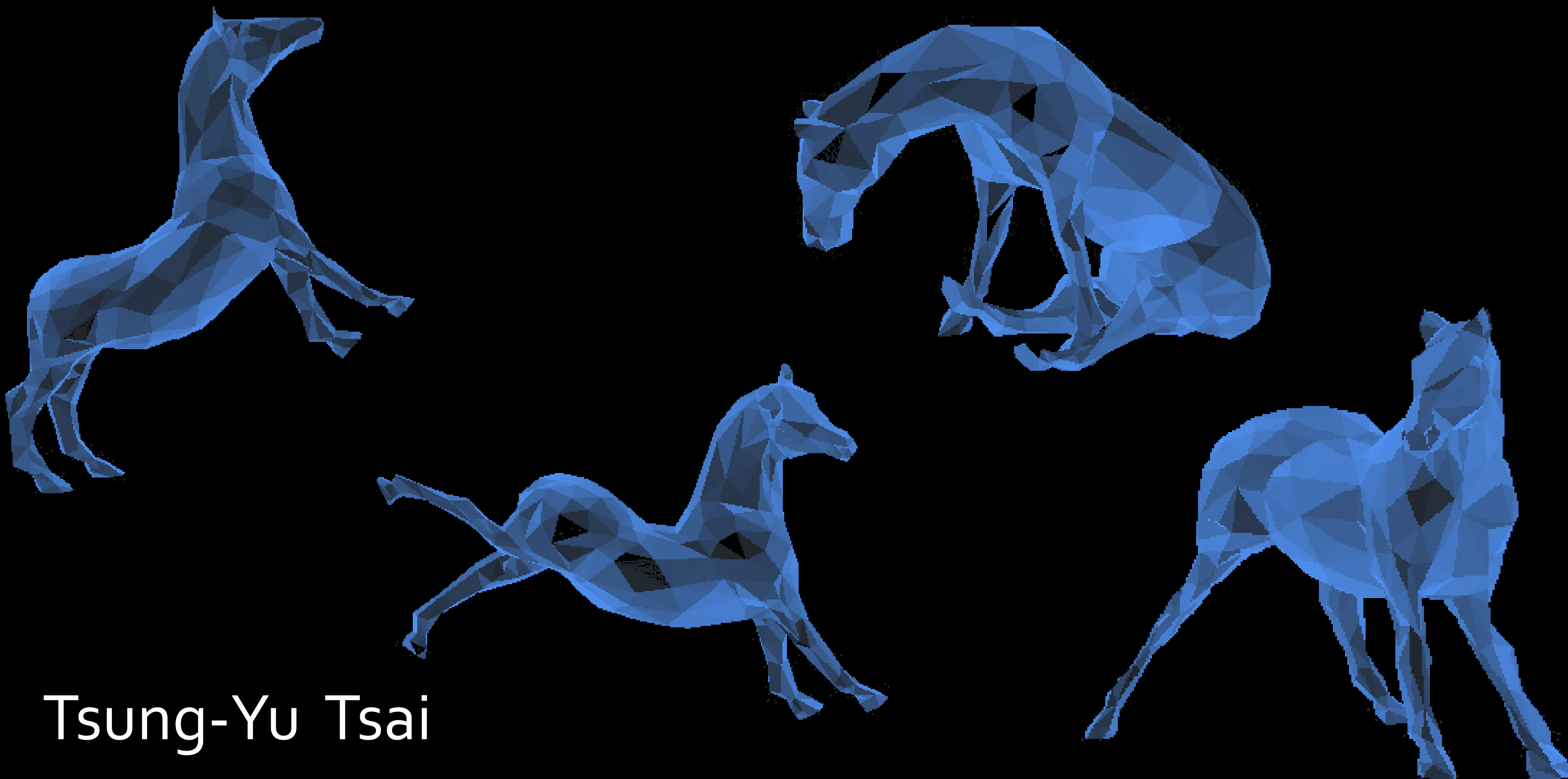


Tsung-Yu Tsai

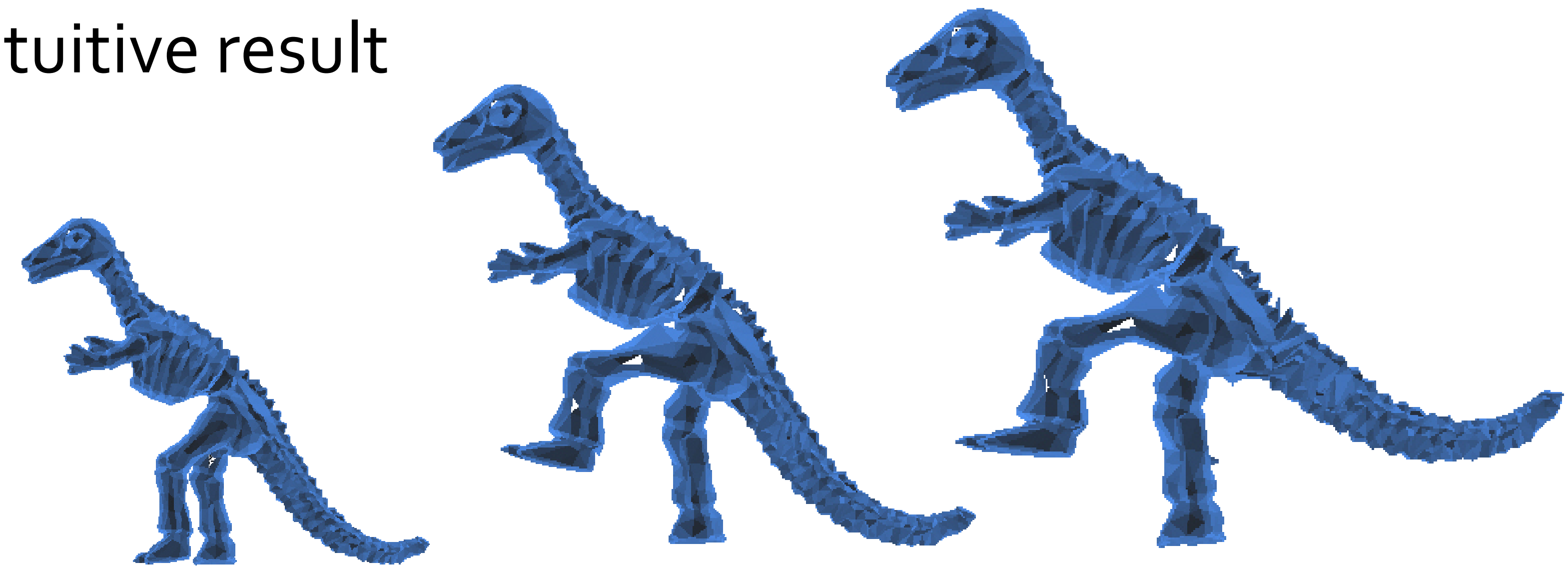
Constraints-based 3D Model Deformation

Advisor : Ming Ouhyoung, Ph.D.

jacktsai@cmlab.csie.ntu.edu.tw

