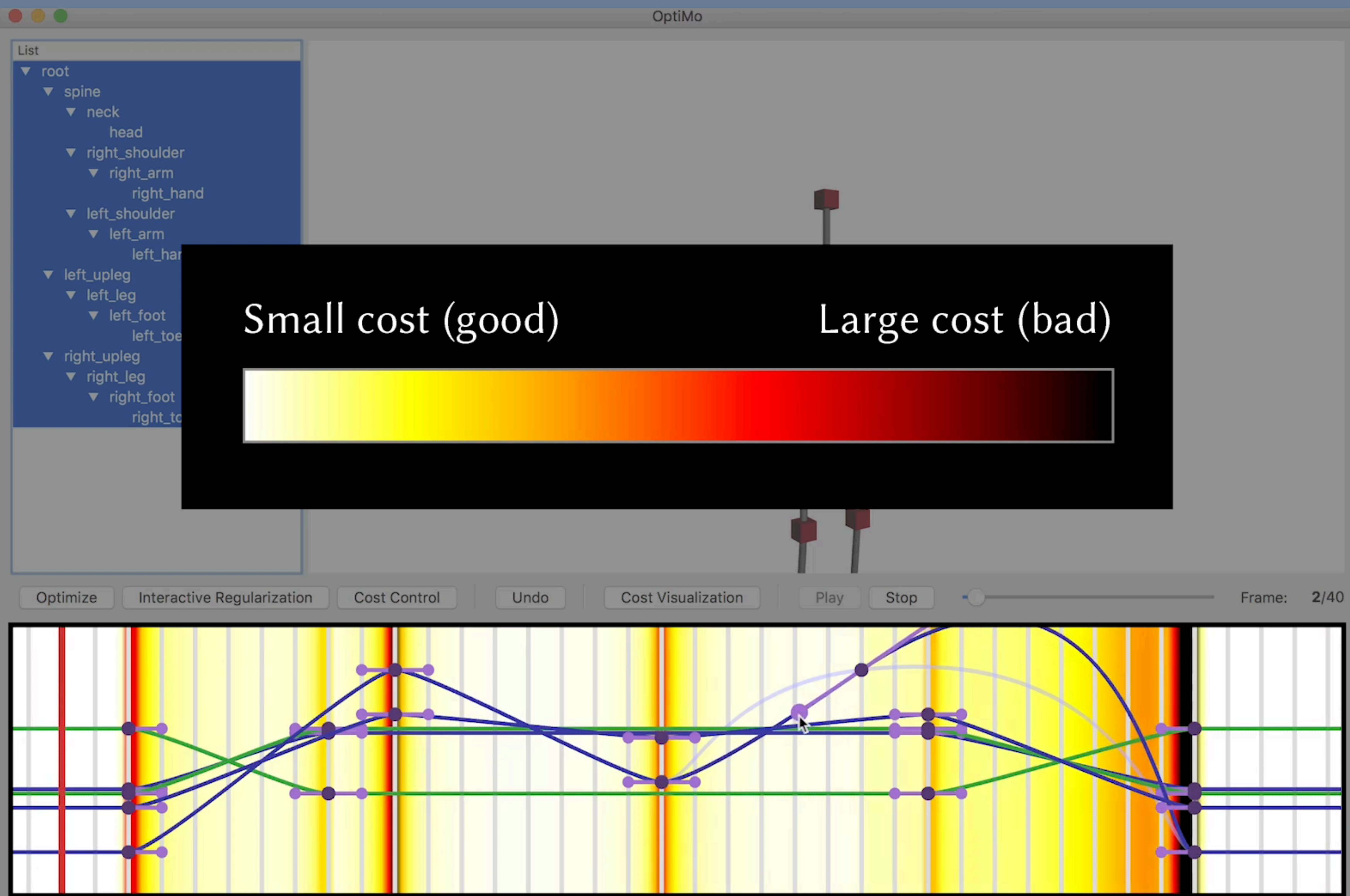
Three Design Goals



1. Transparent

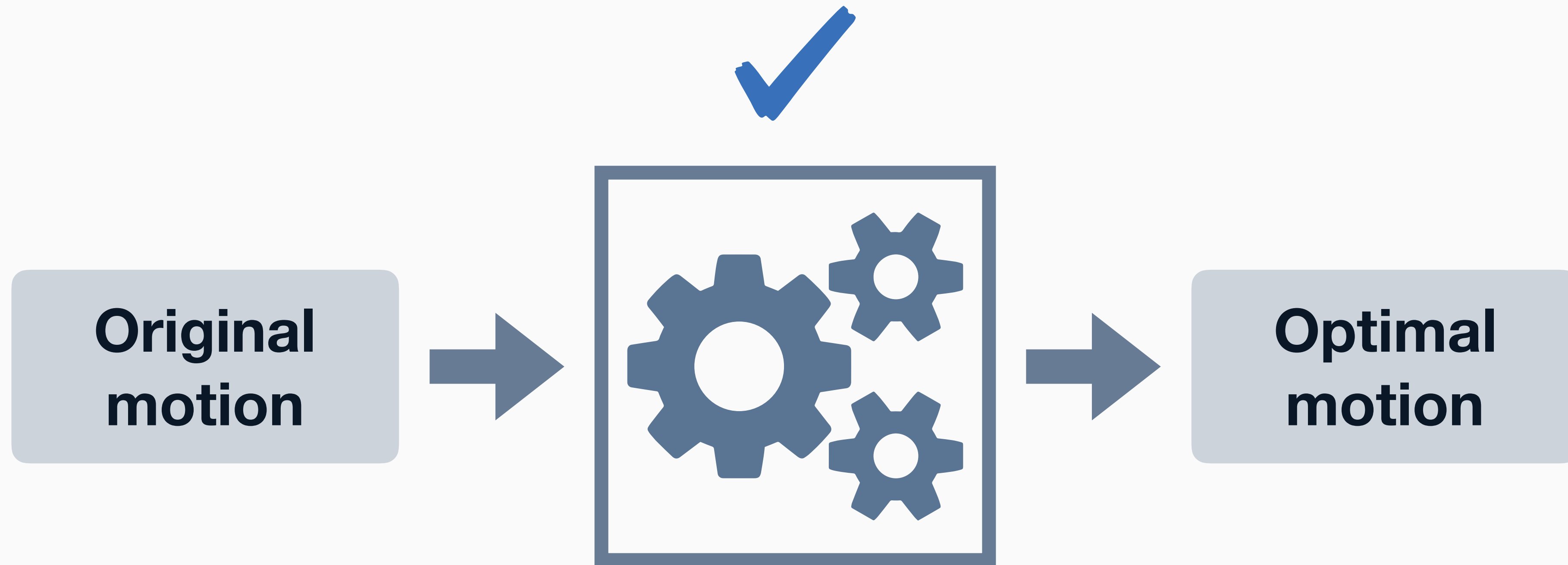


Feature: **Optimization Process Visualization** (for Transparency)



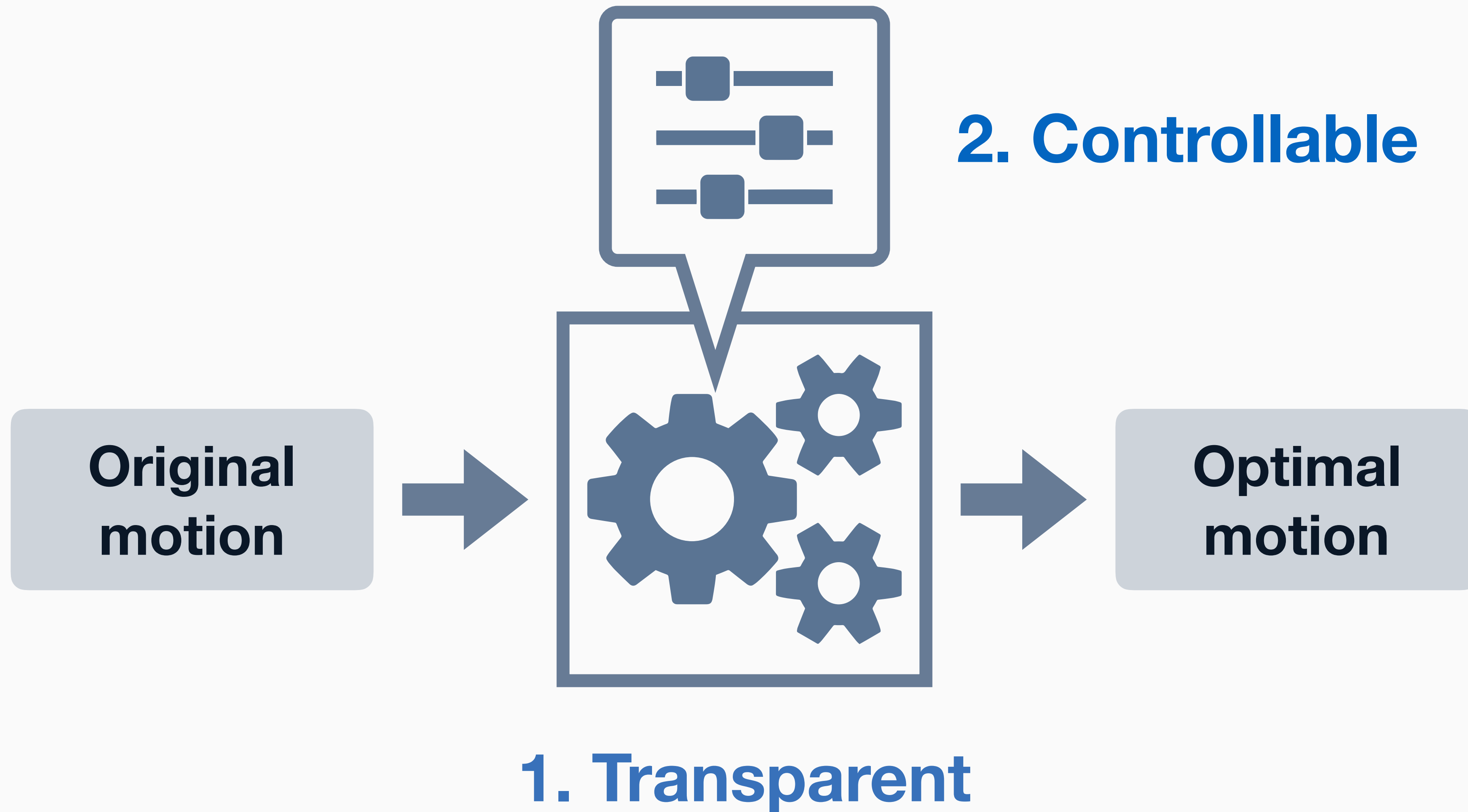
Feature: **Cost Visualization** (for Transparency)