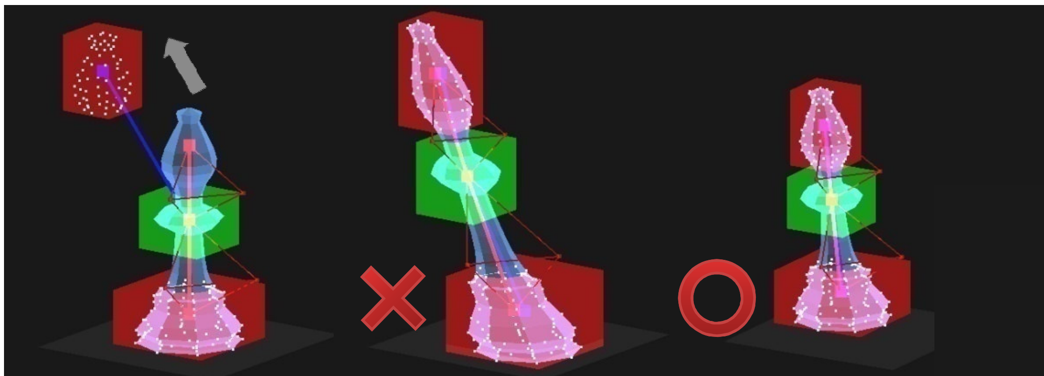
Introduction

- We build a mesh deformation system
  - A user-friendly interface for easy manipulation
  - Detail preservation
  - Satisfying lots constraints
  - Intuitive result