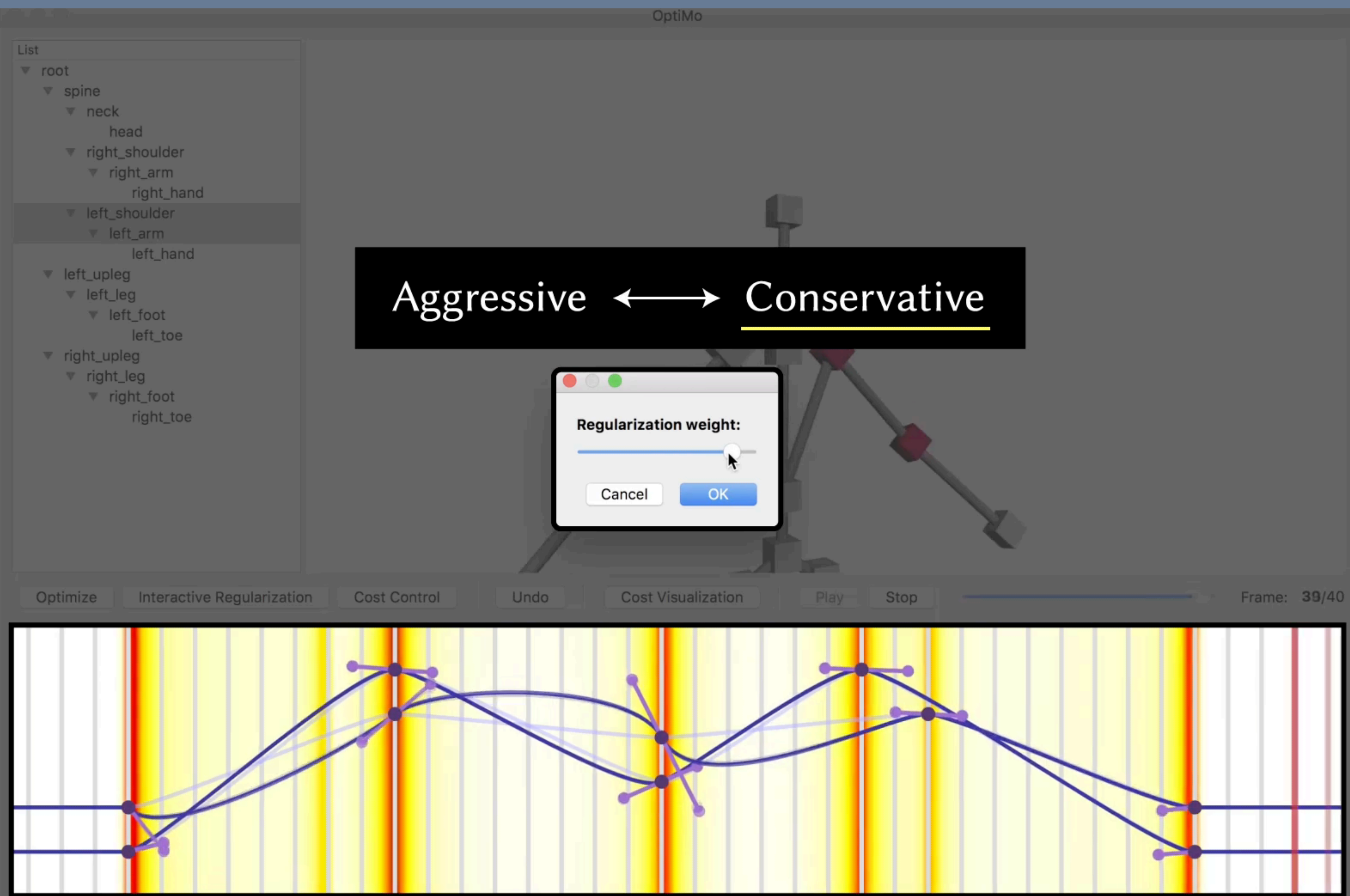
Three Design Goals



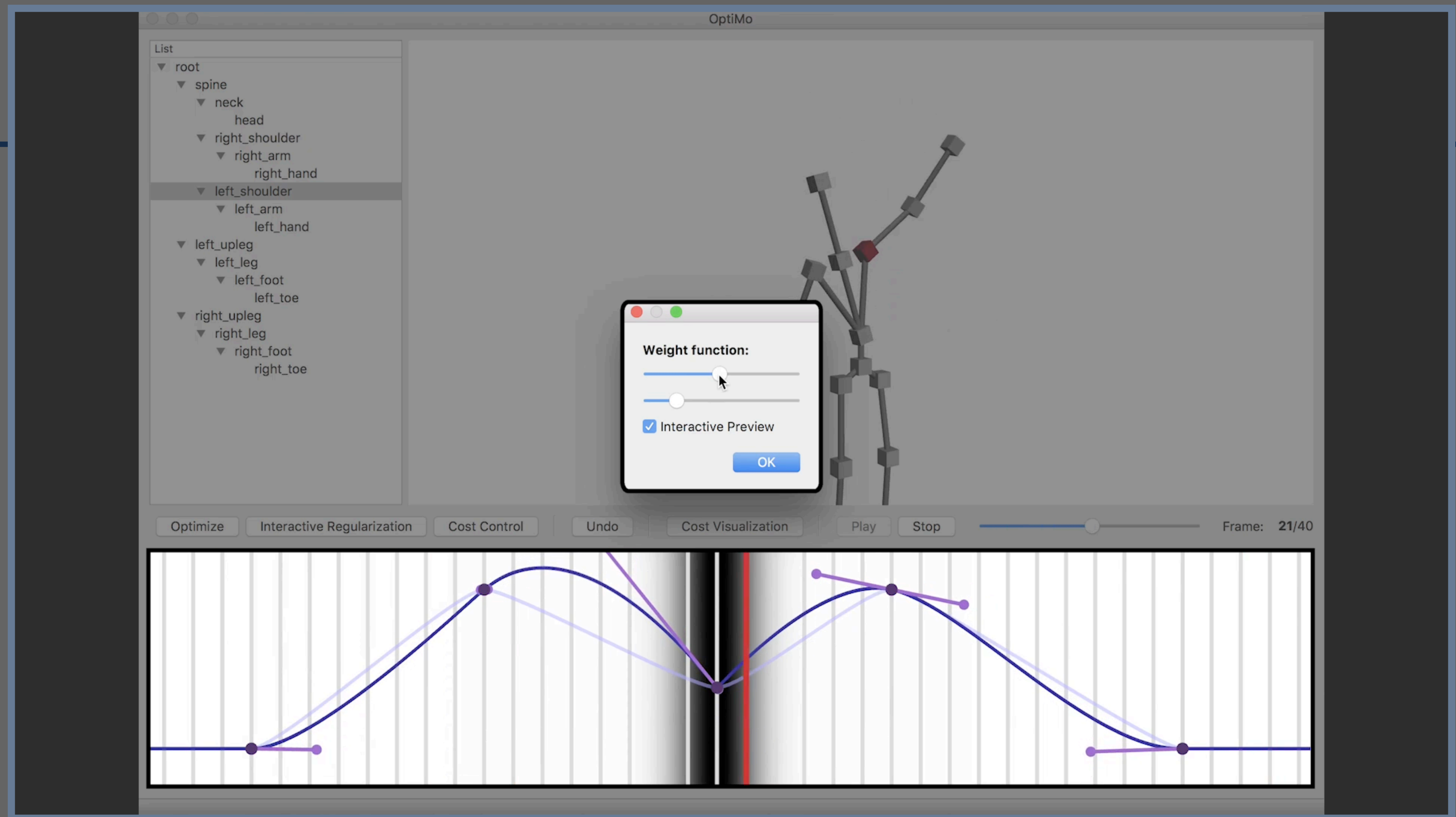
1. Transparent

Three Design Goals



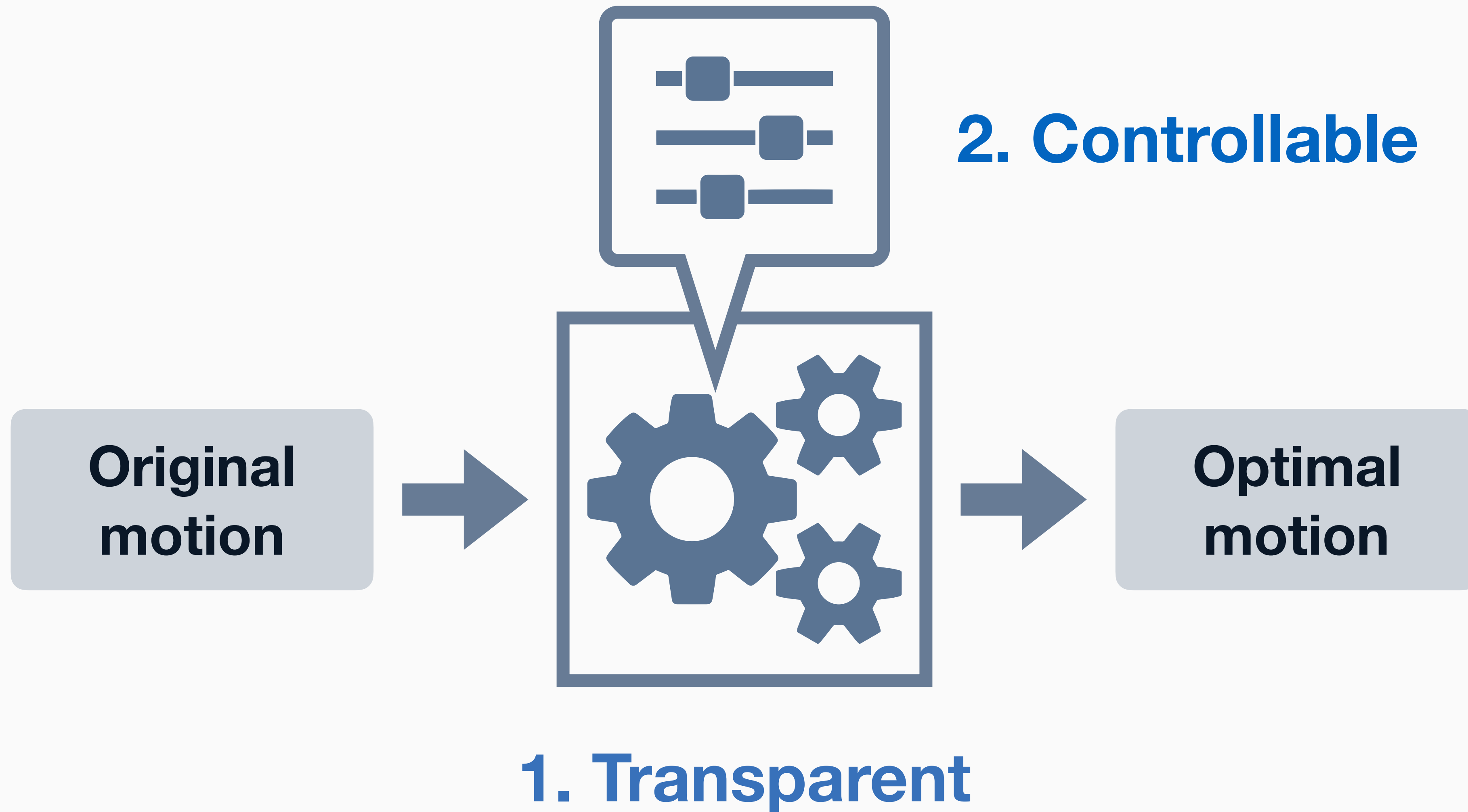


1. Transparent Feature: **Interactive Regularization** (for Controllability)

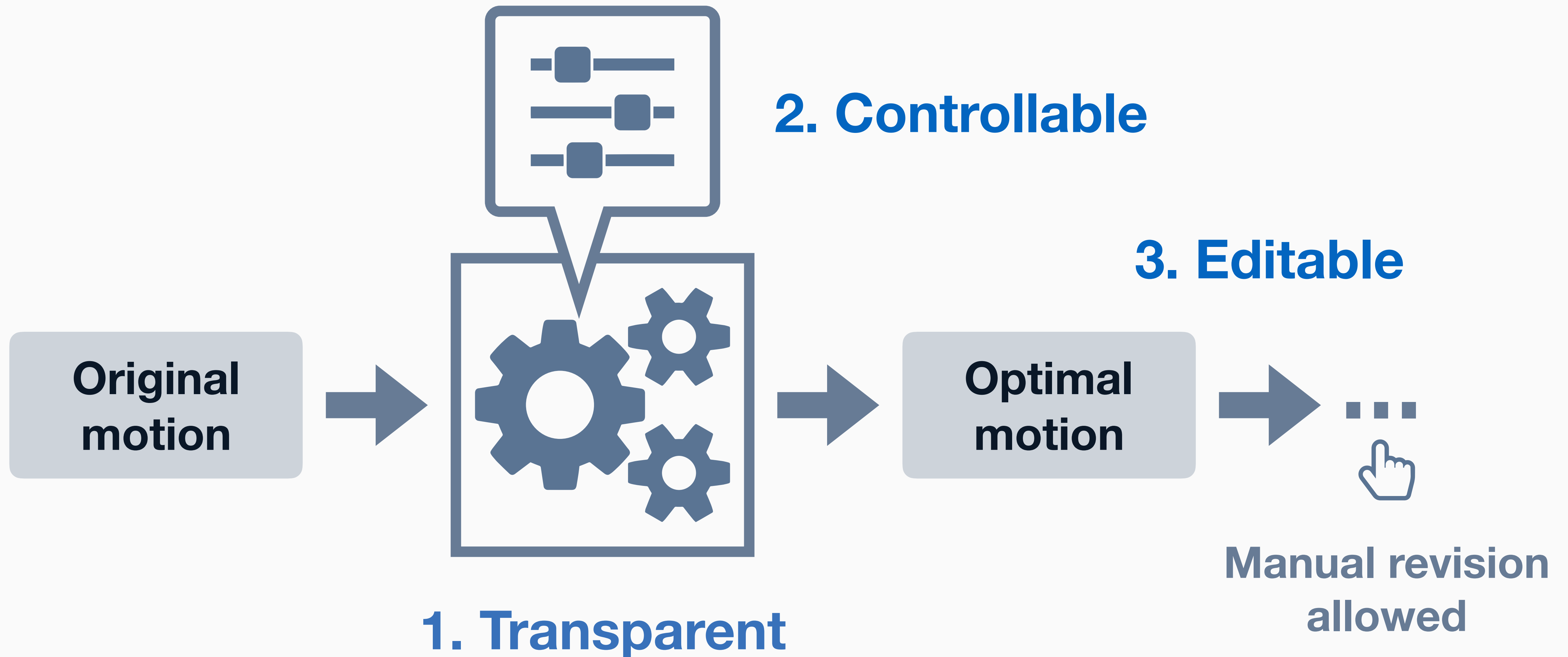


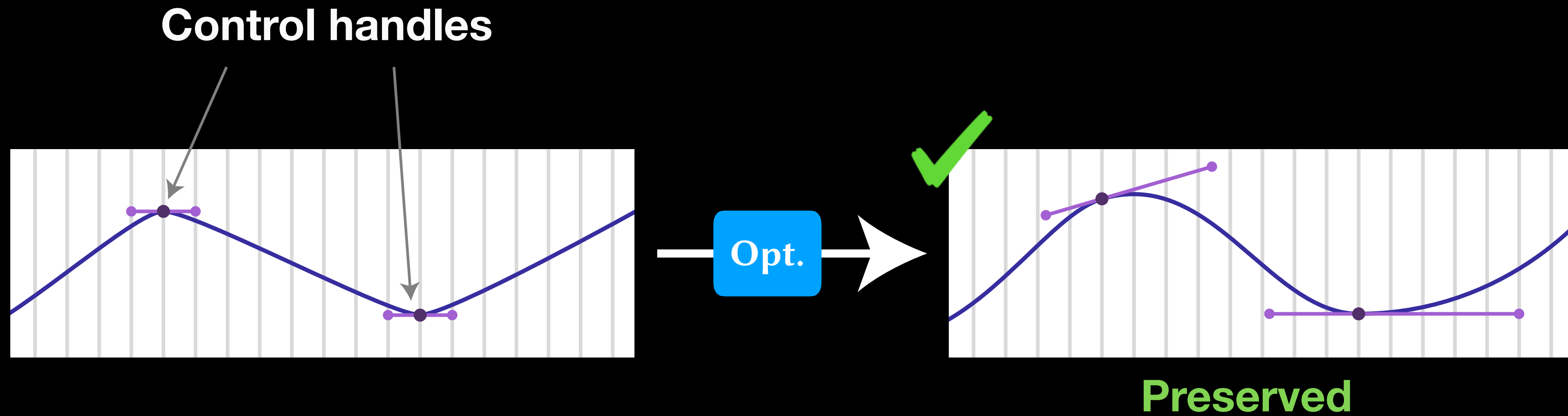
1. Transparent Feature: **Time-Varying Cost Control** (for Controllability)

Three Design Goals



Three Design Goals



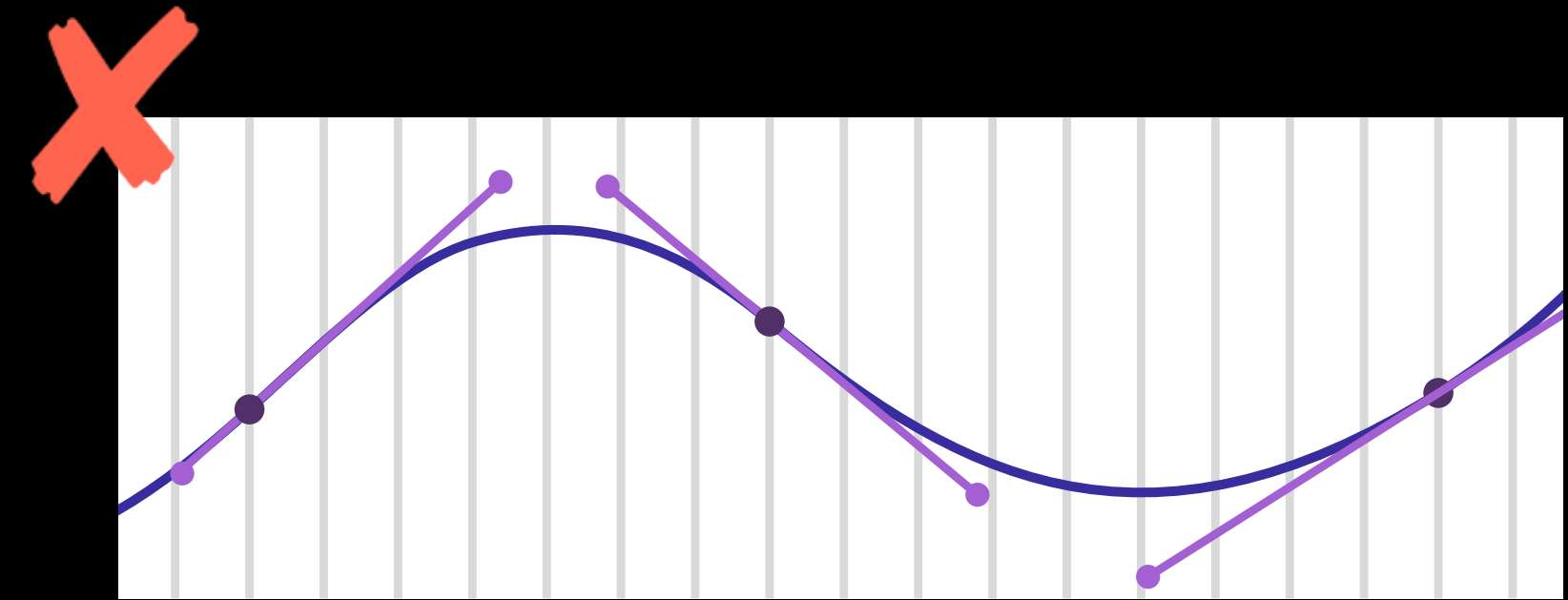


1. Transparent

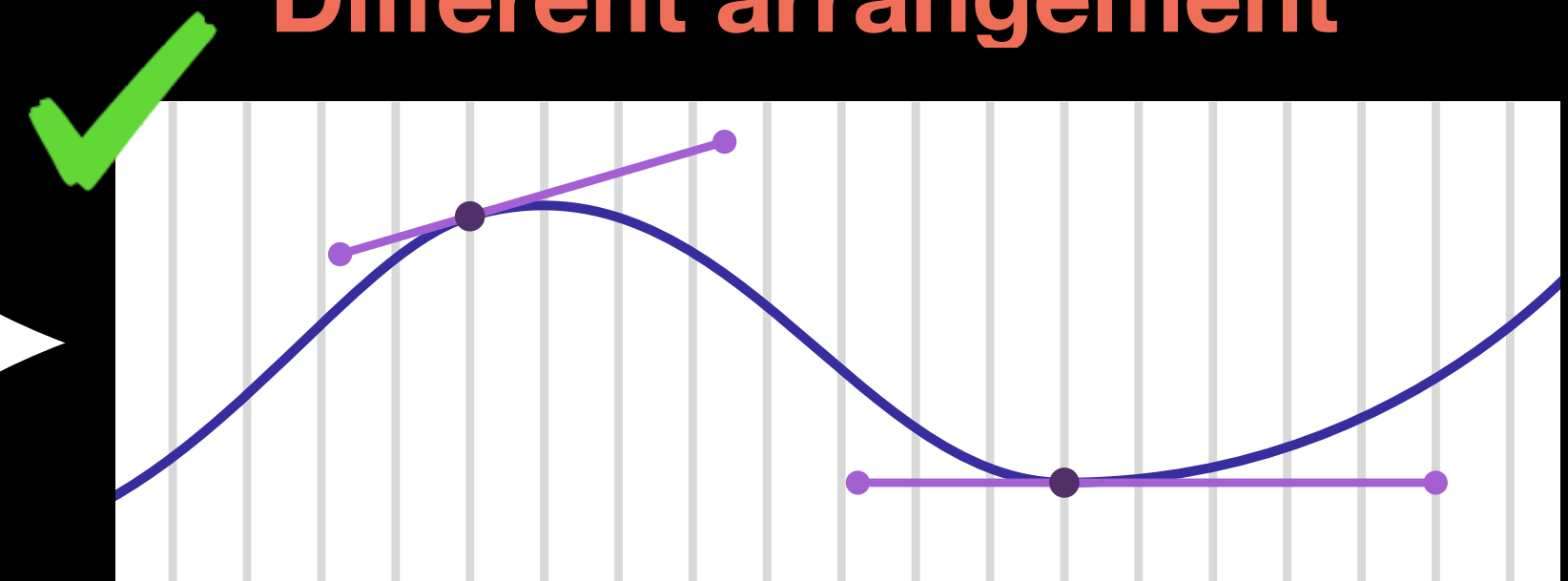
Feature: **Preservation of Control Handles** (for Editability)



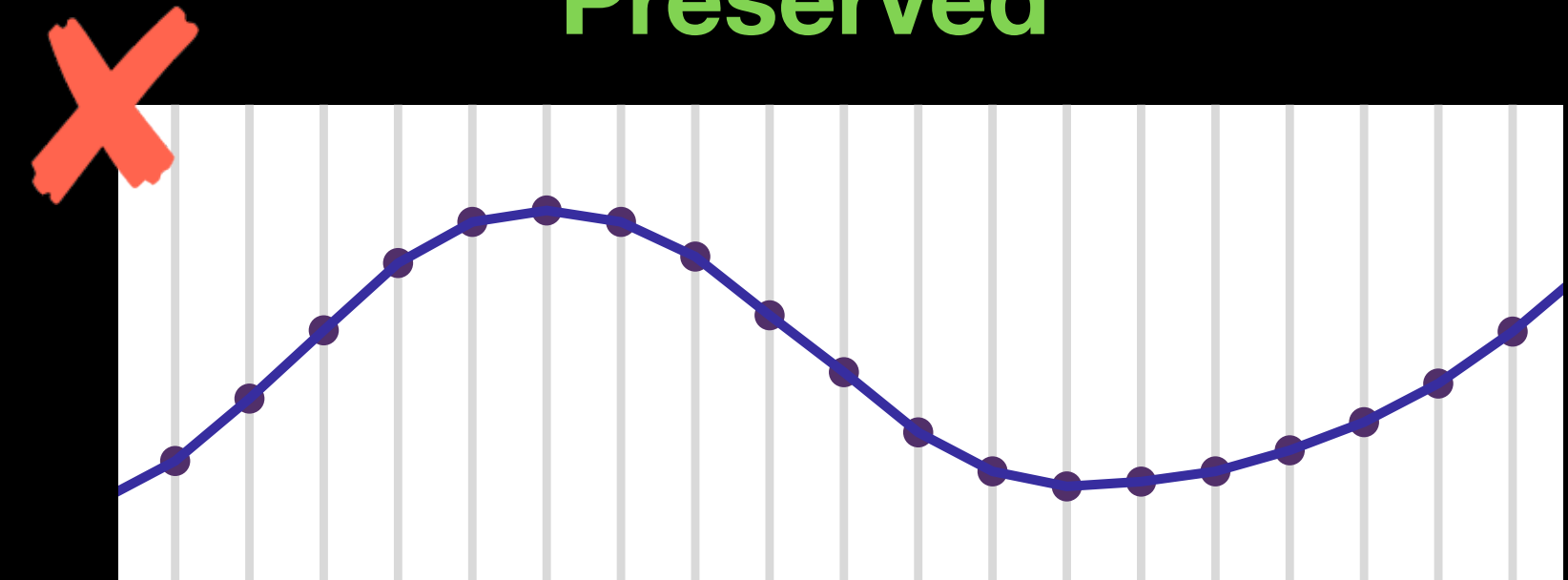
Opt.



Different arrangement



Preserved

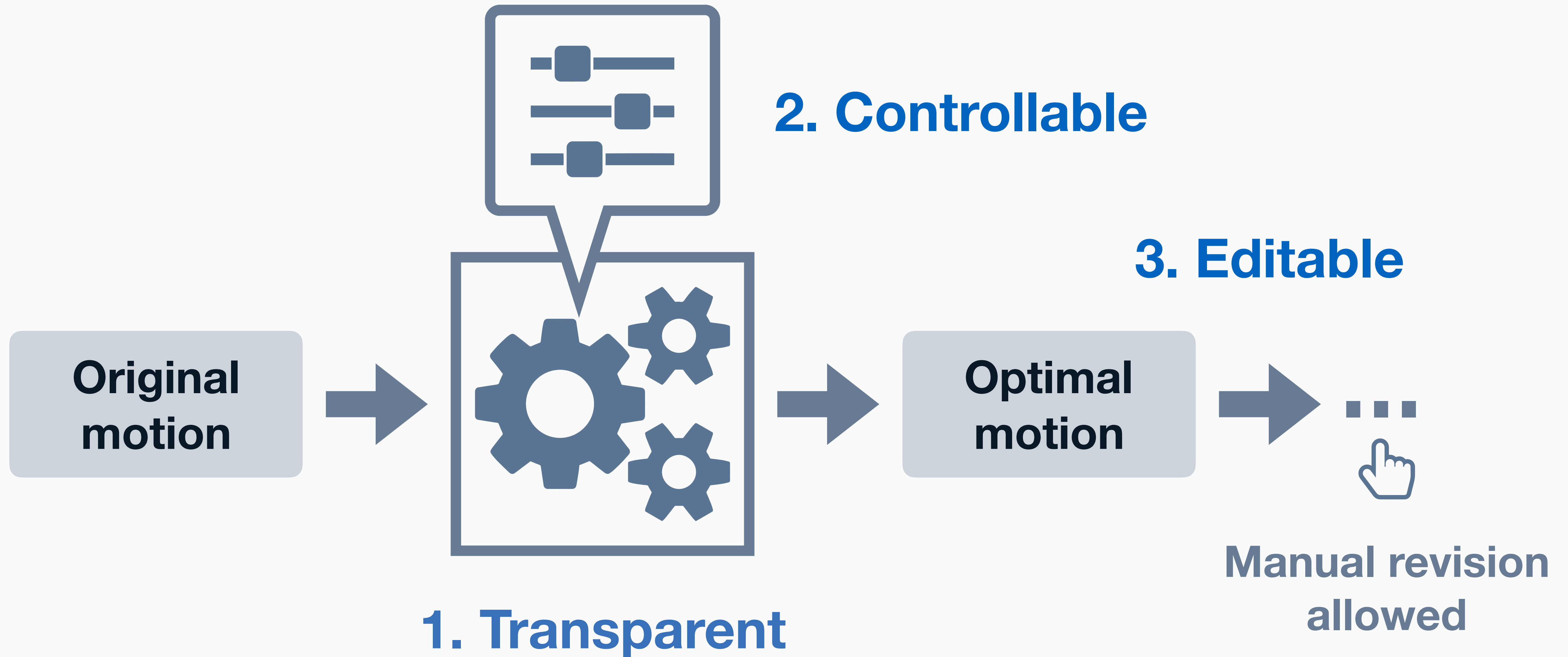


Different data structure

1. Transparent

Feature: **Preservation of Control Handles** (for Editability)

Three Design Goals



Math: How to Formulate Optimization with Controllability

Optimization with Controllability

Typical
optimization

$$\min_{\mathbf{x}} \left\{ \int C(\mathbf{x}, t) dt \right\}$$

Optimization with Controllability

Typical optimization

$$\min_{\mathbf{x}} \left\{ \int C(\mathbf{x}, t) dt \right\}$$

Controllable optimization

$$\min_{\mathbf{x}} \left\{ \int w^{\text{cost}}(t) C(\mathbf{x}, t) dt + w^{\text{reg}} D(\mathbf{x}, \mathbf{x}^{\text{init}}) \right\}$$

Control by
time-varying cost weight

Control by
regularization weight

Optimization with Controllability

**Typical
optimization**

$$\min_{\mathbf{x}} \left\{ \int C(\mathbf{x}, t) dt \right\}$$

Adjusted in real time

**Controllable
optimization**

$$\min_{\mathbf{x}} \left\{ \int \underline{w^{\text{cost}}(t)} C(\mathbf{x}, t) dt + \underline{w^{\text{reg}}} D(\mathbf{x}, \mathbf{x}^{\text{init}}) \right\}$$

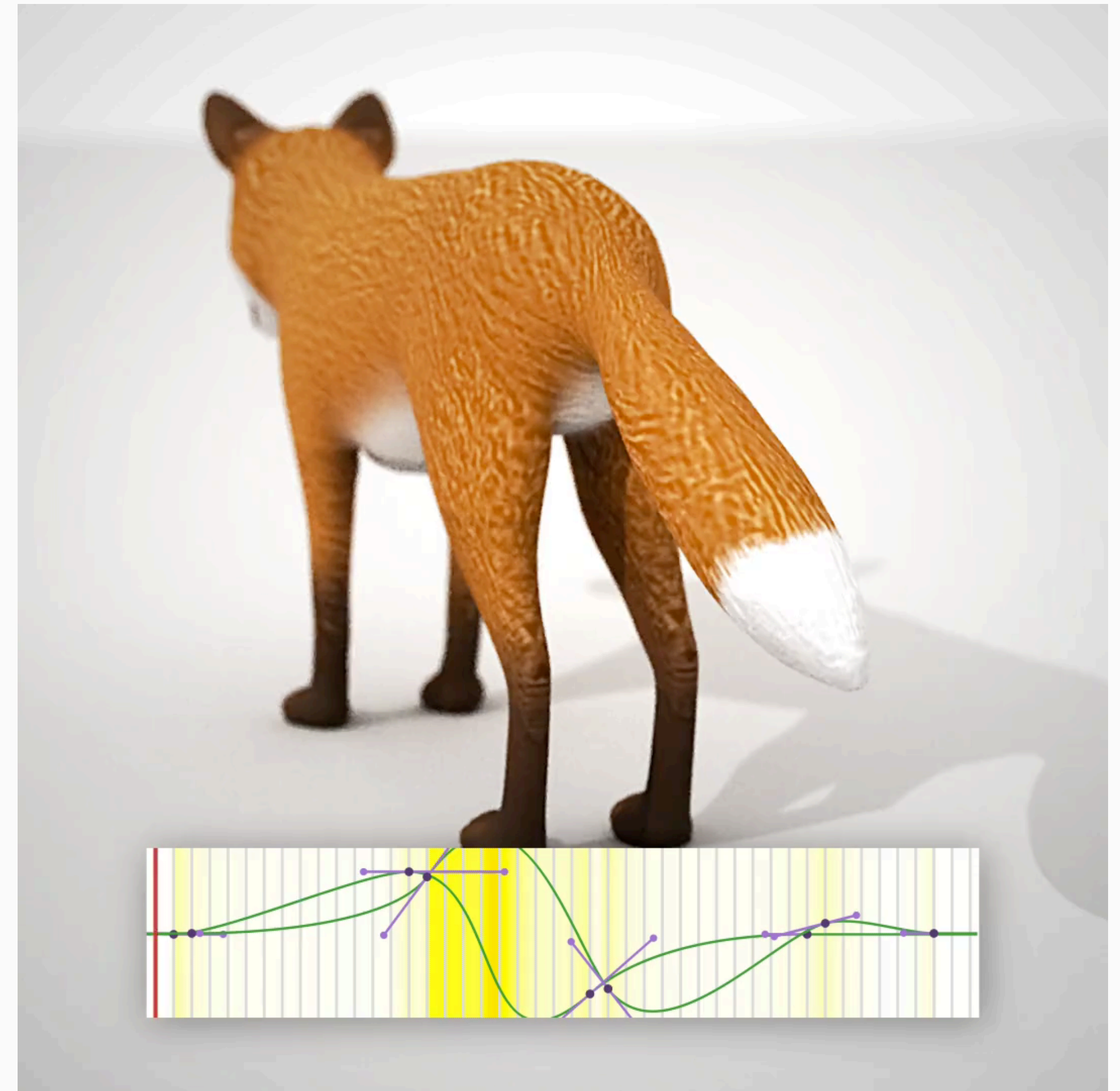
Control by
time-varying cost weight

Control by
regularization weight

Example Usage Scenarios

Example: Fox Tail

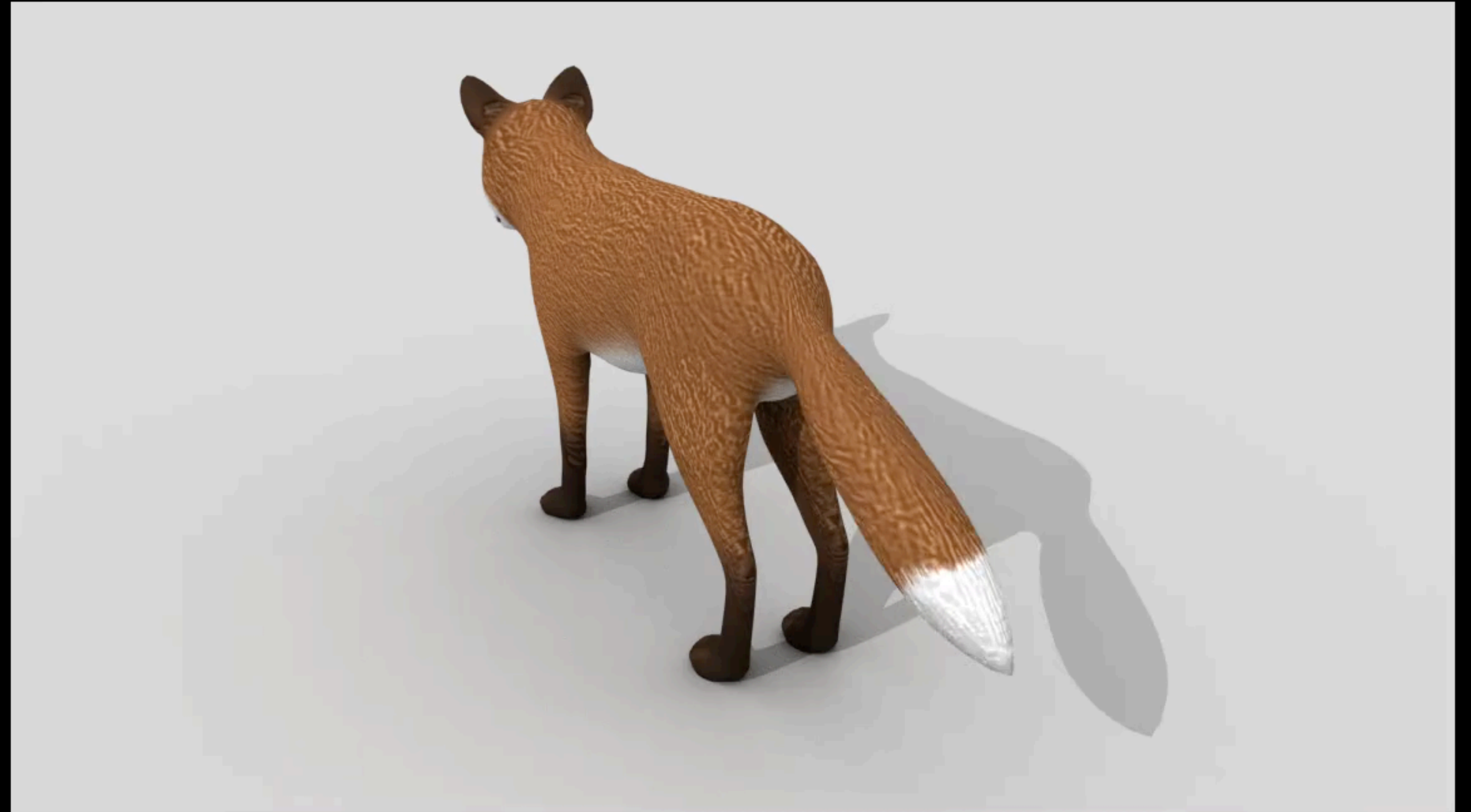
- #parameters: 27
- Rig: Forward kinematics (FK)