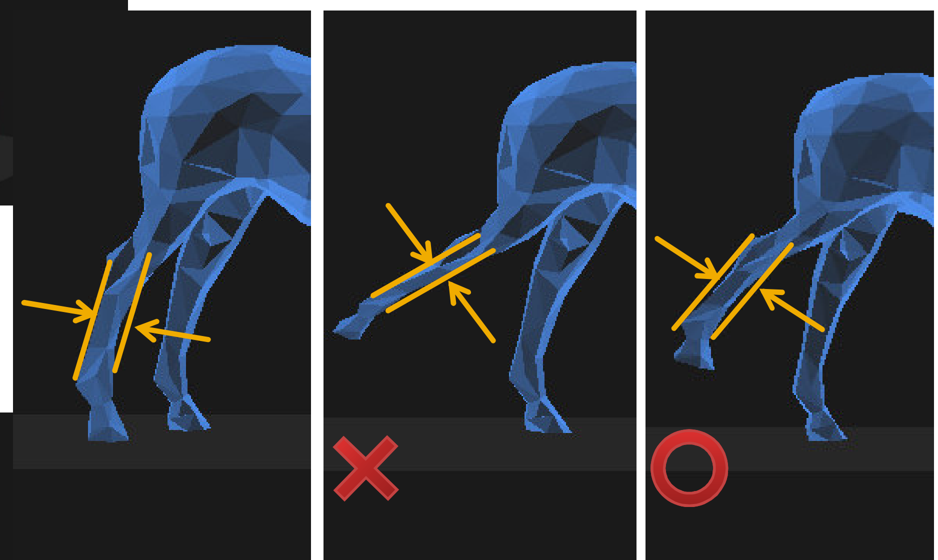


Introduction

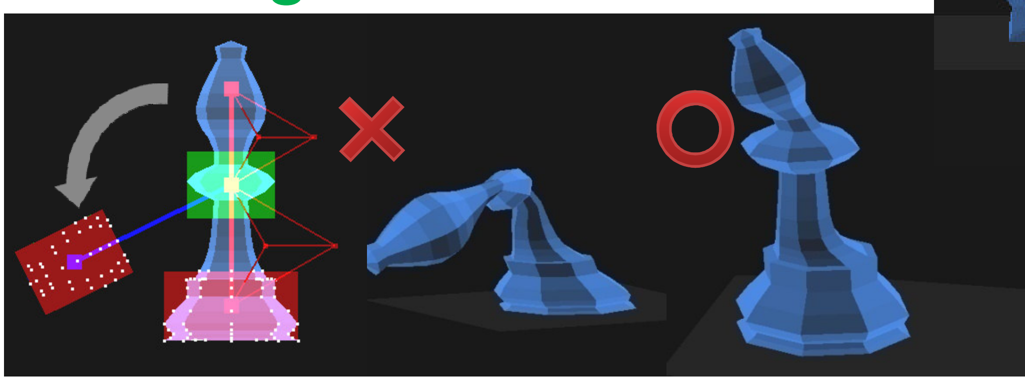
Length constraint



Rigidity constraint



Joint angle constraint



Outline

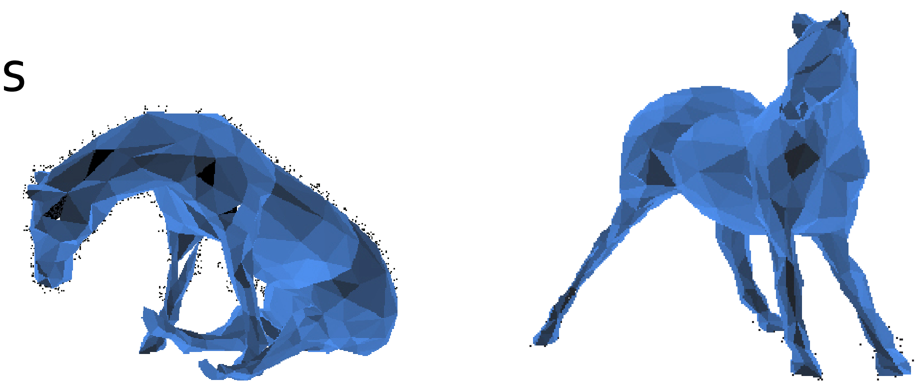
- Introduction
- Related Work
- System
- Results
- Conclusions and Future Work

Outline

- Introduction
- **Related Work**
- System
- Results
- Conclusions and Future Work

Related Work

- Mesh Deformation
  - Motivation : Mesh editing
    - Creating and modifying the shape of model
    - Modify global shape
      - Preserve local features and global continuity
    - Simple control mechanism
    - Intuitive results

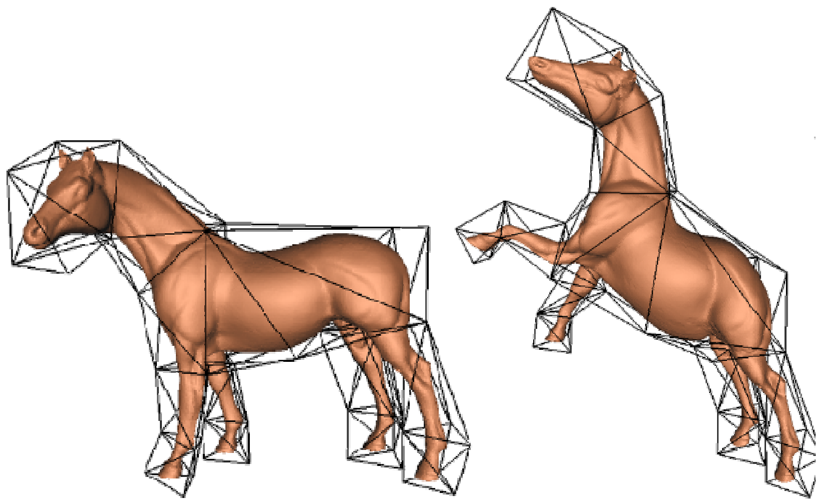


Related Work

- Mesh deformation
  - Detail preservation based on local coordinates
  - **Laplacian surface editing** [*Sorkine et al. 2005*]
    - **Laplacian coordinates**
    - Cast mesh deformation as an energy minimization problem
  - The optimizations involved are often nonlinear and require Gauss-Newton iterations
    - Slow-converging
  - The limitation can be overcome through
    - **Linear solver (faster)** *[Lipman et al. 2004 , Zhou et al. 2005]*

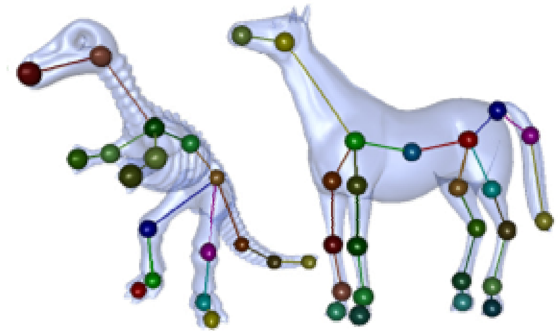
Related Work

- **Mean value coordinates for closed triangular meshes** *[Ju et al. 2005]*
  - A coarser mesh embedding the mesh model
    - Interpolate values assigned to the vertices of a closed mesh
  - The disadvantage
    - Not convenient for user to control
- Other manipulation
  - Control handle
  - **Rigging**



Related Work

- **Mesh Puppetry** [*Zhou et al. 2007*]
  - Direct manipulation & detail preservation
  - A set of high-level IK constraints (length, rigidity, joint limit , balance)
  - A cascading optimization procedure