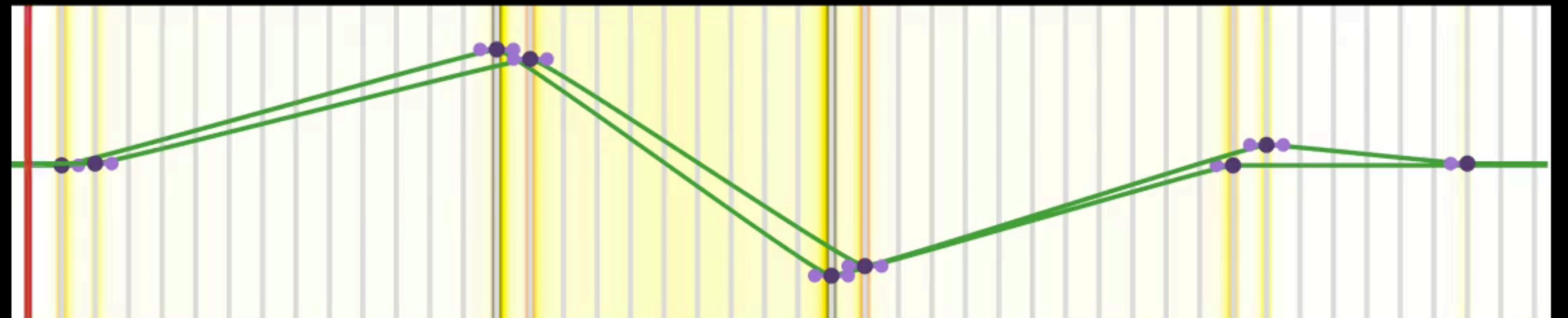
Example: Fox Tail

Initial motion

- Too robotic...



Let's apply the (naïve)
optimization first

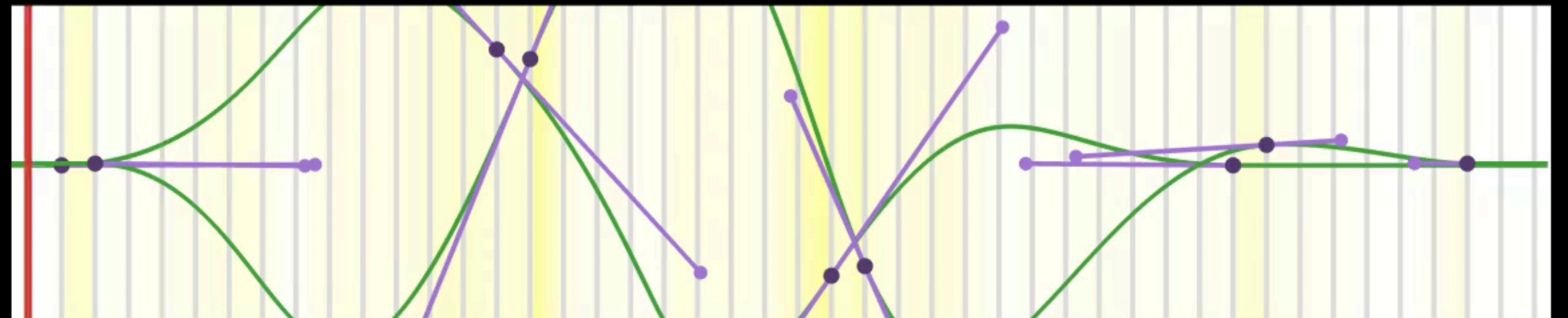
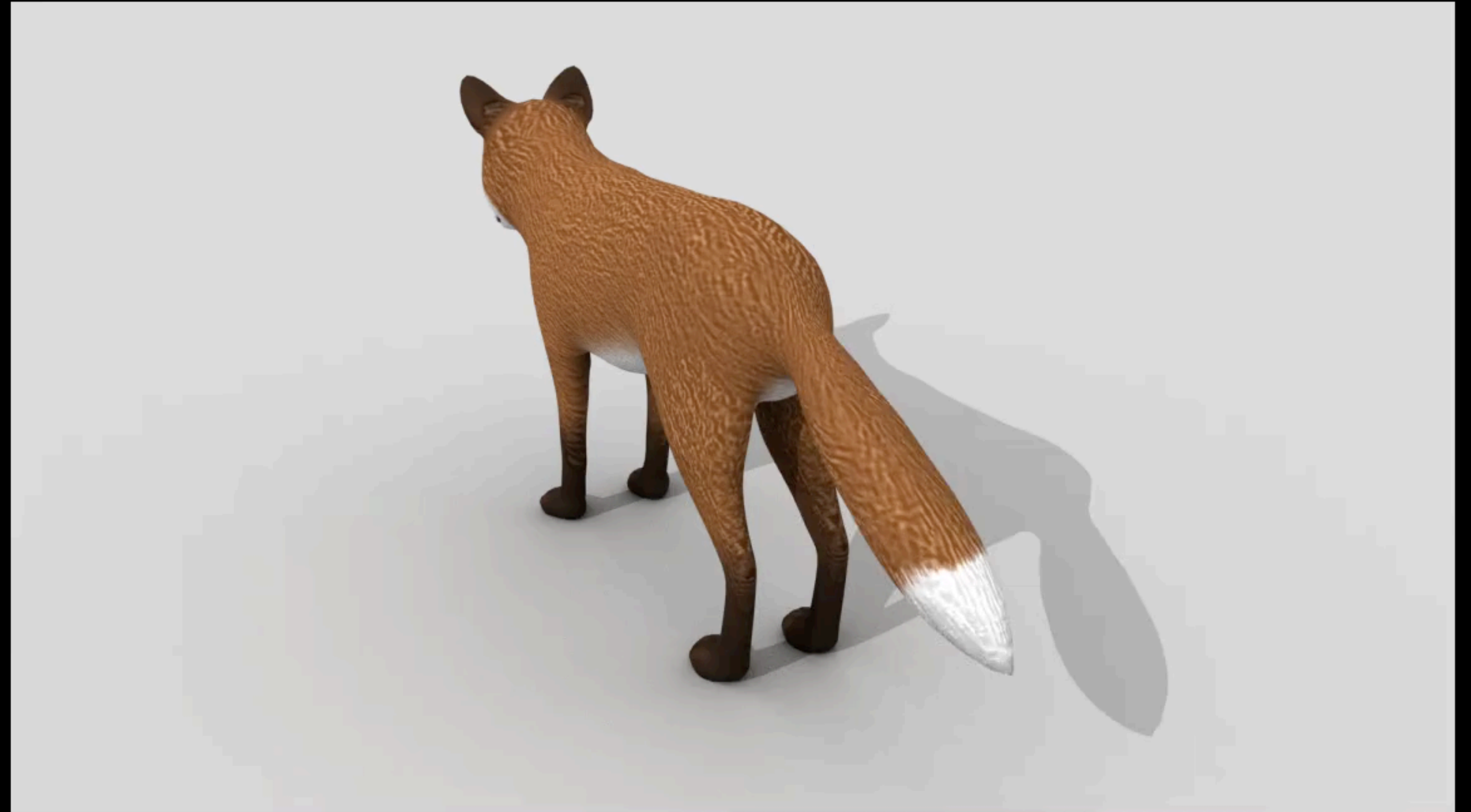


Example: Fox Tail

Naïve optimization

- Too much changed!

Let's control it by regularization



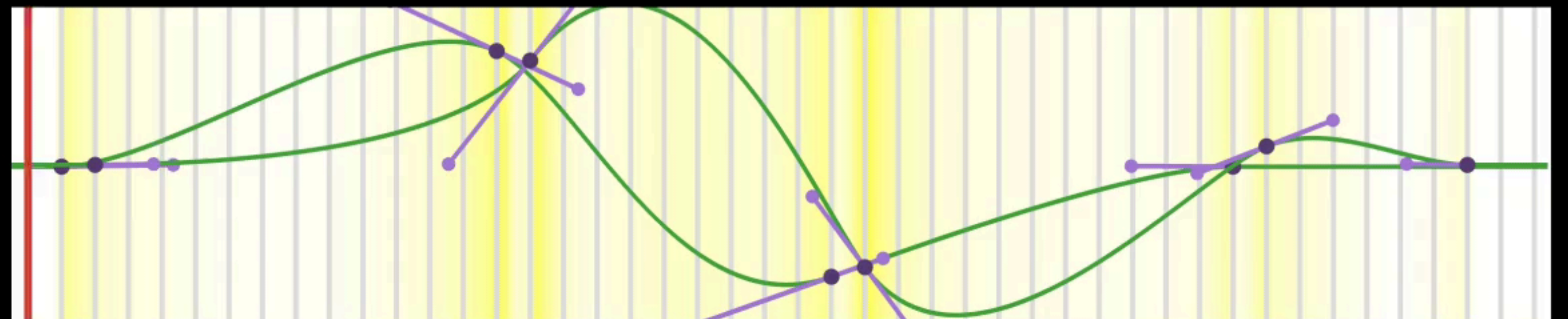
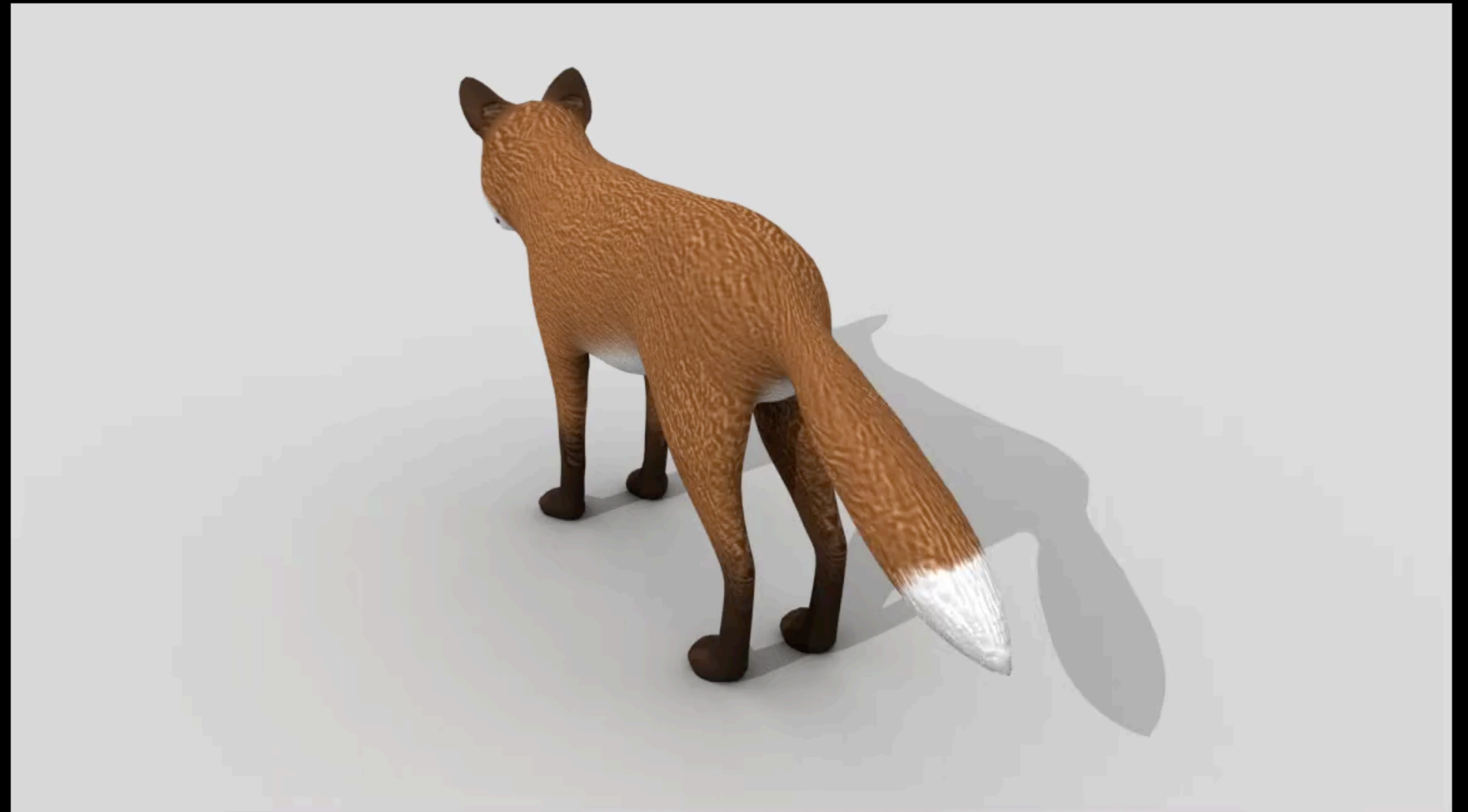
Example: Fox Tail

Optimization
with **regularization**

- Looks good!!

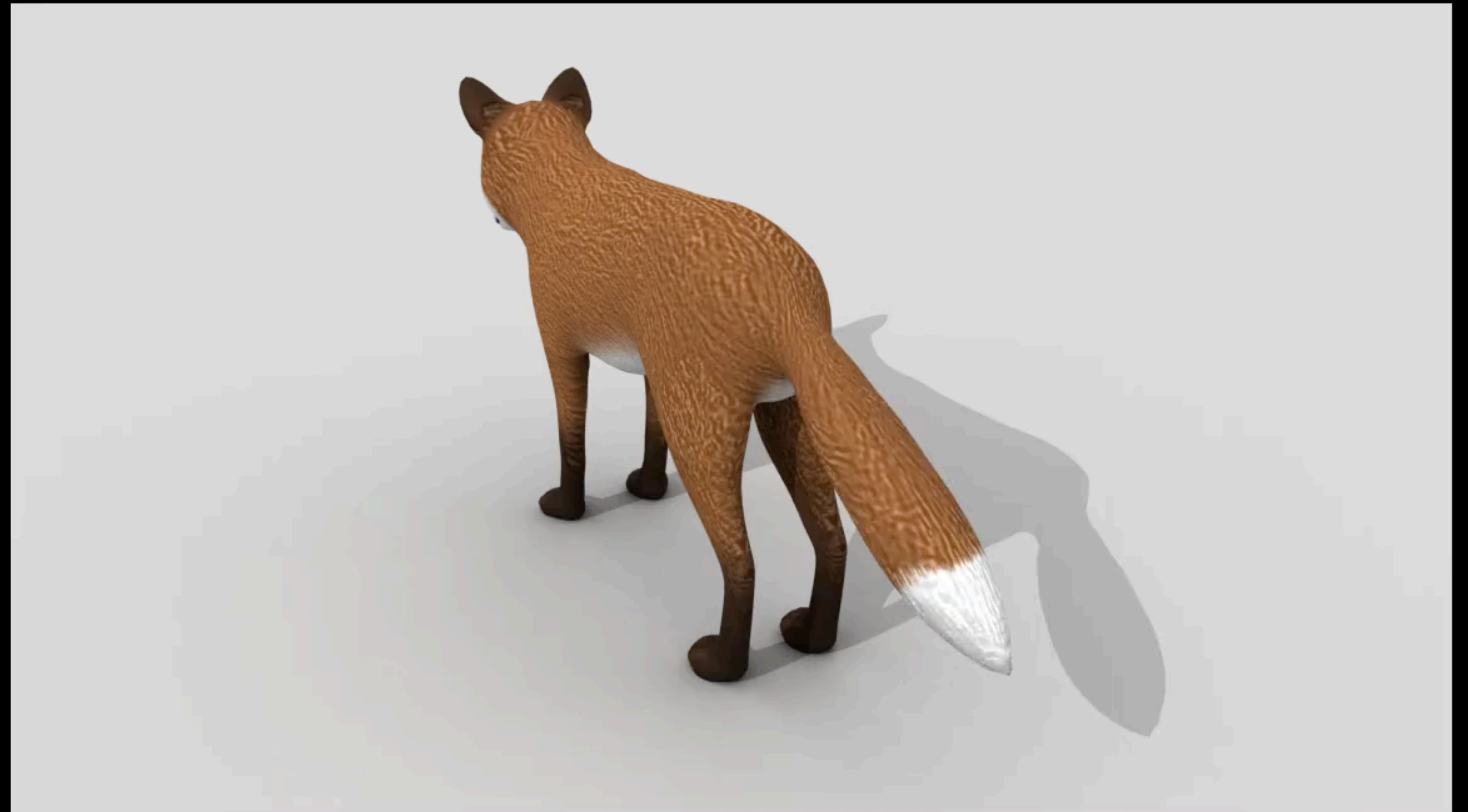
Inspiration:

How about making
the swing speedier?

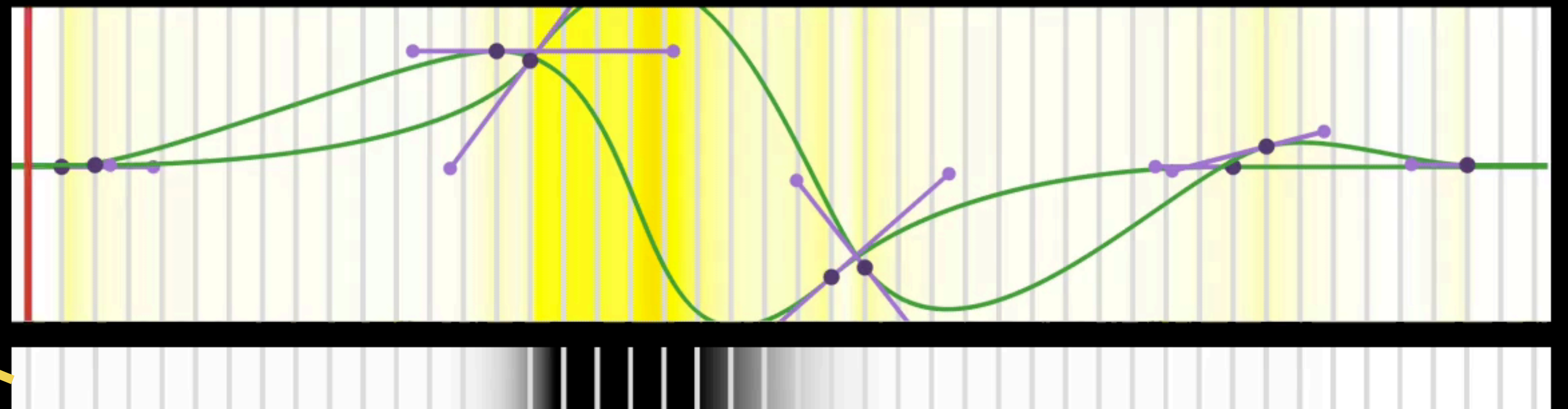


Example: Fox Tail

Optimization
with regularization
& **time-varying cost**



Time-varying
cost weight



Validation: Interview with Professional Animators

Informal Interview

■ Goals

- Validation of our approach (design goals, interactions, etc.)

■ Participants

- 2 professional animators (**A1** & **A2**) who are familiar with 3D character animation authoring

■ Procedure

- Explain our design goals, explain our system features, and ask feedback comments (approx. 1 hour)

Results

■ Editability

- Both **A1** and **A2** loved editability, which was described “*indispensable*” (**A1**)

■ Transparency

- We observed that both **A1** and **A2** **easily understood the concept of optimization** by just seeing the visualizations

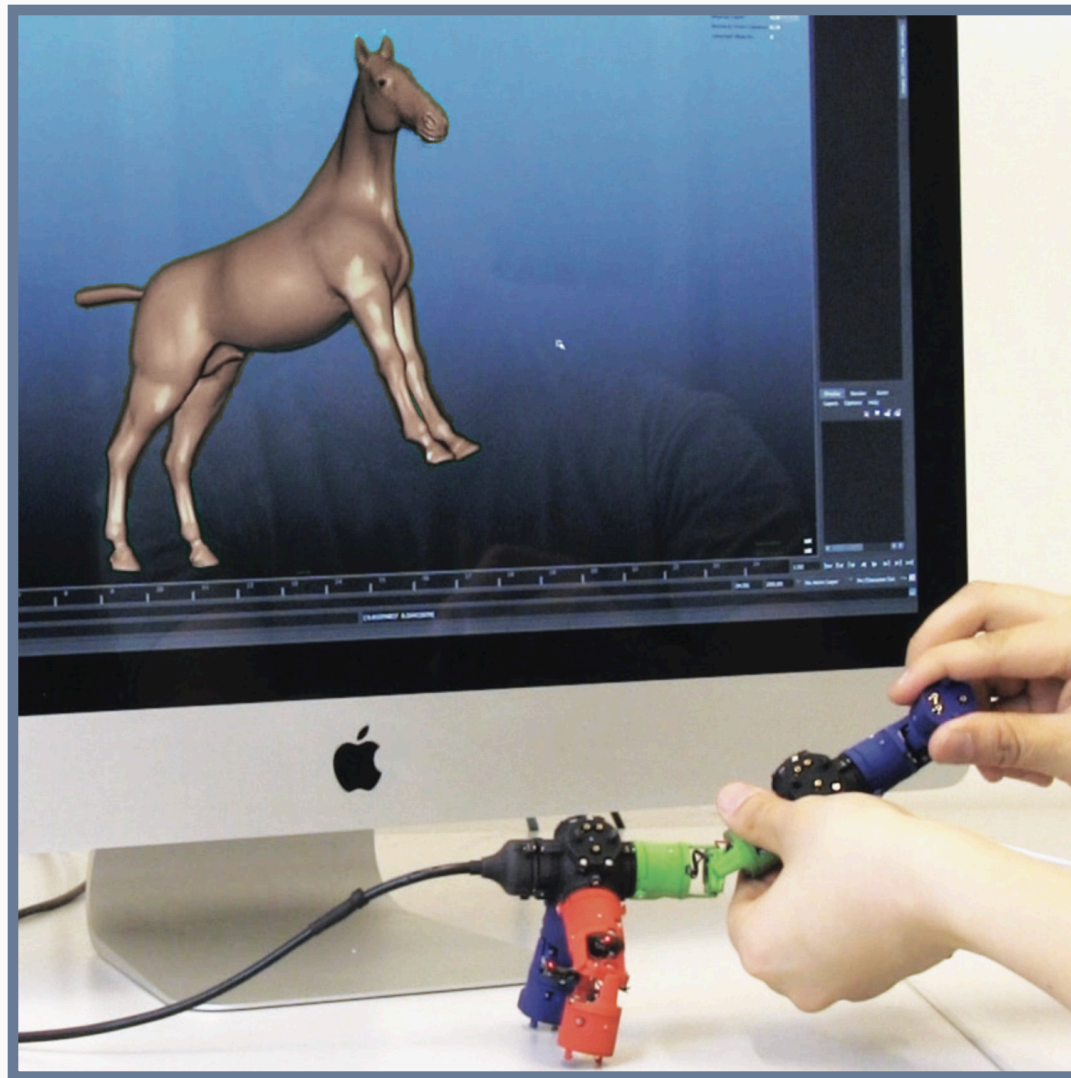
■ Controllability

- All features on controllability were ***strongly appreciated***
- The **interactivity** makes the adjustment “*easier*” and “*less stressful*” (**A1**)

Discussions

Related Work: Motion Editing Techniques

Tangibles



[Glauser+16]

Sketches



[Choi+16]

Principles



[Kazi+16]

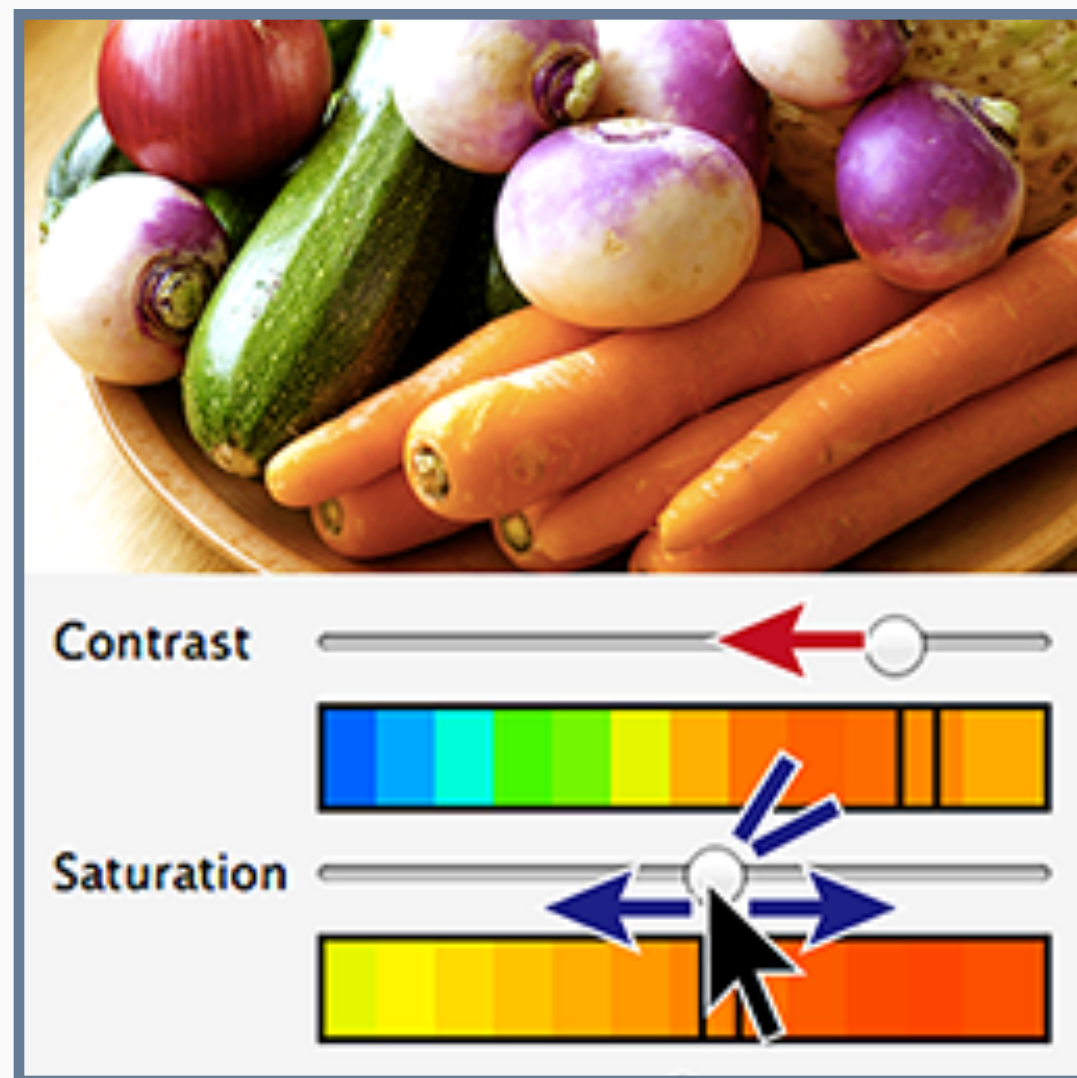
Optimization



Ours

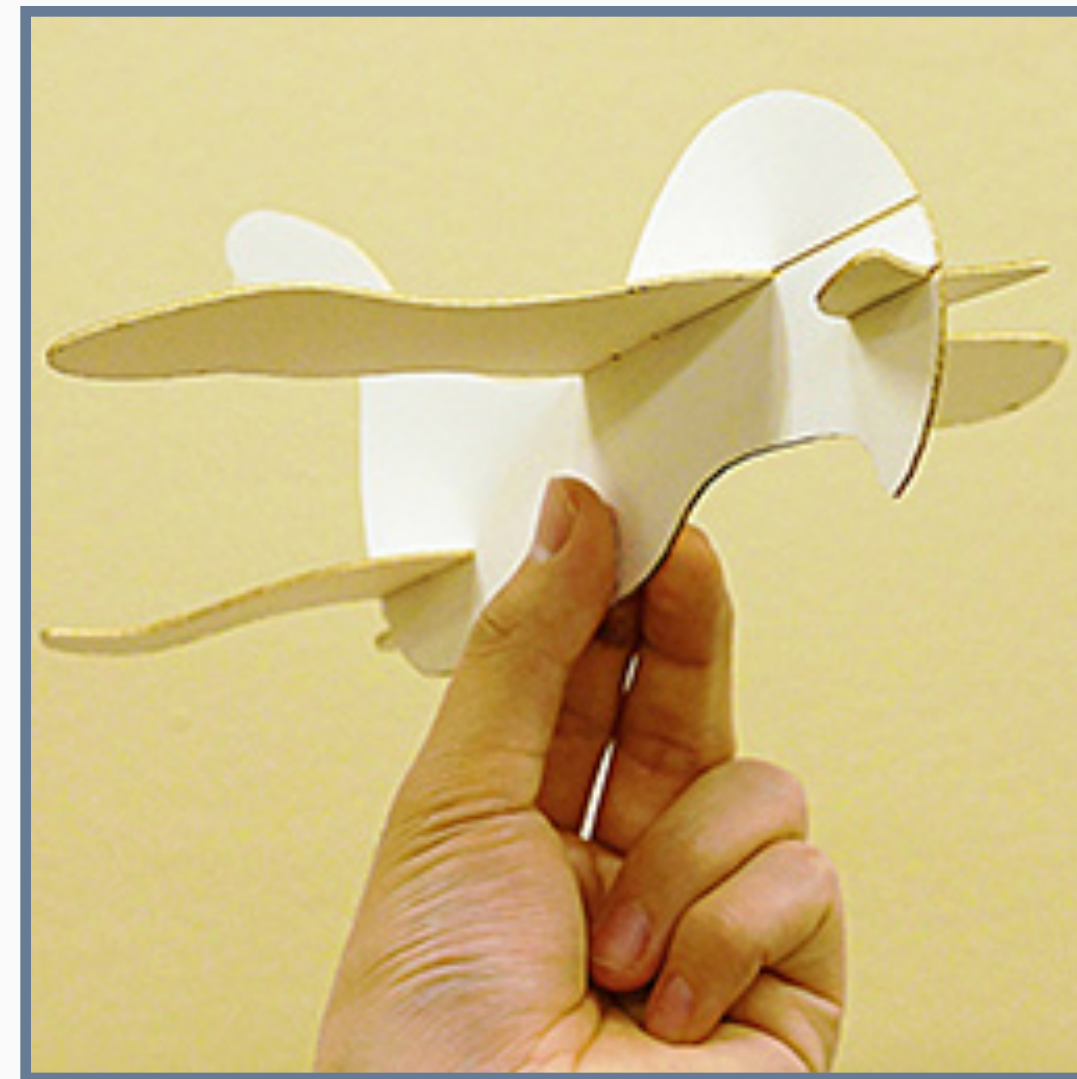
Related Work: Design Interface with Optimization

Photo color adjustment, etc.



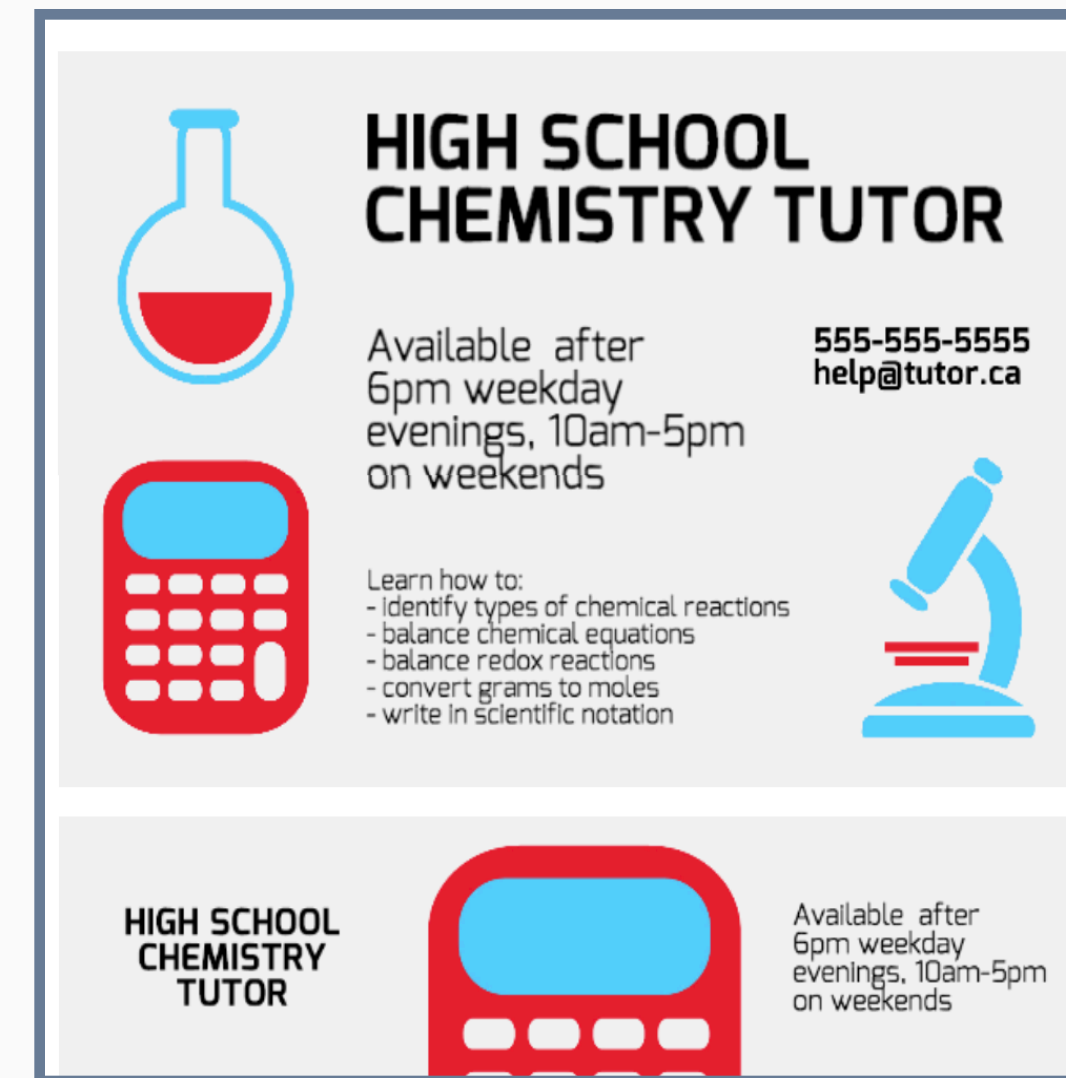
[Koyama+14]

Paper airplanes



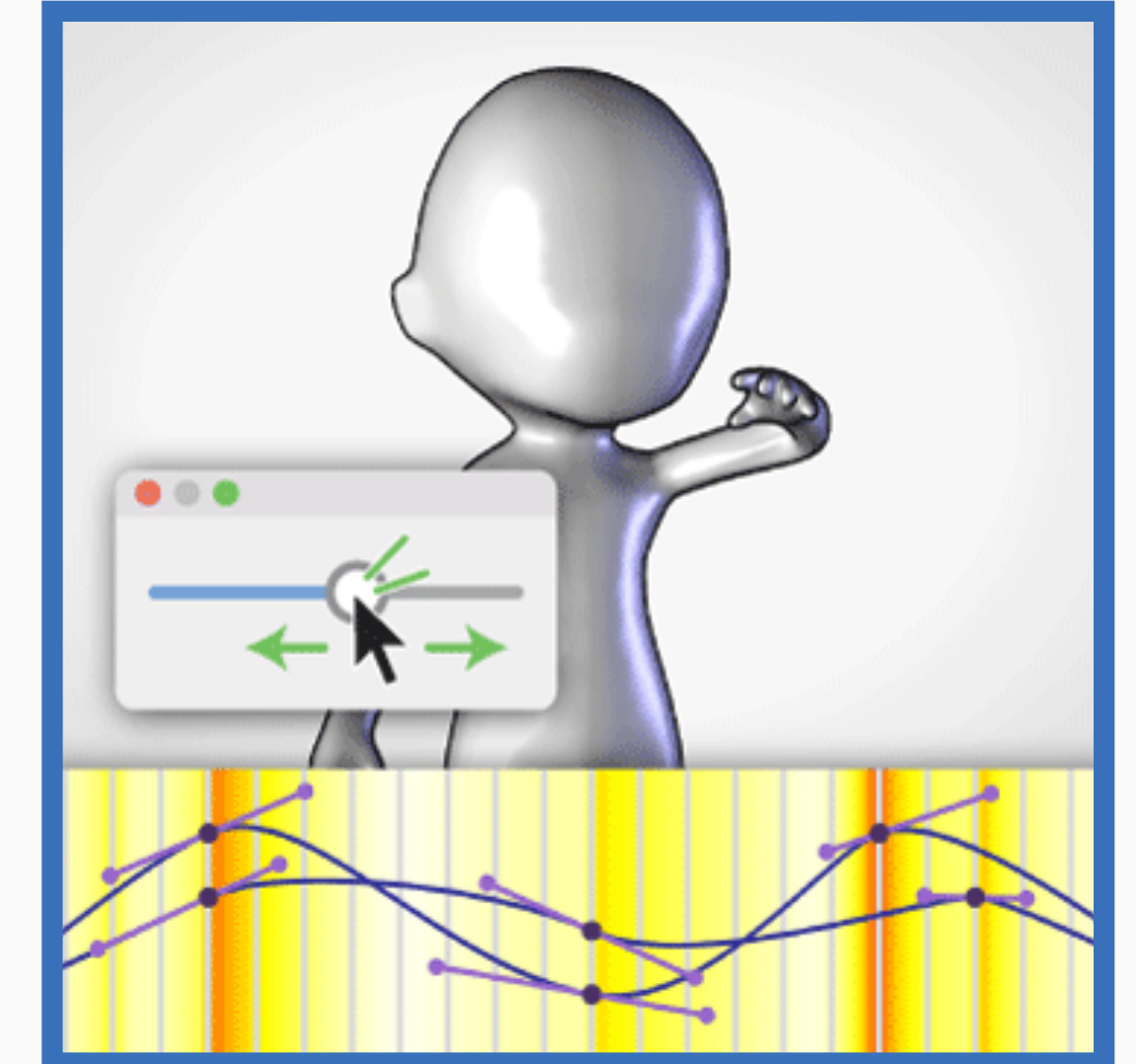
[Umetani+14]

Graphic layout



[O'Donovan+15]