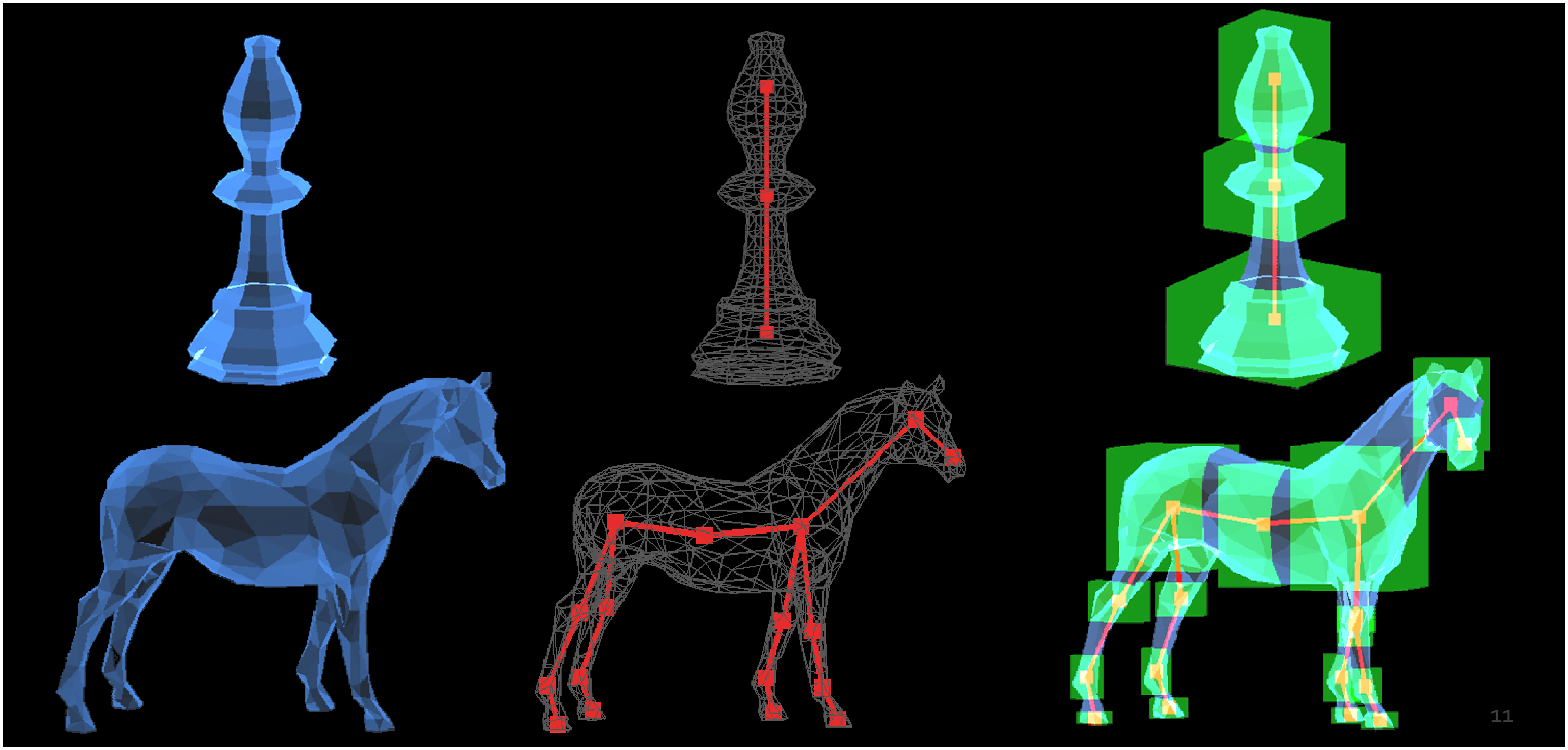
- Our system
  - Easy manipulation & Rigging
  - Satisfying high-level constraints
  - Linear solver for deformation energy function

Outline

- Introduction
- Related Work
- **System**
- Results
- Conclusions and Future Work