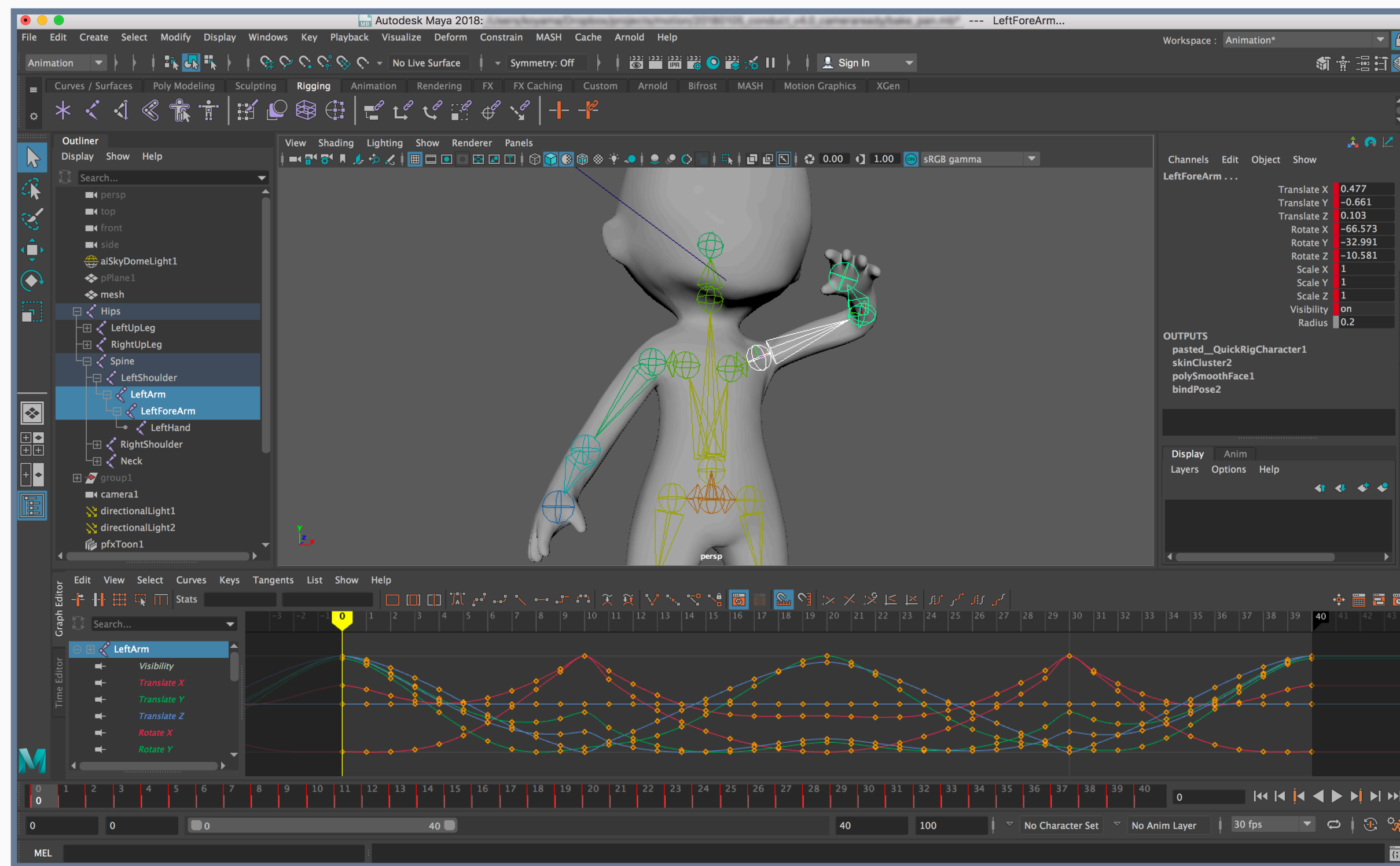
Character motions



Ours

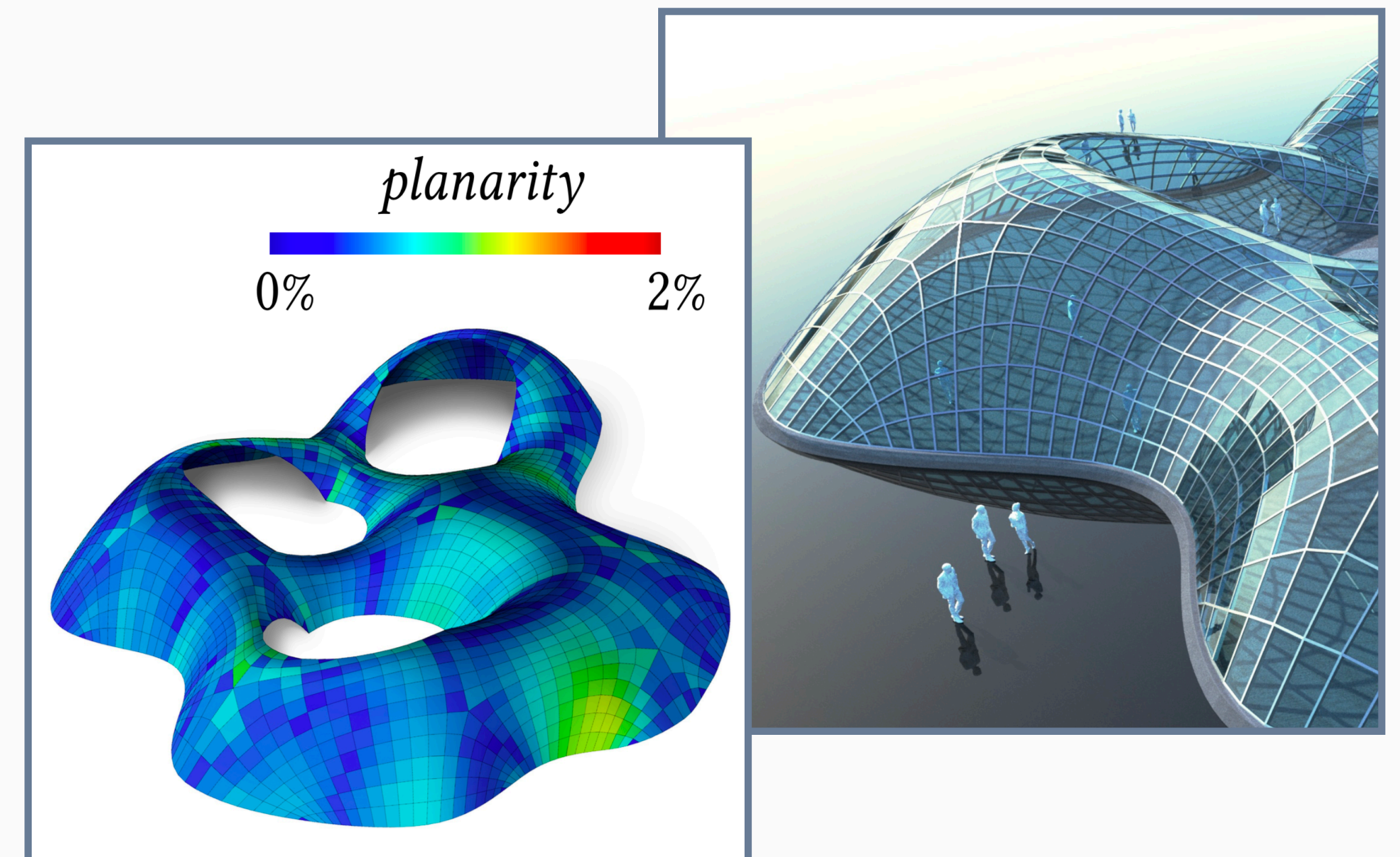
Future Work

Integration to & evaluation in professional workflow



E.g., Autodesk Maya

Applications to other design domains

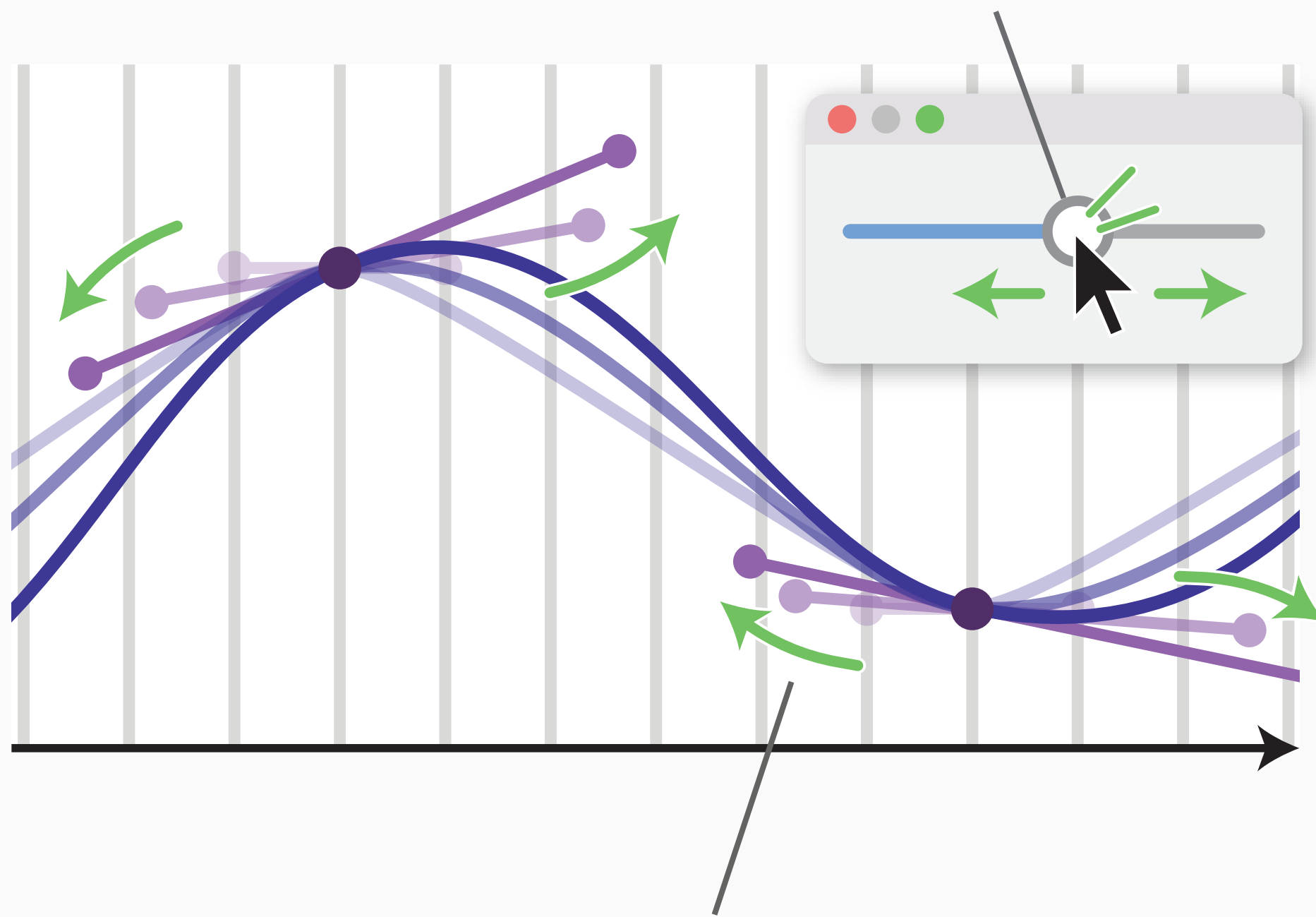


E.g., architecture design
[Kilian+17]

Summary

Optimization-Guided Motion Editing

Interactive control
by animators

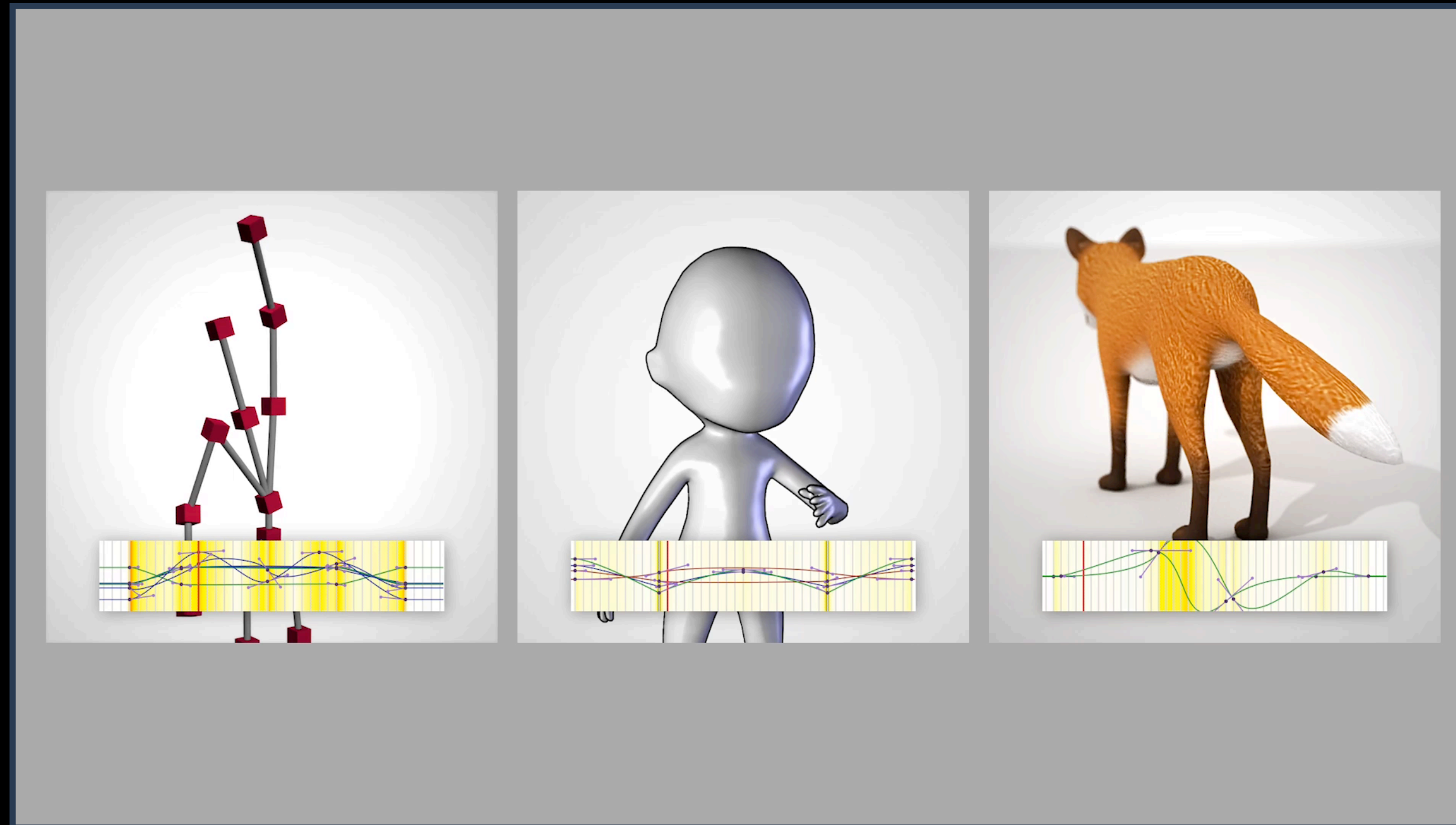


Automatic adjustment
by optimization

- **Optimization** is used for **enhancing manual editing**
- OptiMo allows animators to **interactively control** optimization

<http://koyama.xyz/project/optimo/>

Slides & videos available! We also plan to release our source codes!



Yuki Koyama and Masataka Goto. OptiMo: Optimization-Guided Motion Editing for Keyframe Character Animation. CHI 2018.

OptiMo: Optimization-Guided Motion Editing for Keyframe Character Animation

Yuki Koyama & Masataka Goto

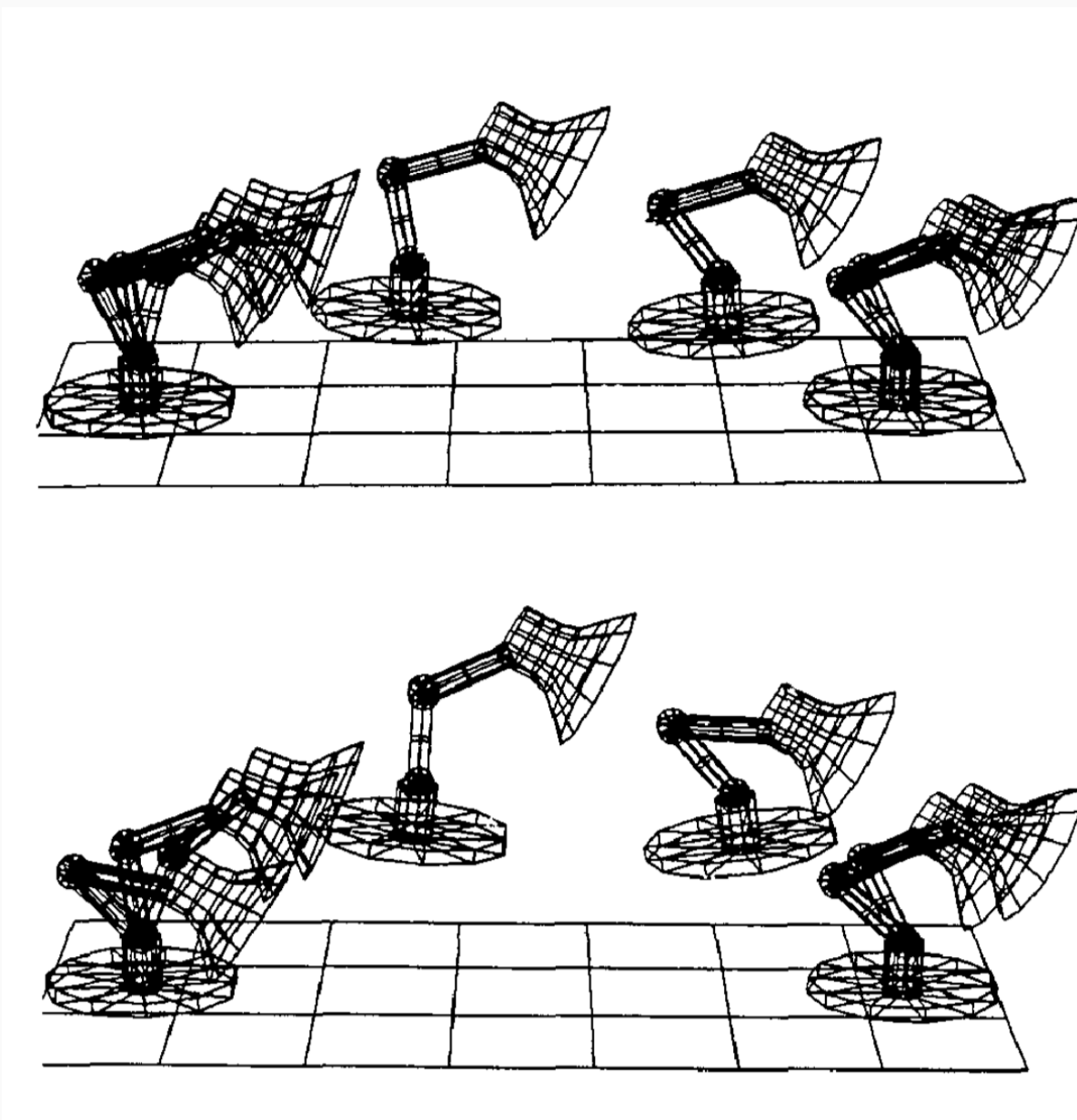


Appendix

Related Work

Motion Editing Techniques — Automatic

Optimization with space-time constraints



[Witkin+88]

Use of learned motion controllers



[Merel+17]

Difficult to reflect artistic intention

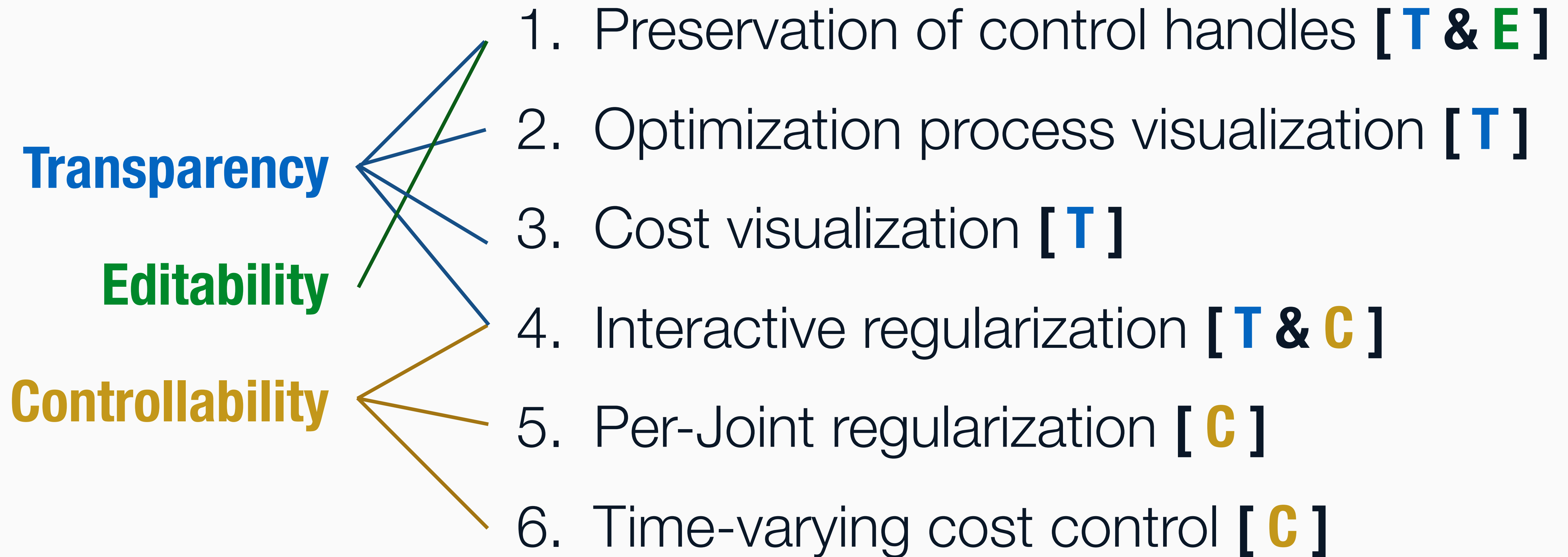
➔ **Lack of controllability**

Resulting motions do not have curve-based data representation

➔ **Lack of editability**

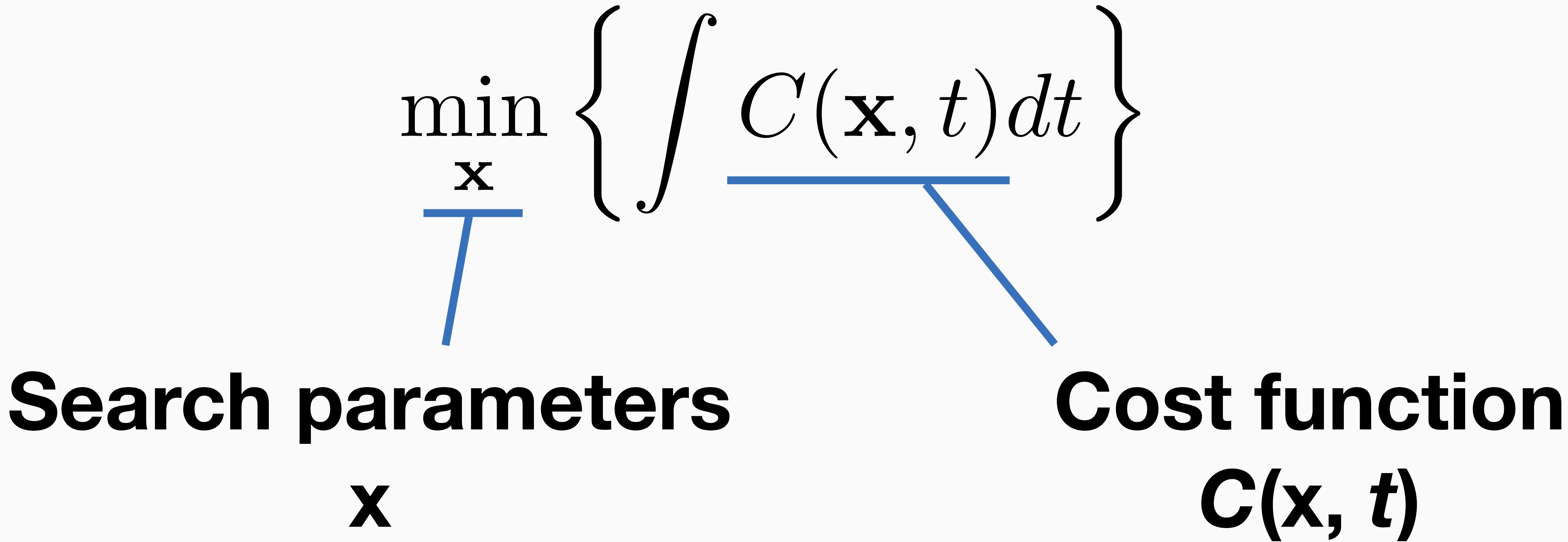
OptiMo's Features

We Designed 6 Features for OptiMo



Algorithm: How Does Optimization Work?

Typical Optimization Formulation

$$\min_{\mathbf{x}} \left\{ \int \frac{C(\mathbf{x}, t) dt}{\quad} \right\}$$


The diagram illustrates the typical optimization formulation. It features the mathematical expression $\min_{\mathbf{x}} \left\{ \int \frac{C(\mathbf{x}, t) dt}{\quad} \right\}$. A blue arrow points from the label 'Search parameters' to the variable \mathbf{x} in the minimization term. Another blue arrow points from the label 'Cost function' to the fraction $\frac{C(\mathbf{x}, t) dt}{\quad}$ inside the integral.

Search parameters

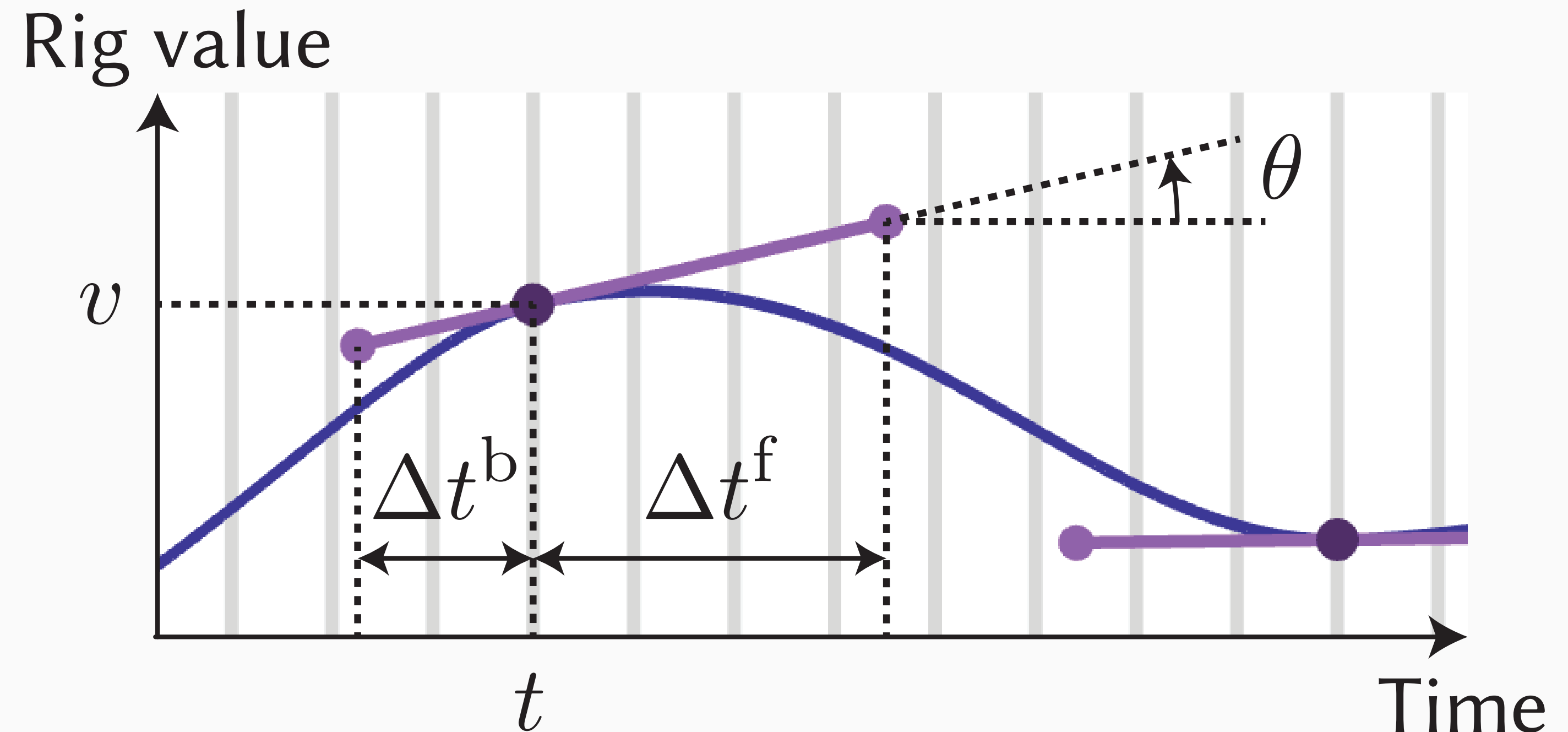
\mathbf{x}

Cost function

$C(\mathbf{x}, t)$

Parametrization (Search Space)

Use the control handle parameters (that animators otherwise manipulate manually) as the search parameters of the optimization



➡ This ensures **editability** (and also transparency)

Cost Function: Our (Current) Choice

$$C(\mathbf{x}, t) = \sum_{j \in \mathcal{J}} \|\boldsymbol{\tau}_j(\mathbf{x}, t)\|^2$$

- Is well-established and used in computer graphics and robotics (c.f. [Safonova+04])
- Is a summation of necessary **torques** (rotational forces) on each joint
- Is useful for **avoiding physically unfeasible motions**

Validation: Interview with Professional Animators

Comments on Editability

- Both **A1** and **A2** loved editability
 - **A1**: Editability is “*indispensable*”
- This comment especially validates our parametrization:
 - **A2**: It is acceptable for “*the [tangent] handles to change,*” but “*the keyframes (key points) should not be change*” by optimization

Comments on Transparency

- Both **A1** and **A2** loved the visualization features
- **A1**: The process visualization “is *very interesting*”
- We observed that both **A1** and **A2** **easily understood the concept of optimization** by seeing the visualizations

Comments on Controllability

- All features on controllability were **strongly appreciated**
- **A2:** *“I would be puzzled if I can’t use this”*
- The **interactivity** of control features was appreciated
- **A1:** These features could make adjustment **“easier”** and **“less stressful”**

Comments on Usages in Production

- The current system is not perfect yet, but it could be useful...
- **A2**: For “*sub-characters’ motion*,” or
- **A1**: In “*projects with only limited time*”
- Both **A1** and **A2** got very excited about the (production-level) quality of the fox tail animation
- **A2**: “*It is possible to provide it by hand, but it requires much tweaking to make it (the motion) look natural*”

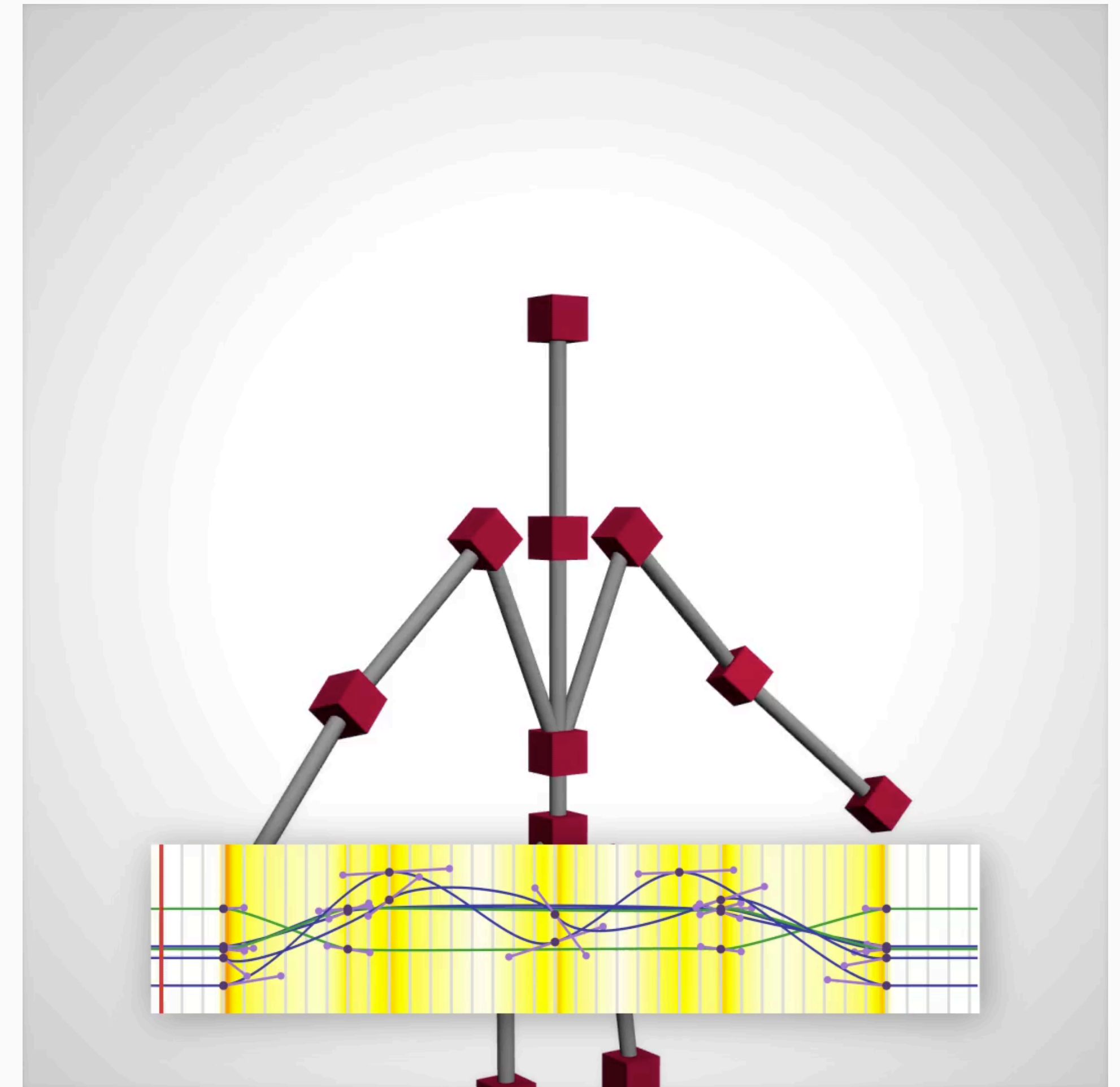
Other Comments

- It would be nice if **motion styles** could be handled in the cost function as well as physics
- **A2:** “*Each art project has a specific style of motions*” and “*it would be nice if it automates [the process of applying such styles]*”

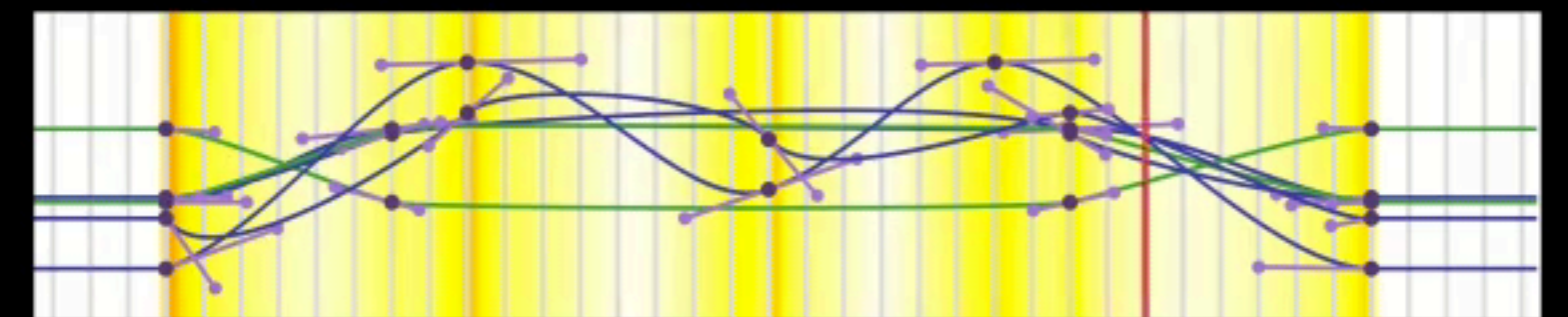
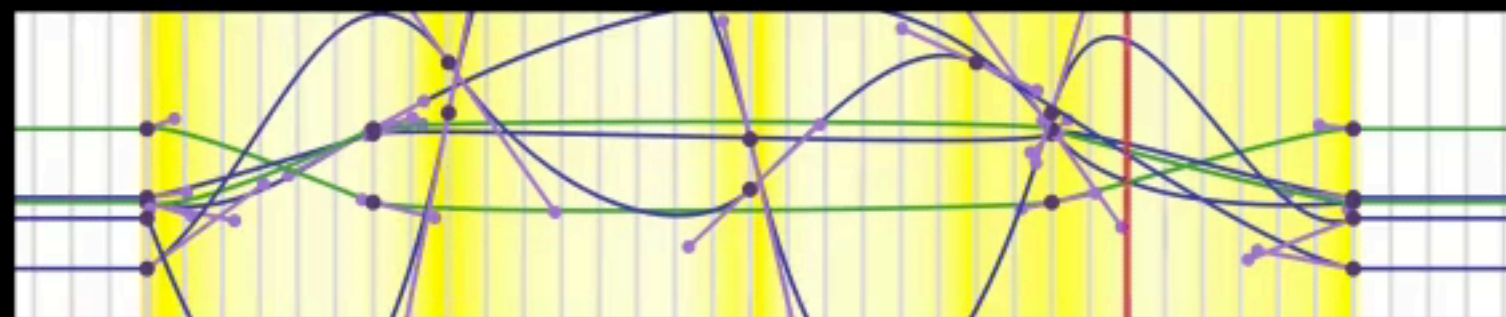
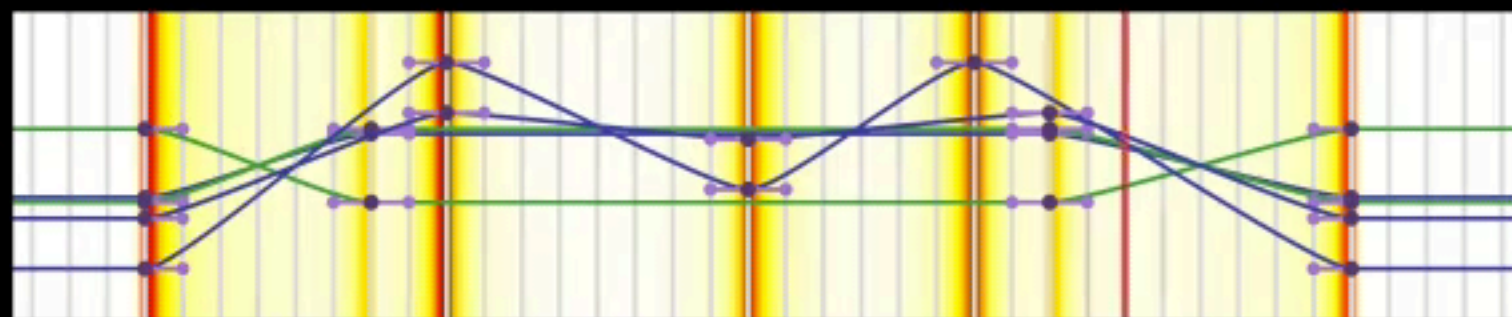
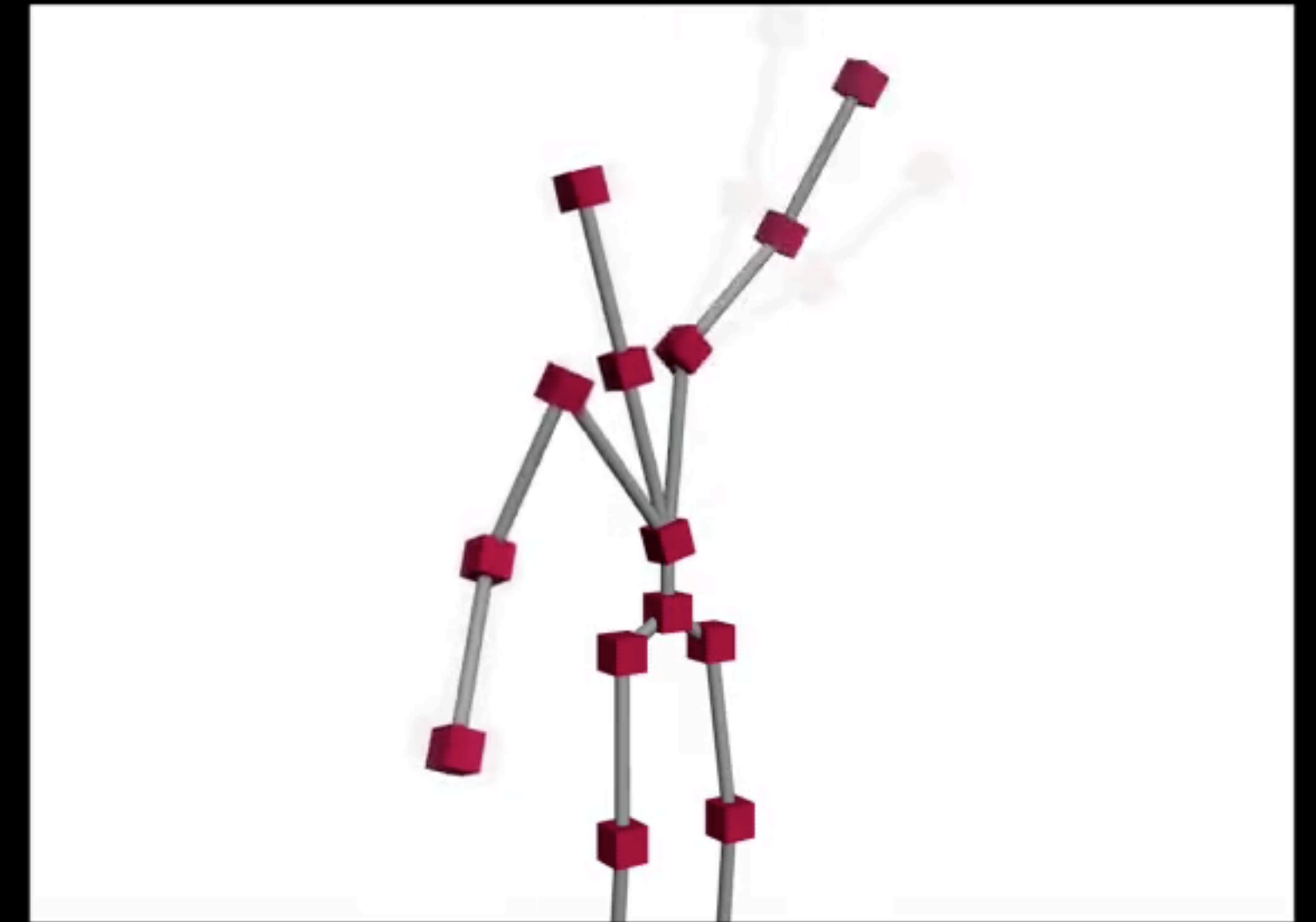
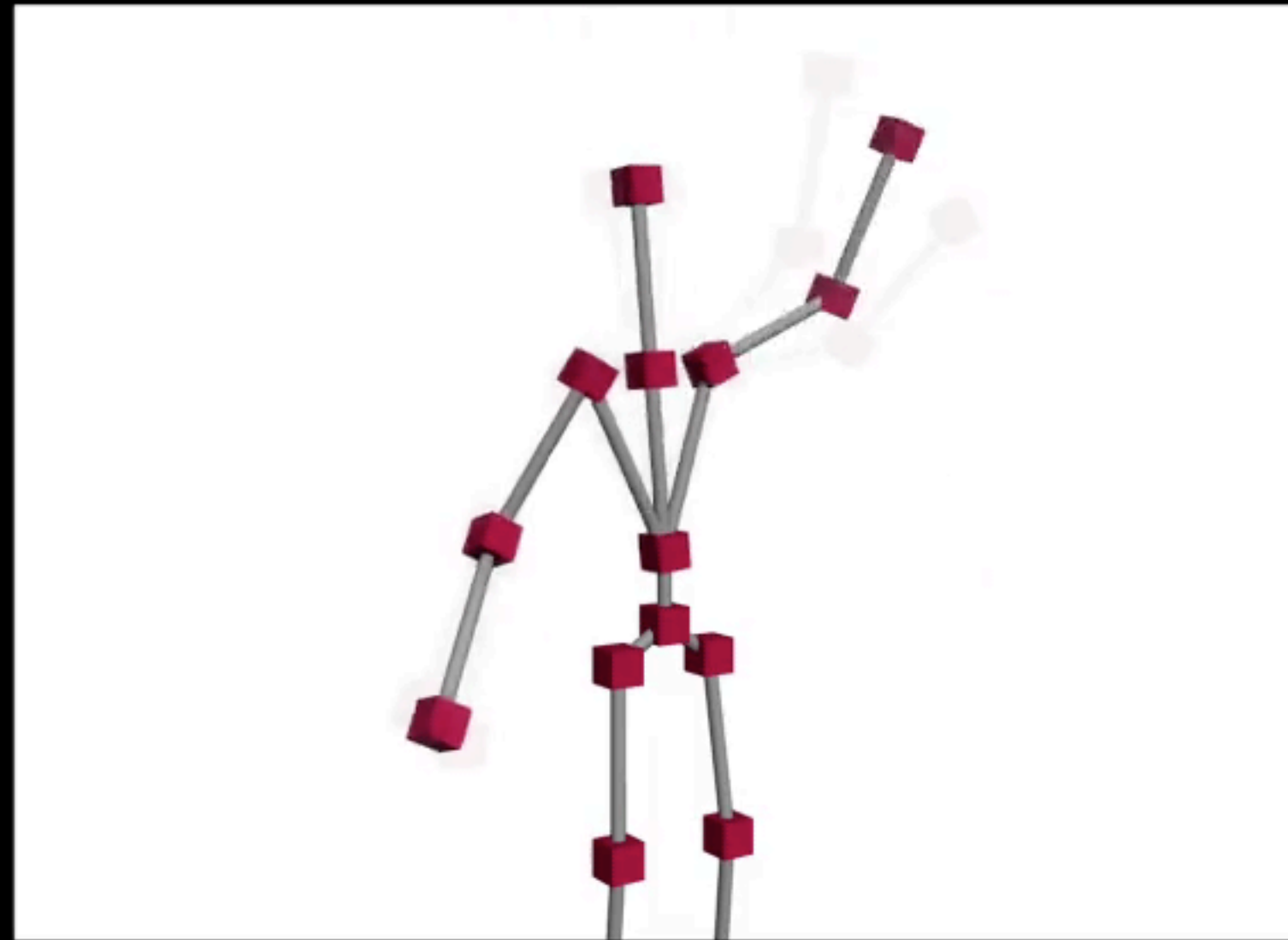
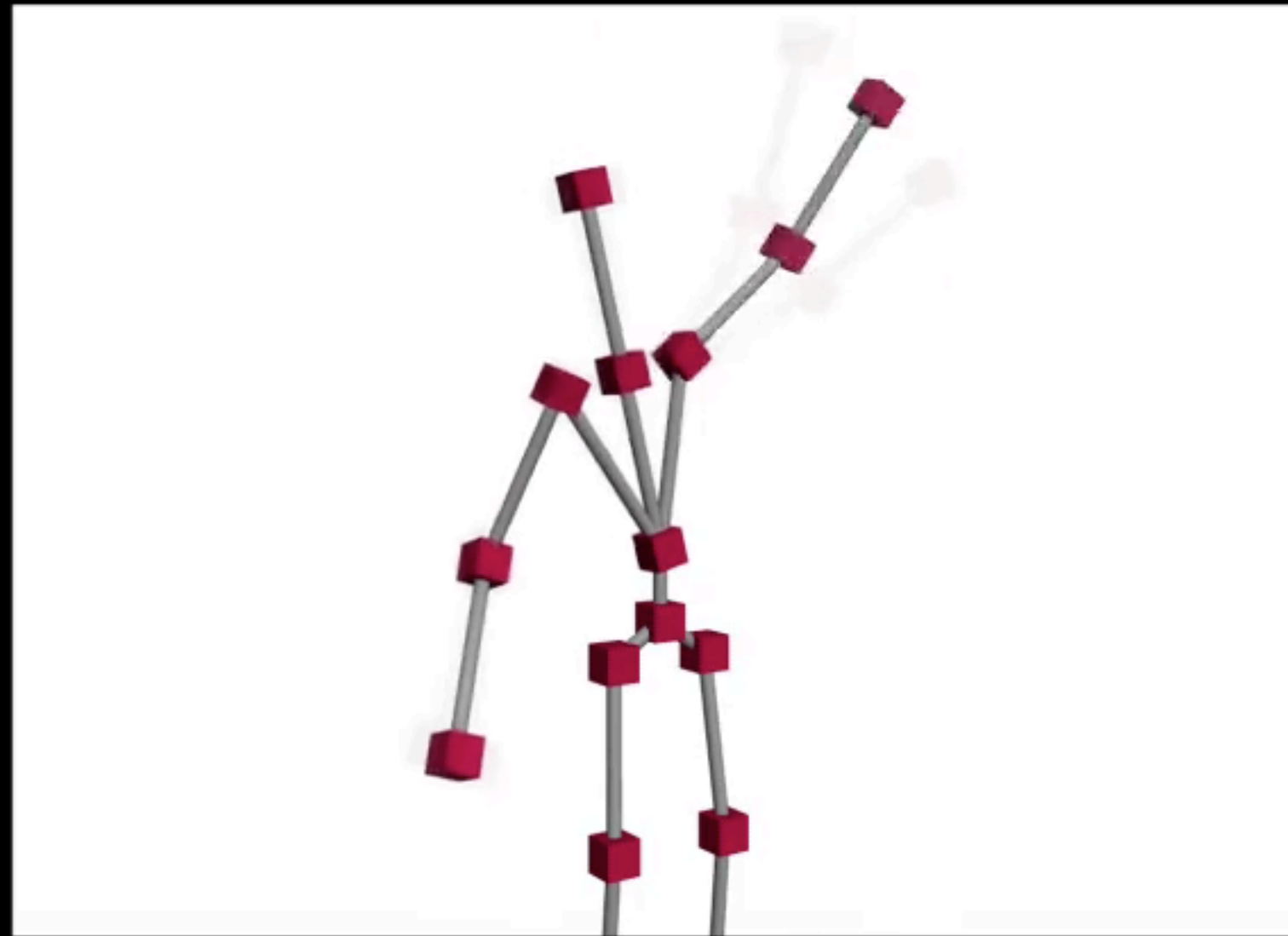
Example Usage Scenarios

Example: Stick Figure Waving His Arm

- #parameters: 78
- Rig: Forward kinematics (FK)



Example 1: Stick Figure Waving His Arm



Initial

Too robotic...

Naïve opt.