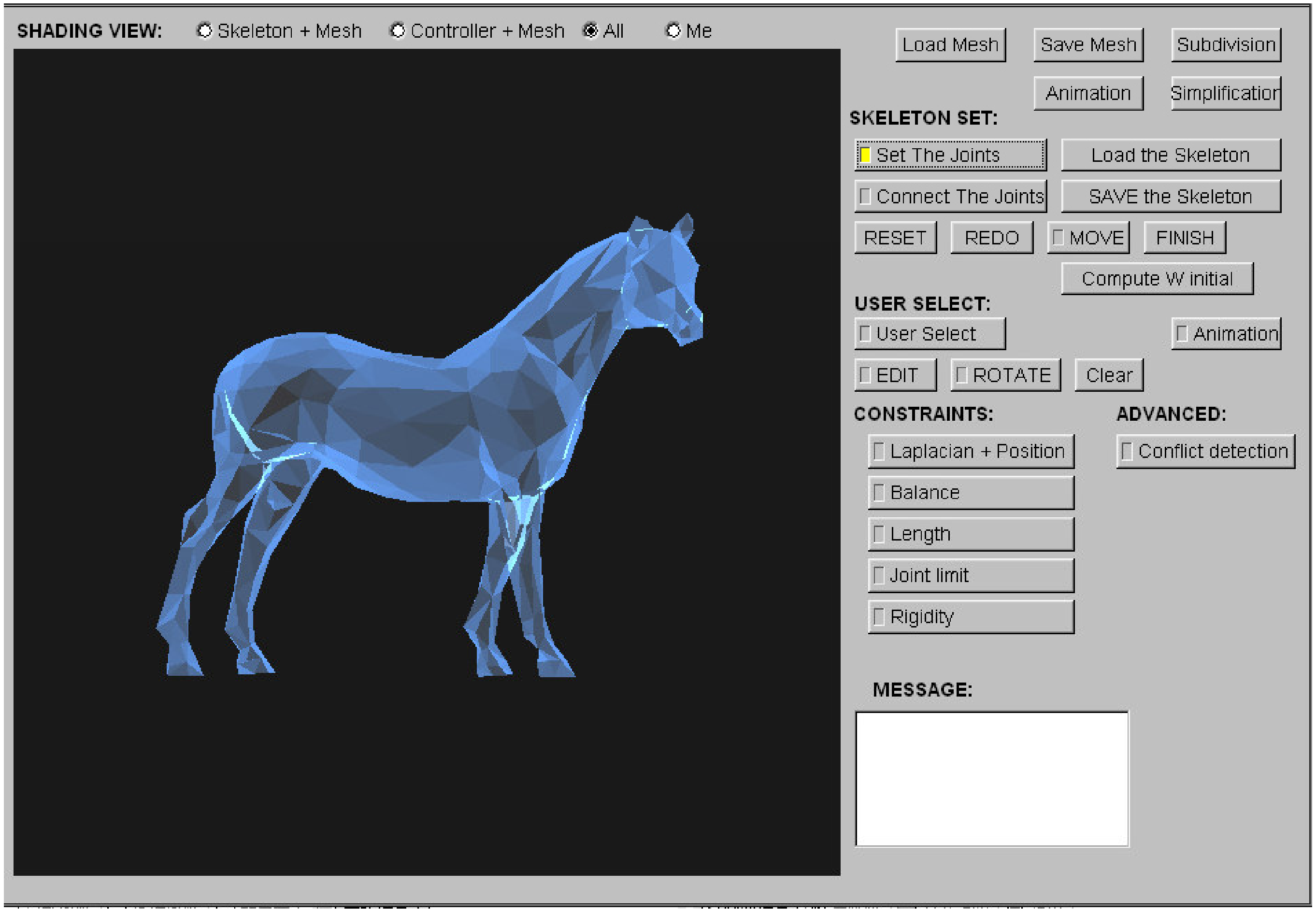
System

- Input: Mesh + skeleton



System

- How to build the skeleton ?