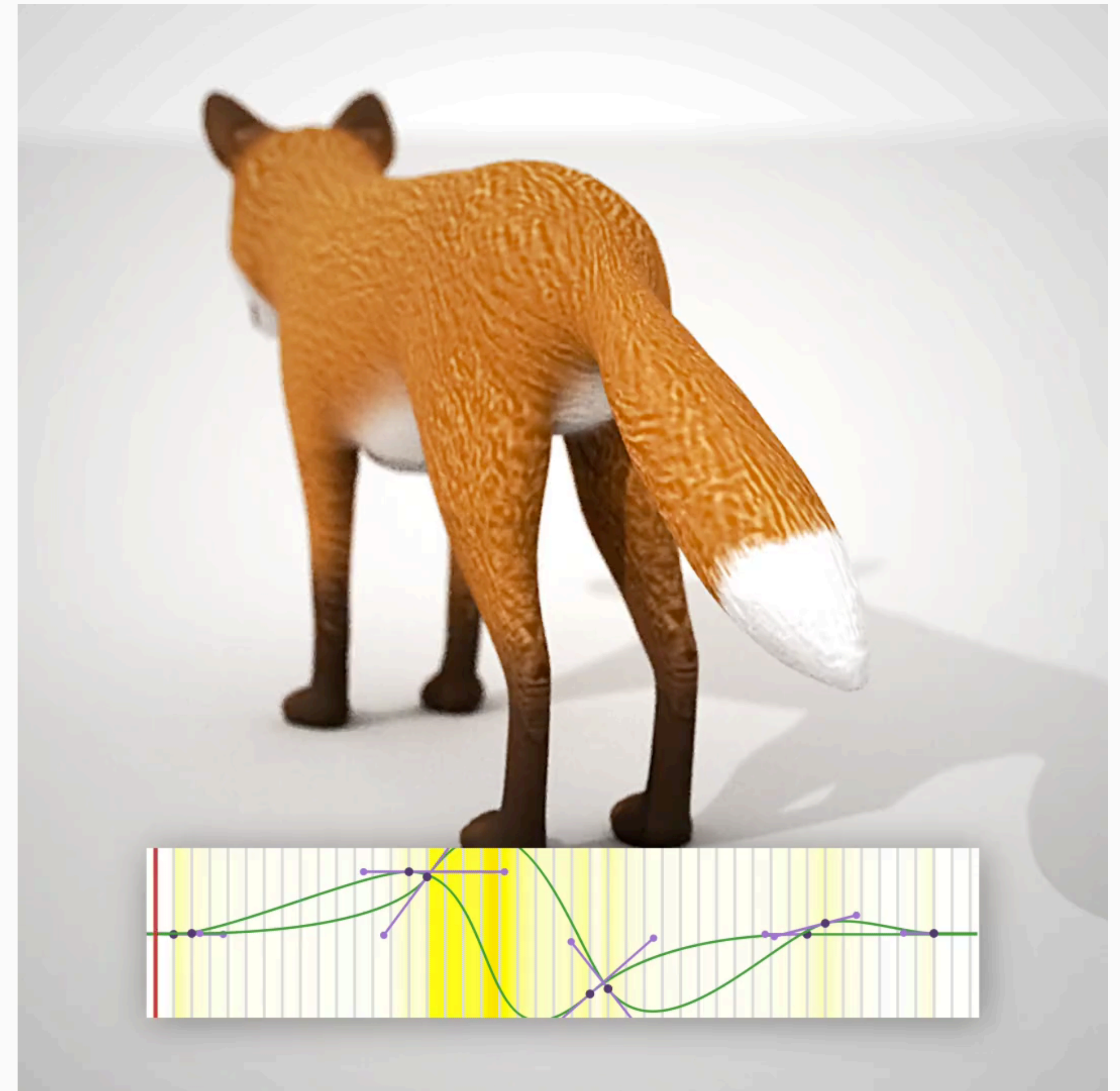
The spine/elbow bend wrongly...

Controlled opt.

Looks natural :)

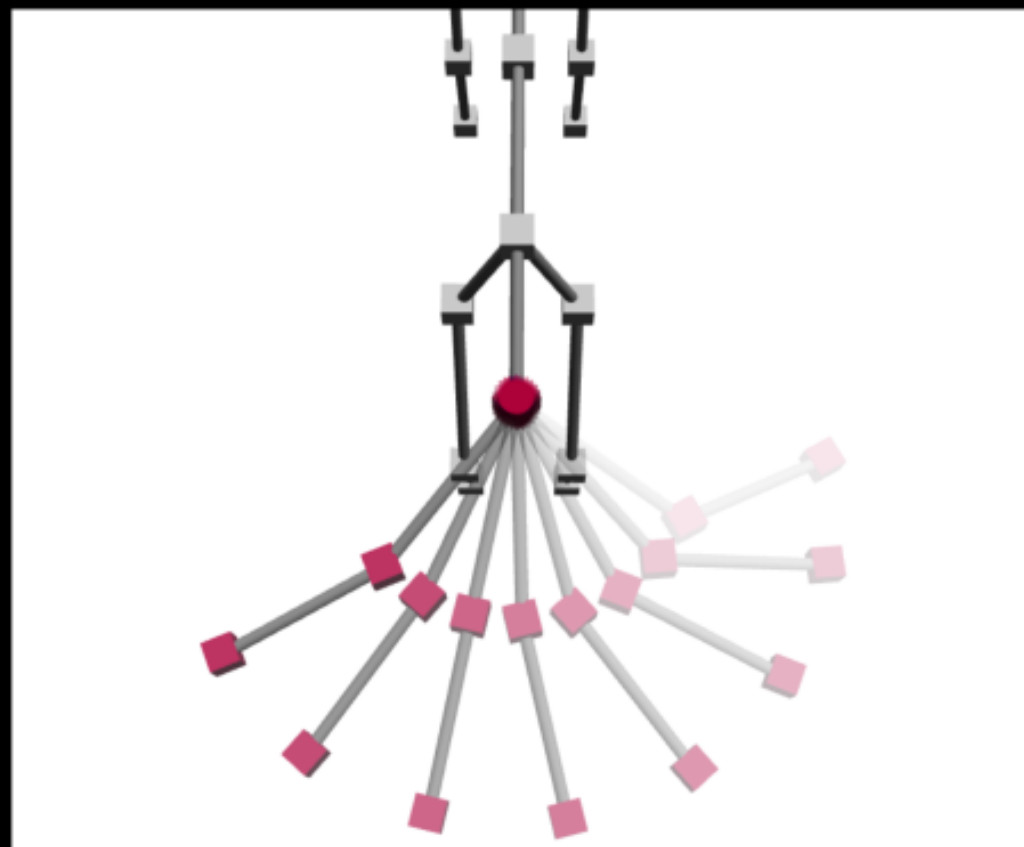
Example: Fox Tail

- #parameters: 27
- Rig: Forward kinematics (FK)

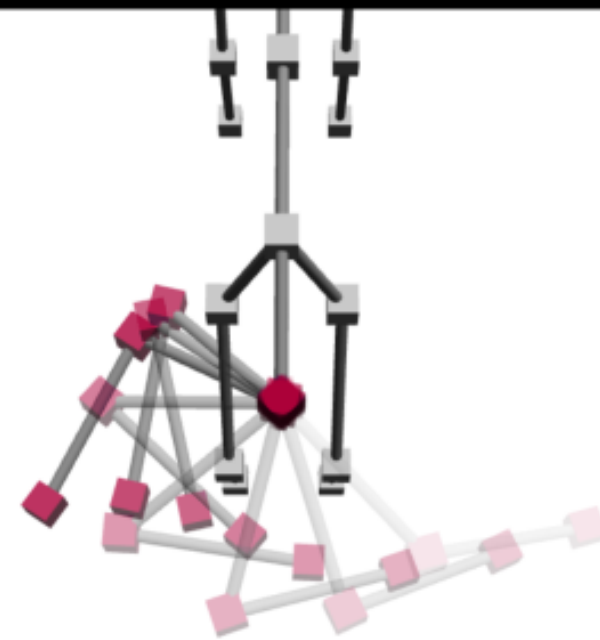


Example: Fox Tail

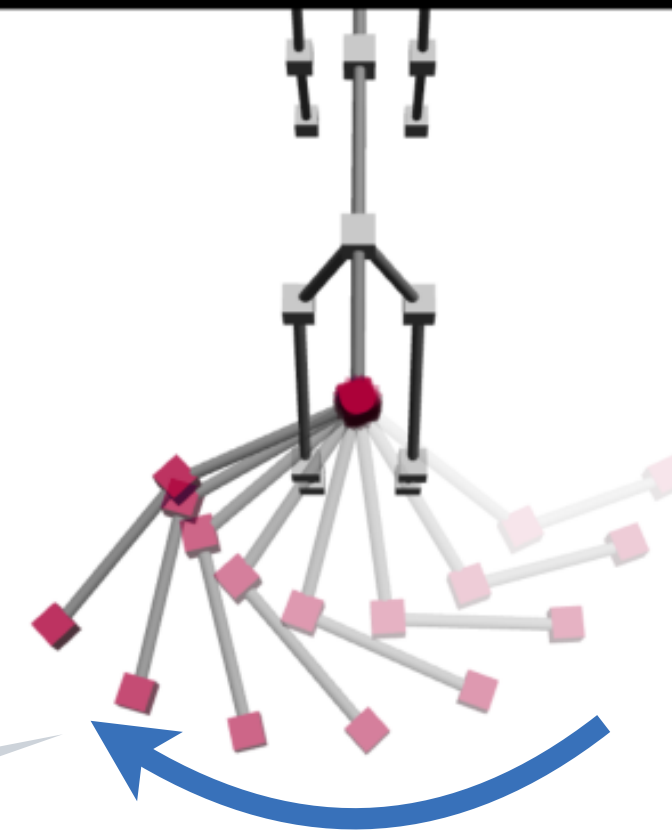
**Initial
motion**



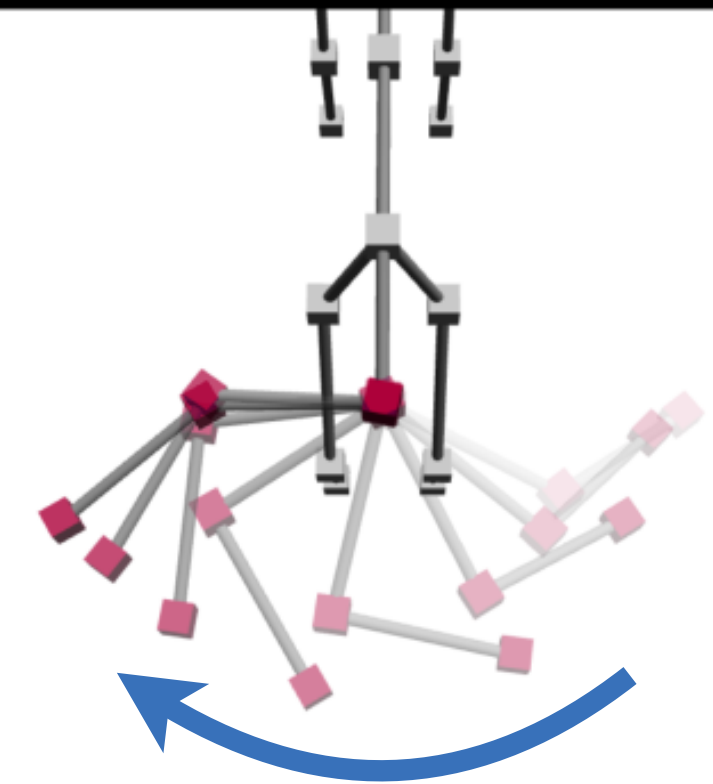
**Naïve
opt.**



**Controlled
opt. #1**



**Controlled
opt. #2**



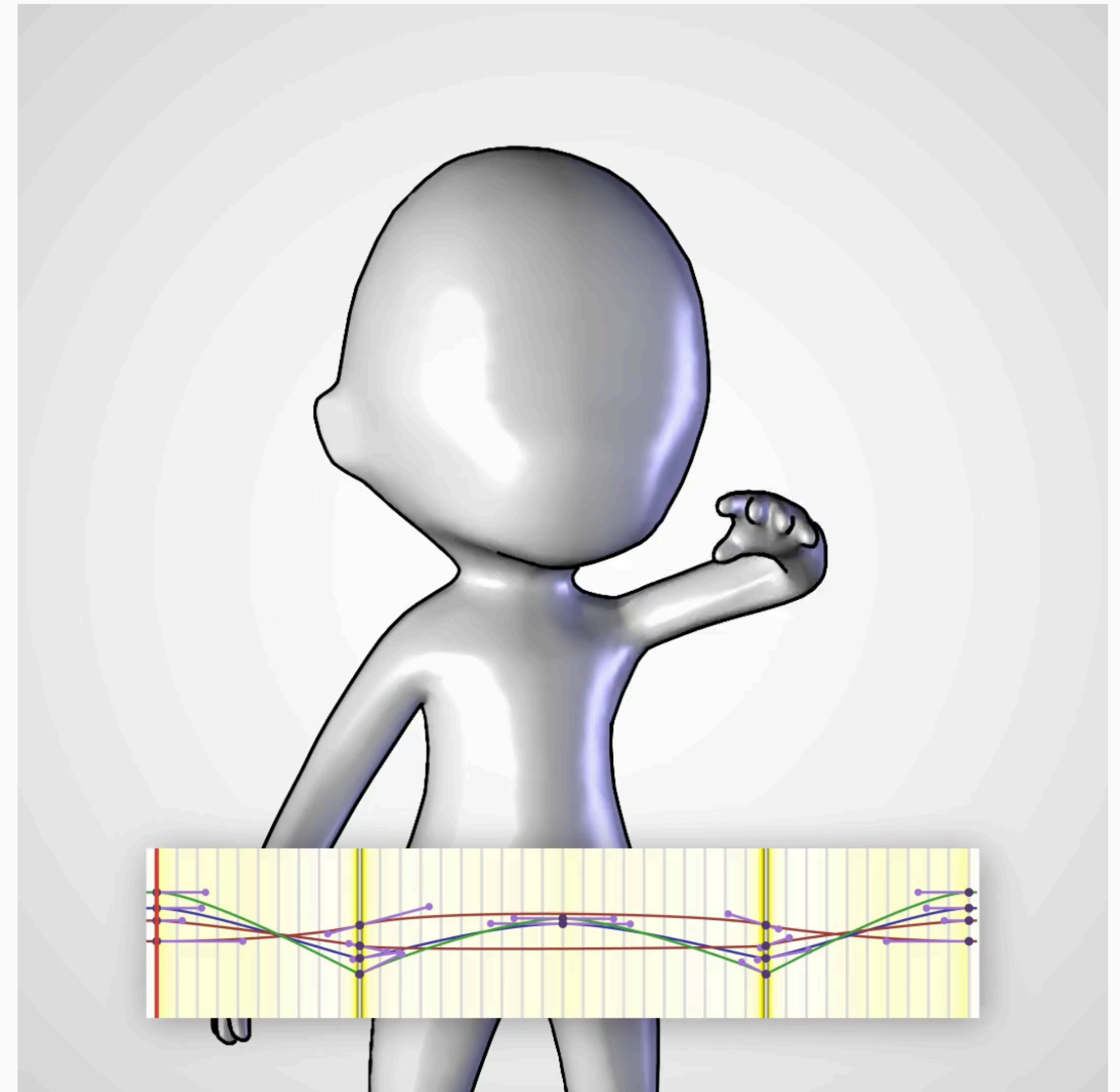
**Inspiration:
Let's make the swing speedier**

Time-varying cost control

Example: Music Conductor

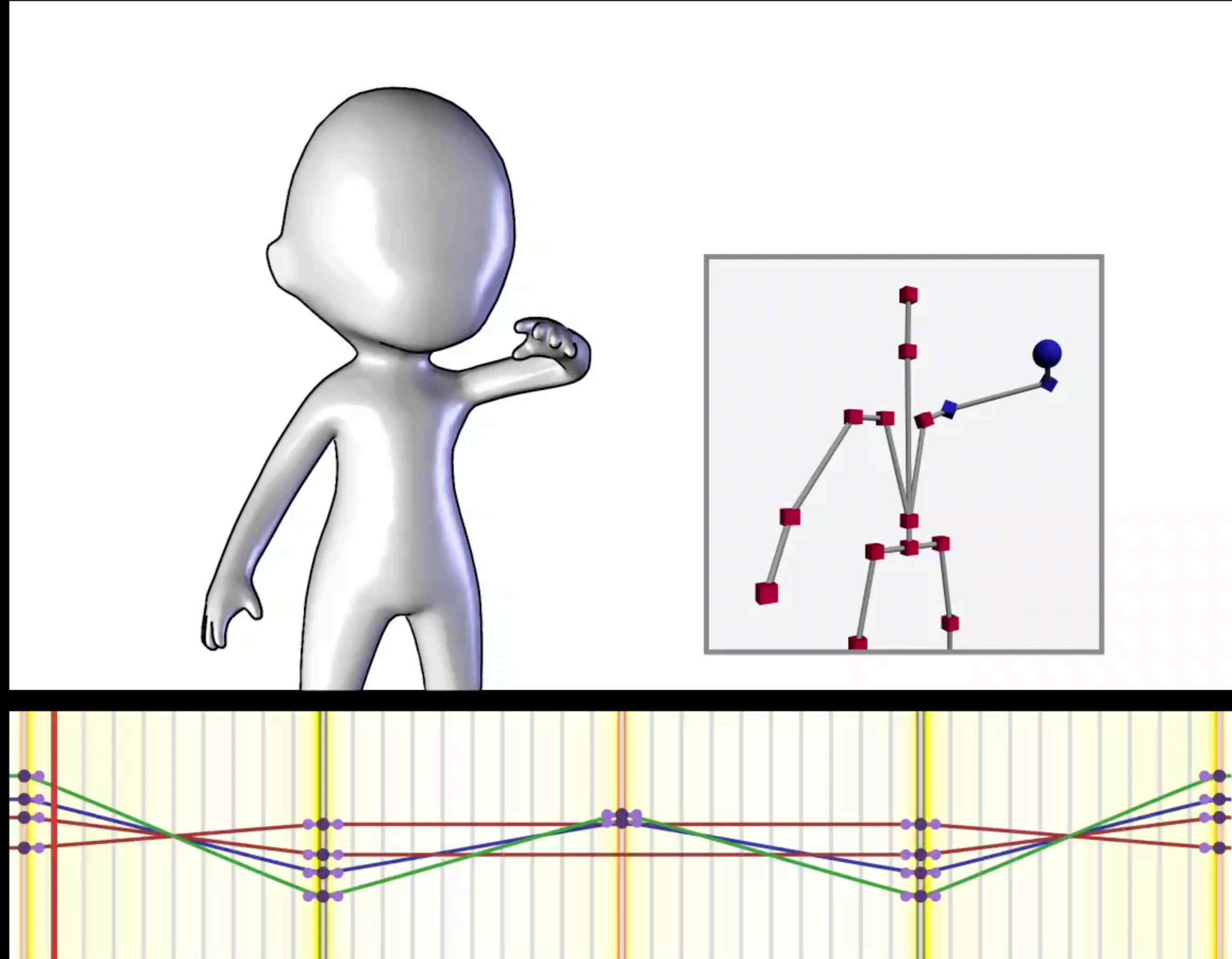
- #parameters: 54
- Rig: Forward kinematics (FK) and inverse kinematics (IK)

The next slide contains audio!



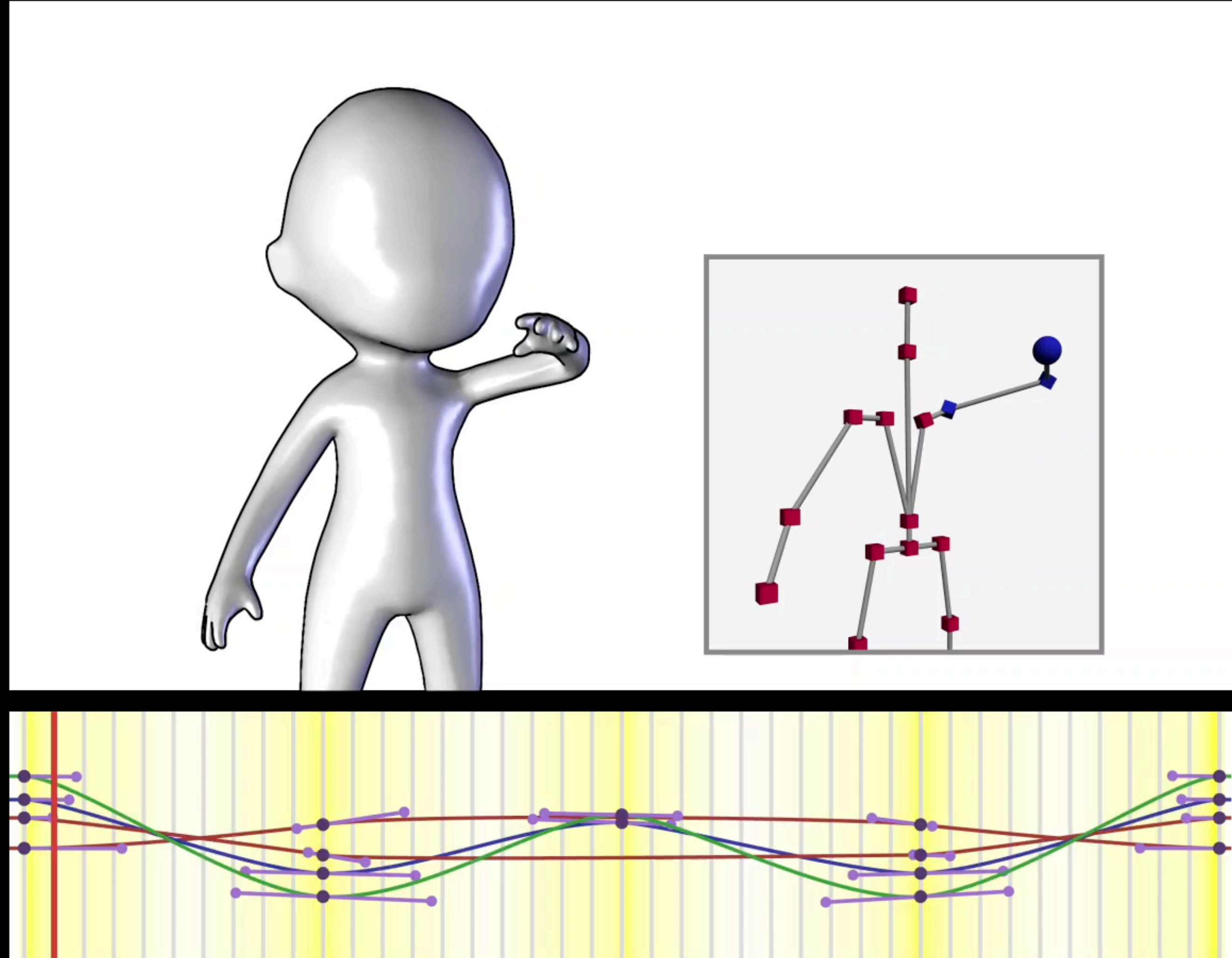
Example: Music Conductor

Initial motion



Example: Music Conductor

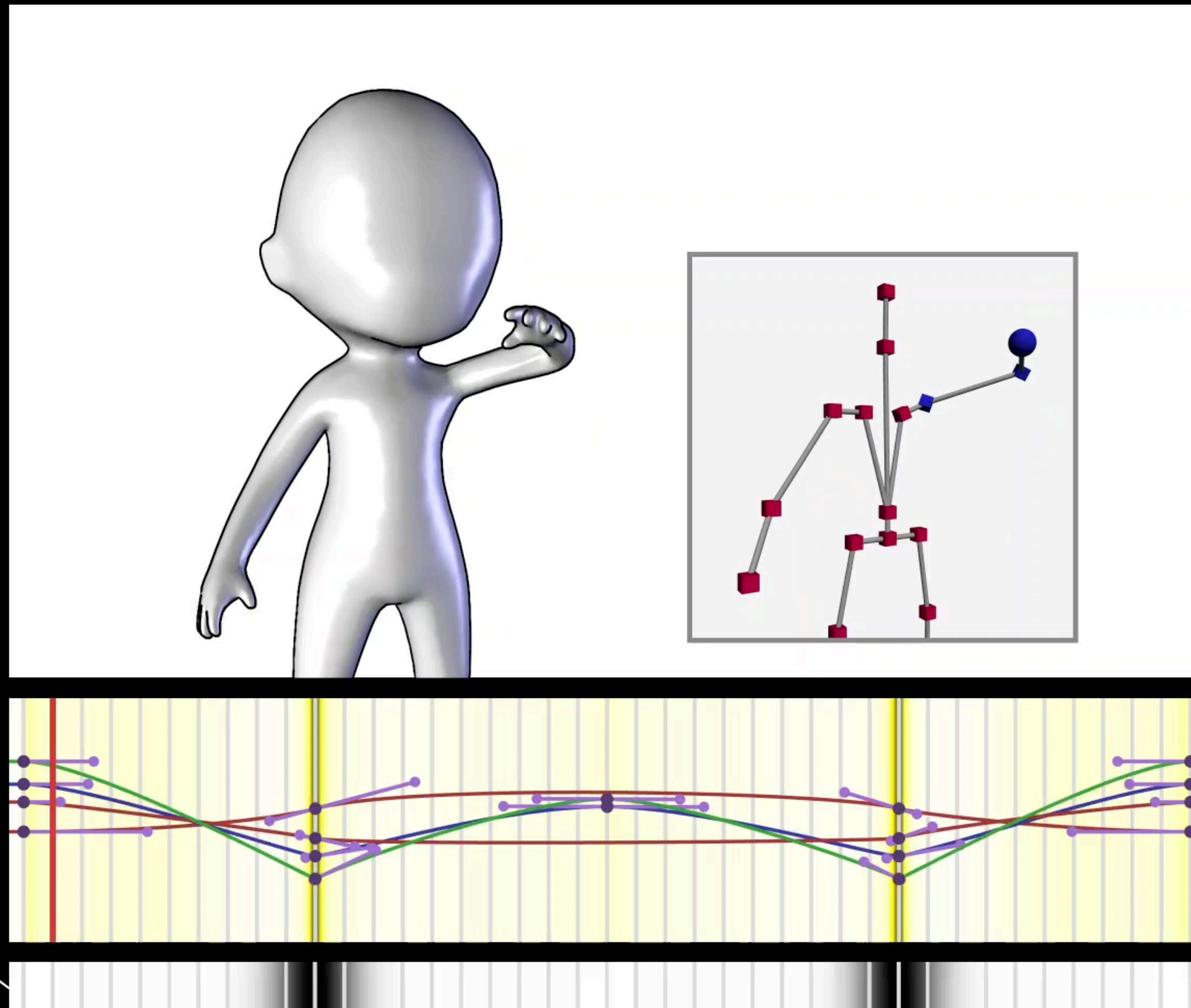
Naïve
optimization



Example: Music Conductor

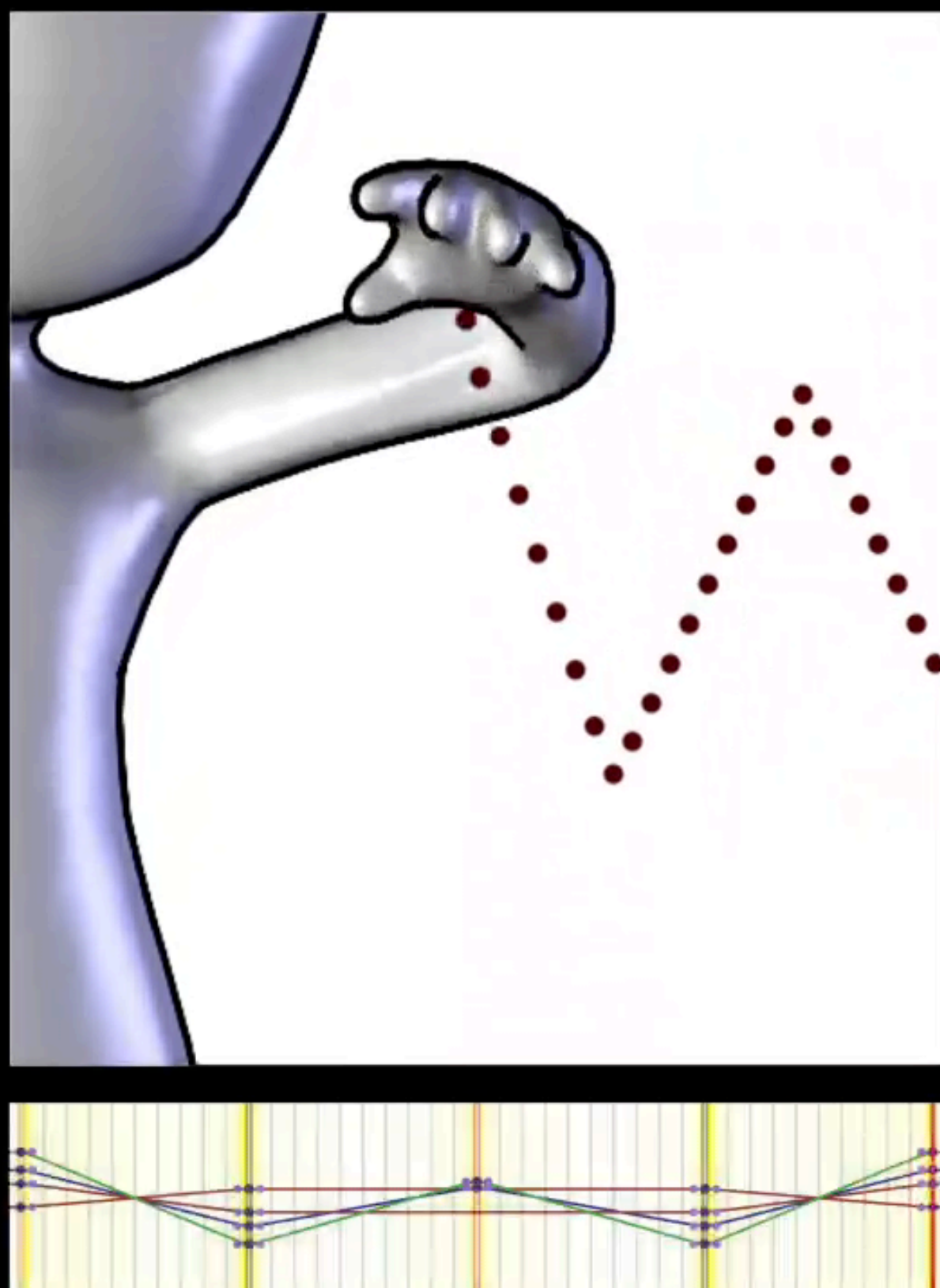
Controlled optimization

- Time-varying cost control

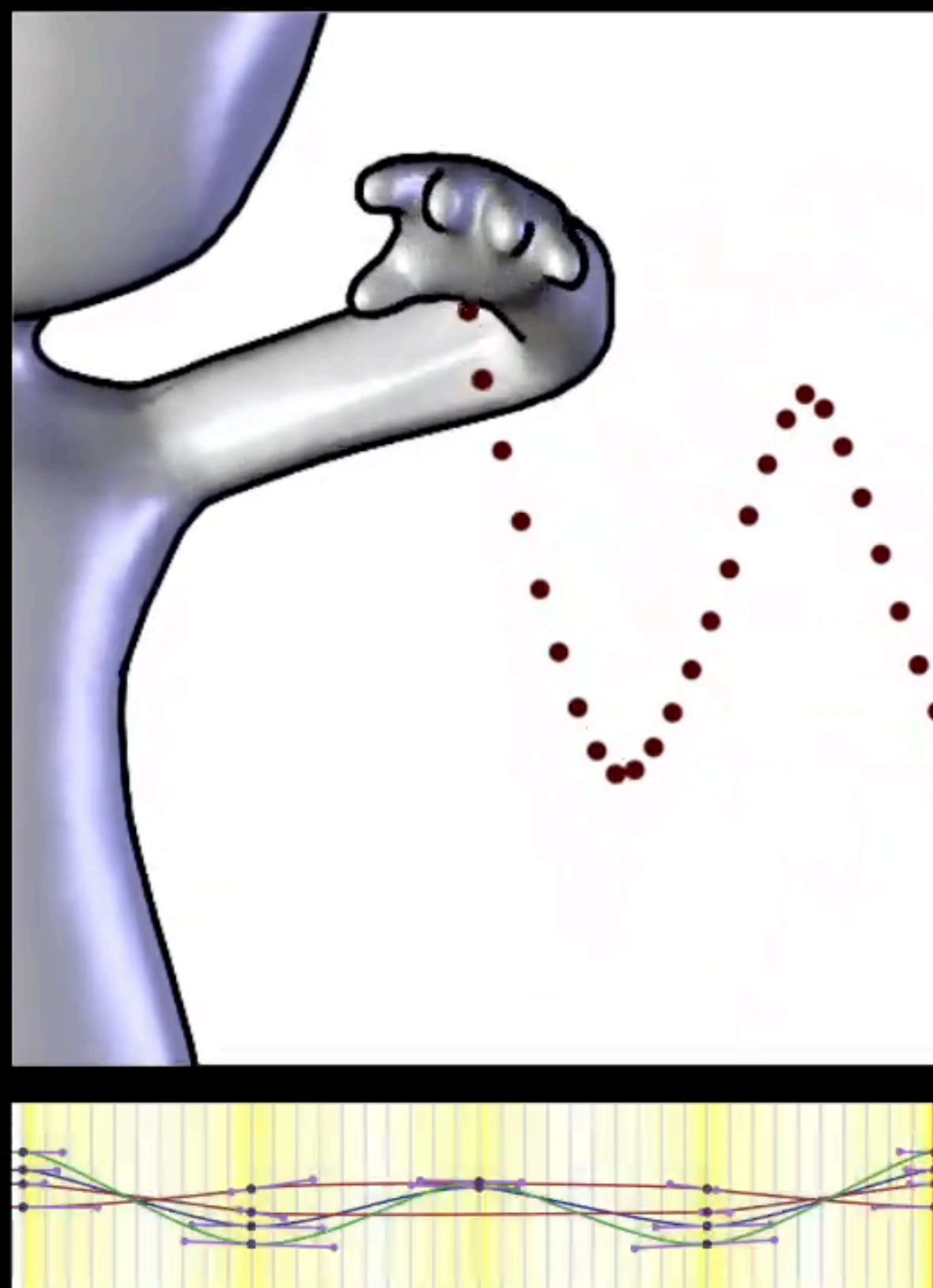


Example: Music Conductor

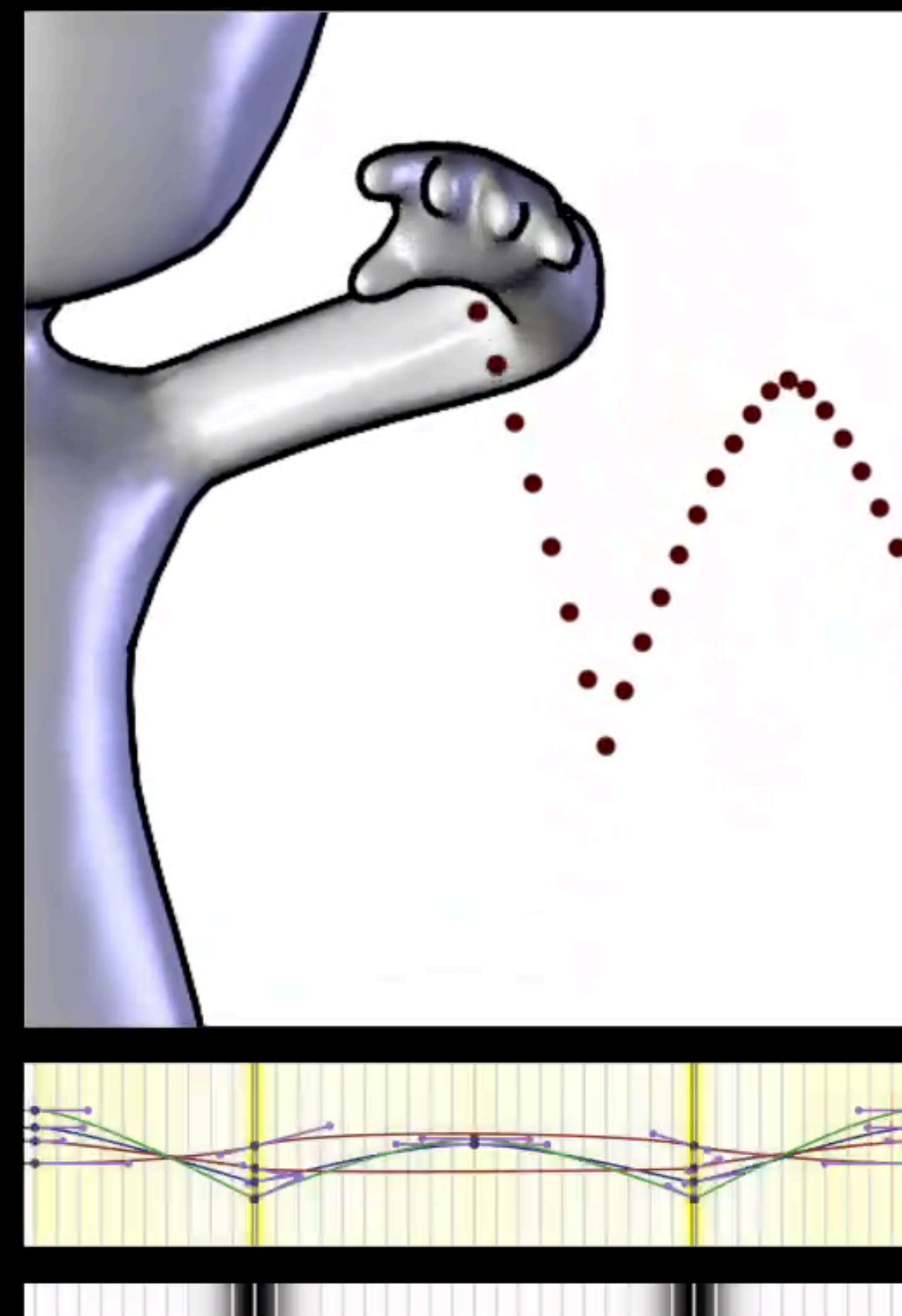
**Initial
motion**



**Naïve
optimization**



**Controlled
optimization**

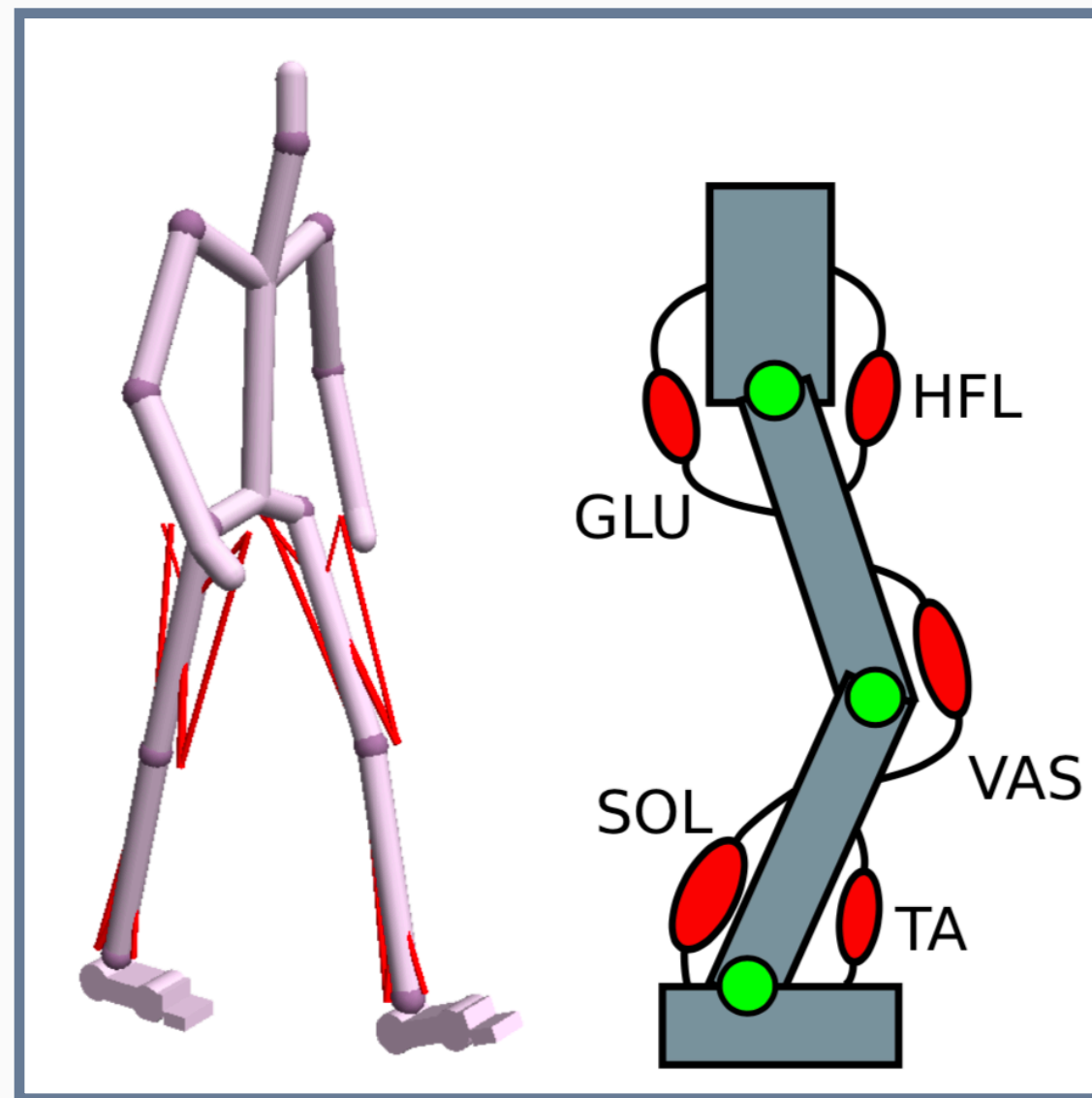


Trajectory Visualization & Playing on 0.5x speed...

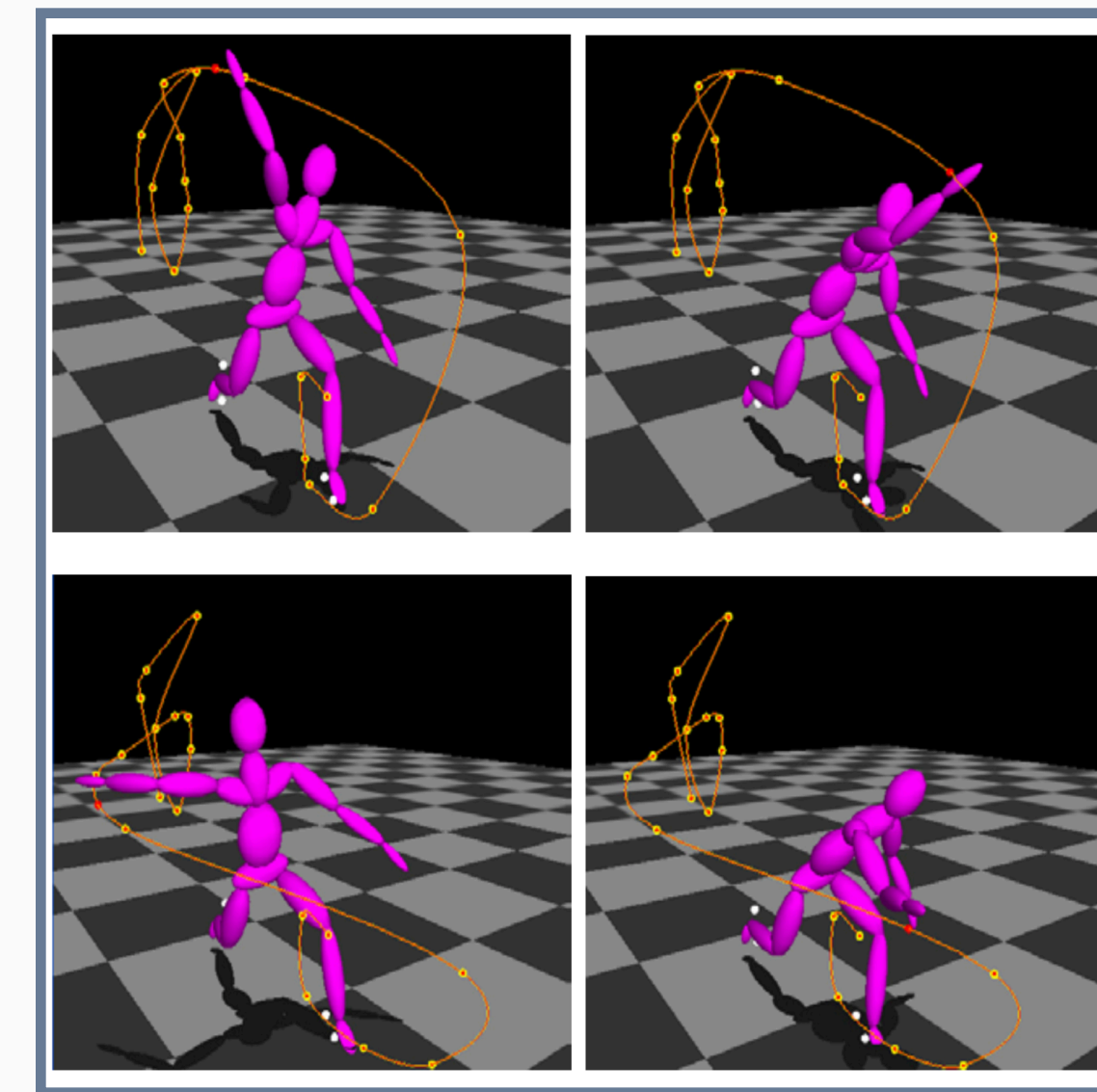
Discussions

Discussion: “One is not Enough”

Providing **more options of cost functions** as well as our physics-based cost function — One is not enough!



Muscle models [Wang+12]



Motion styles [Grochow+04]