System

- Steps of manipulation

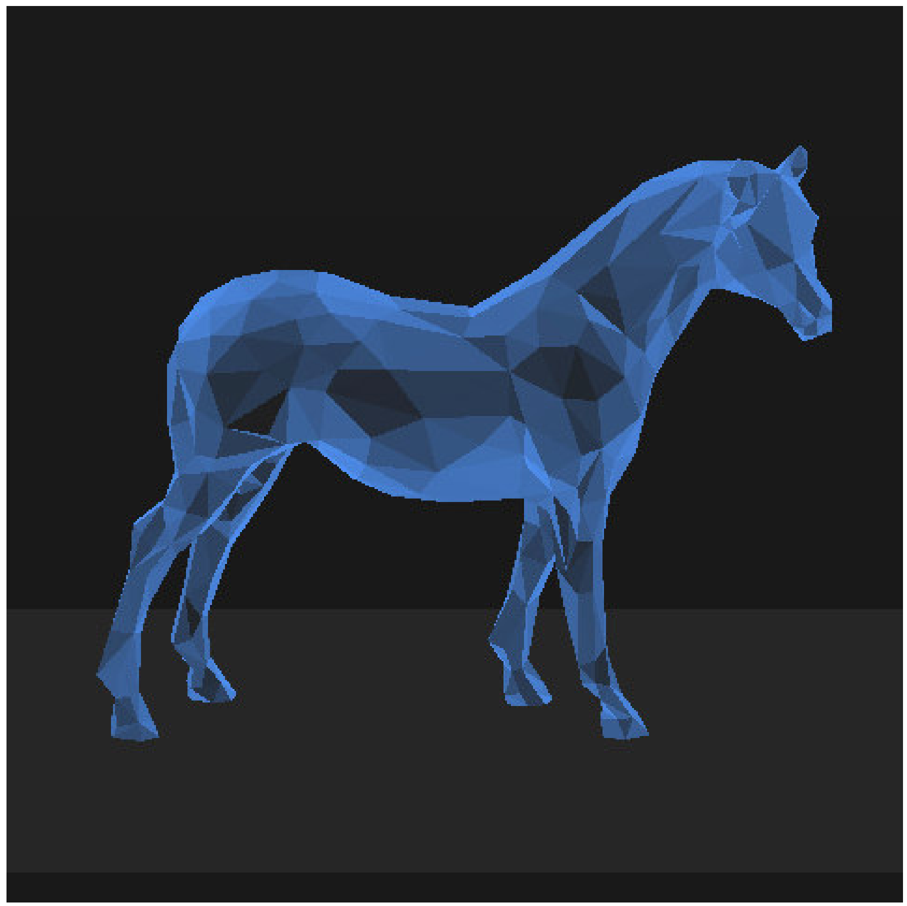
Load Model

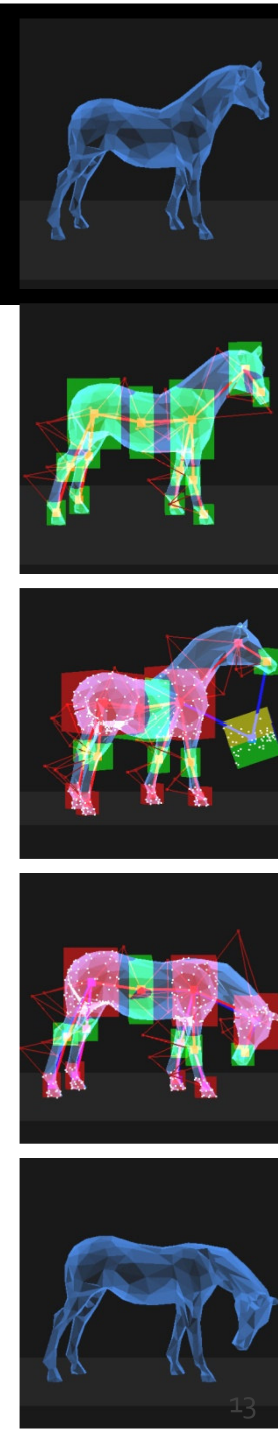
Load skeleton

Move
the selected joint

Deformation

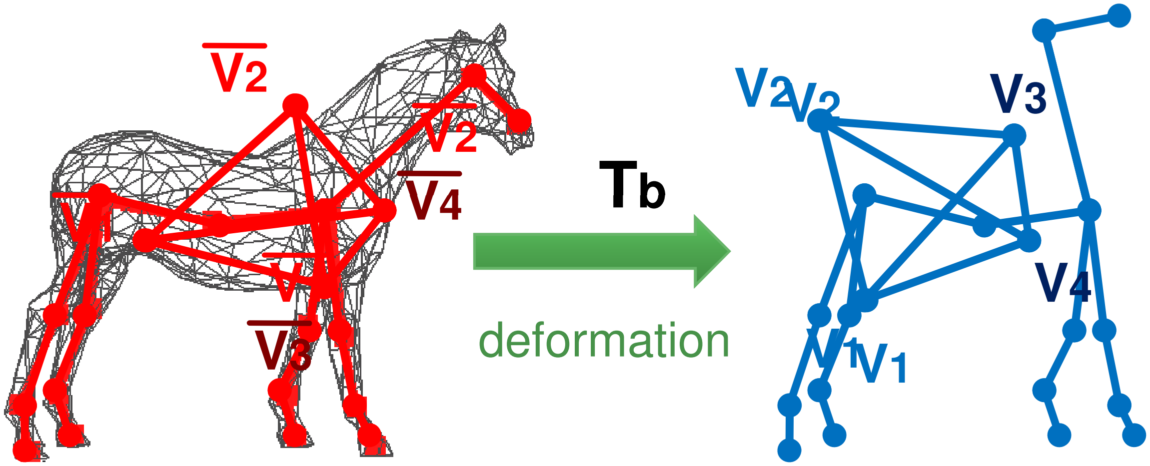
Result



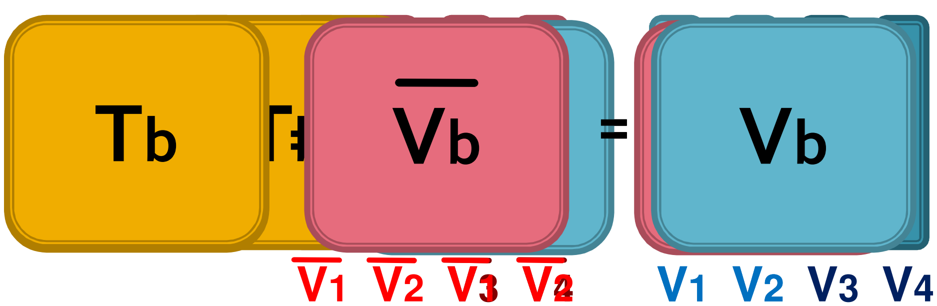


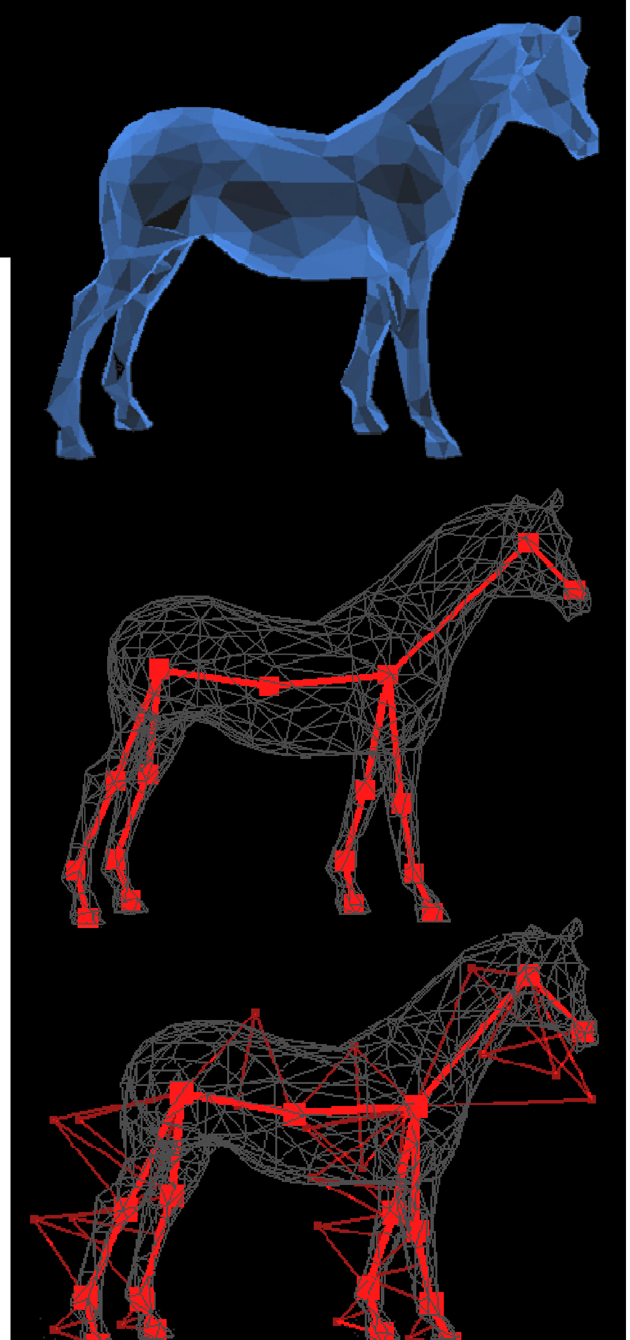
System

- Tetrabone *[Zhou et al. 2007]*



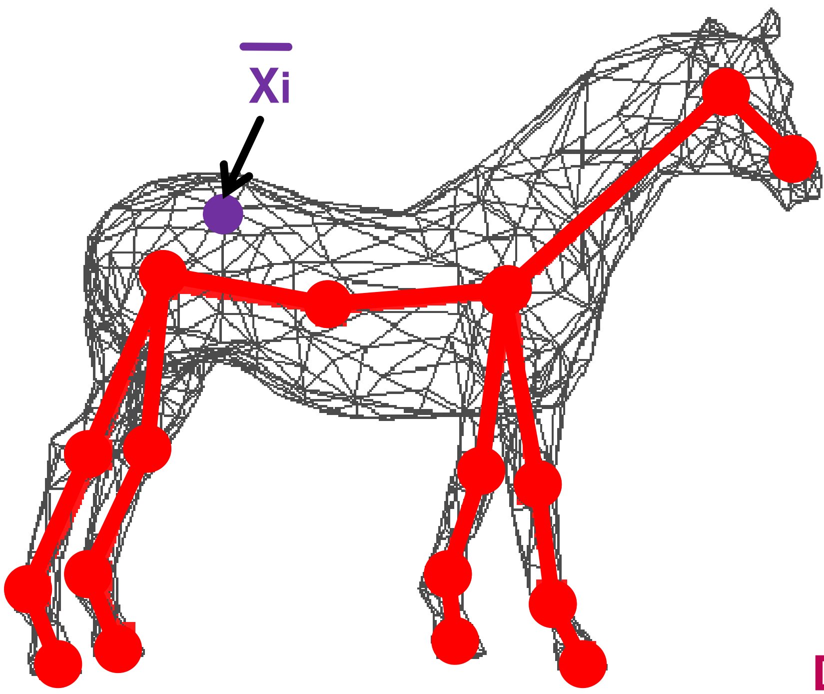
$$\mathbf{V}_b = \mathbf{T}_b \overline{\mathbf{V}}_b \qquad \mathbf{T}_b = \mathbf{V}_b \overline{\mathbf{V}}_b^{\,-1}$$





System

- How to get the deformed mesh ?



$$\mathbf{x}_i = \mathbf{X} = \underset{b\in\text{bones}}{\mathbb{T}\mathbf{W}\overline{\mathbf{X}}'_b}\,)\;\bar{\mathbf{x}}_i$$

$$\boxed{\mathbf{T}_b = \mathbf{V}_b\overline{\mathbf{V}}_b^{\;-1}}$$

$$\mathbf{x}\,\boxed{\mathbf{X}} = \underset{\backslash b\in\text{bones}}{\mathbb{V}\,\overline{\mathbb{V}}^{-1}}\;\mathbf{W}\overline{\mathbf{X}}^{1}\Big)\;\bar{\mathbf{x}}_i$$

Deformed mesh model

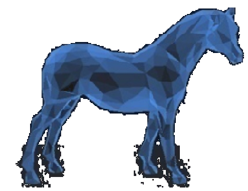
System

- The output of skinned mesh

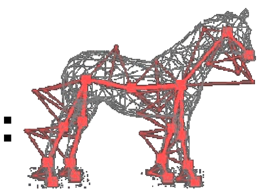
$$\mathbf{X} = \boxed{\mathbb{V}}\,\overline{\mathbb{V}}^{-1}\,\boxed{\mathbf{W}}\,\overline{\mathbf{X}}$$

We want to get them!!

$\overline{\mathbf{X}}$:



$\overline{\mathbb{V}}$:

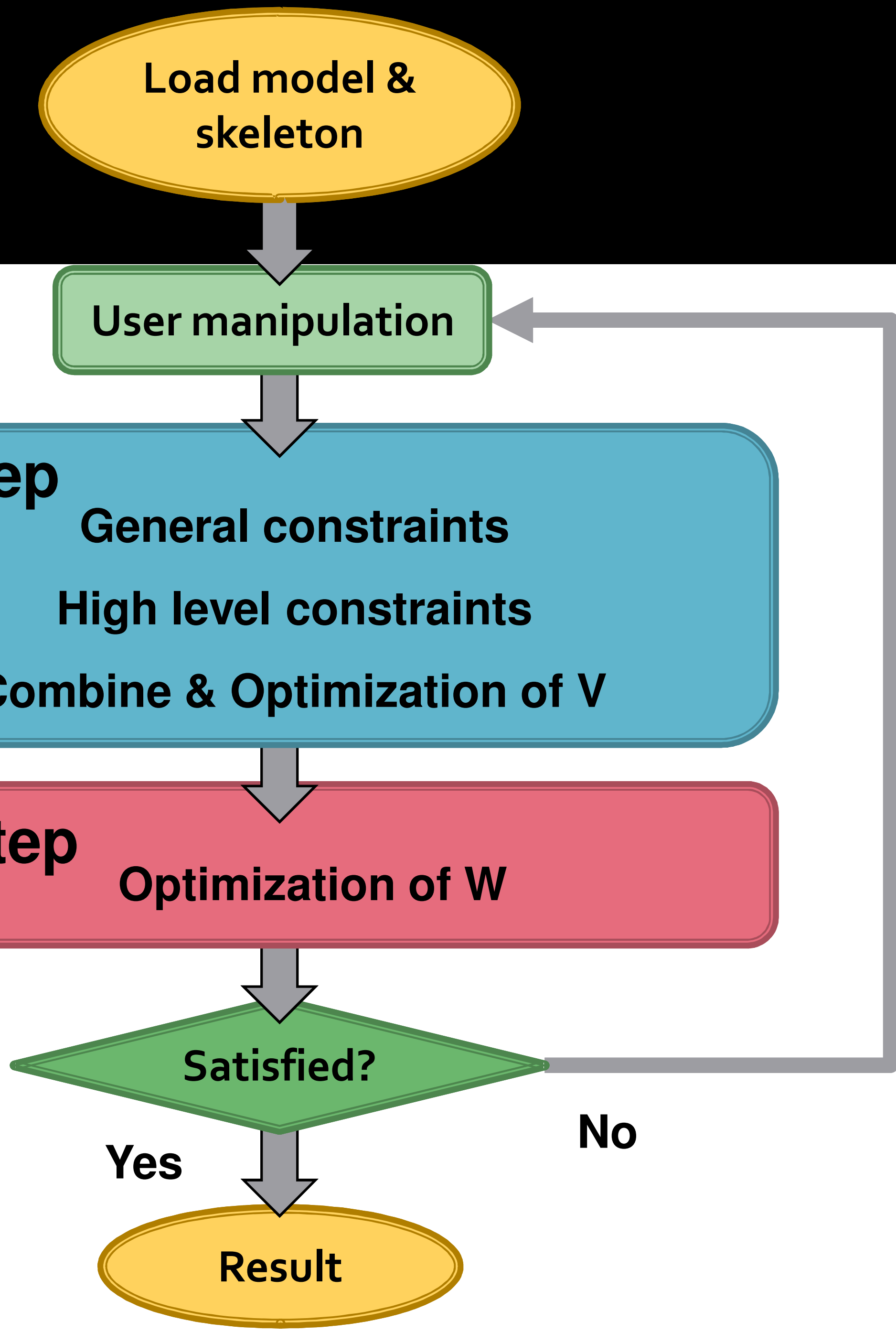


- We look up for a deformed mesh with vertex position **X** (as a function of **V** and **W**)
  - Minimize the global deformation energy

$$\mathcal{M} = \underset{\mathbf{X}=\mathbb{V}\overline{\mathbb{V}}^{-1}\mathbf{W}\overline{\mathbf{X}}}{\arg\min}\ \mathcal{E}(\mathbf{X})$$

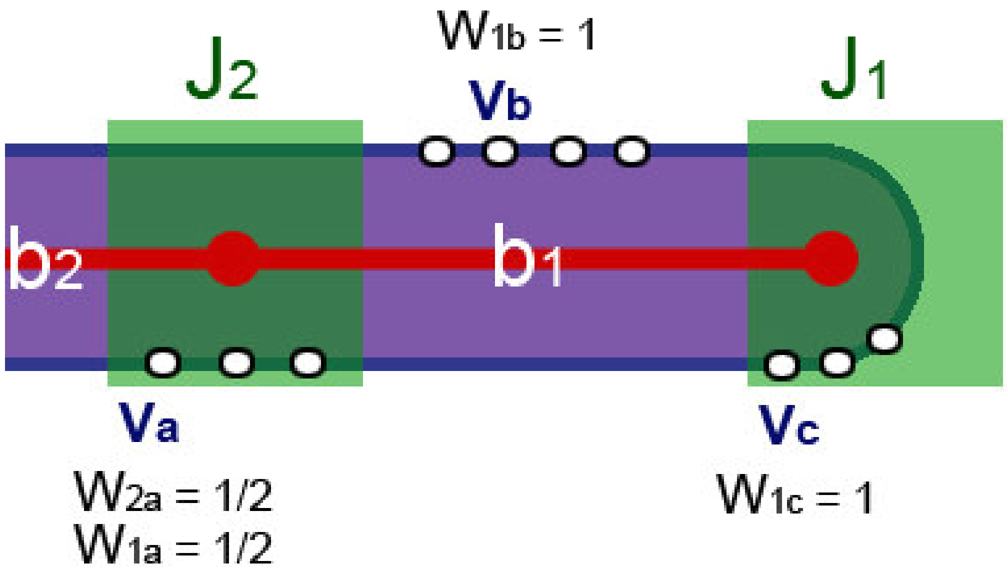
System

flowchart



System

- Initialize W



System

- V-step
  - General constraints
    - Laplacian constraint
    - Position constraint
  - High level constraints
    - Length
    - Rigidity
    - Joint angle limit

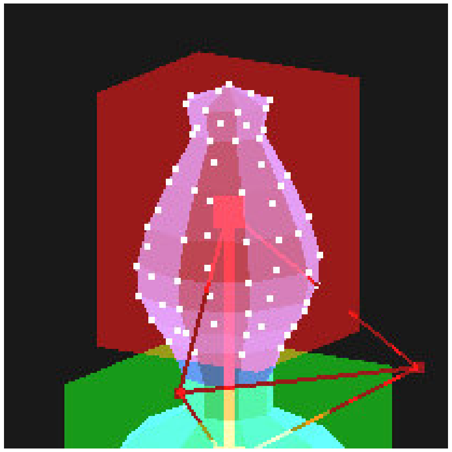
System V-step

- **Laplacian constraint**
  - Preserve the detail of the surface

$$\left\| LX - \frac{LX'}{\|LX'\|} \|L\bar{X}\| \right\|^2$$

- **Position constraint**
  - Allow direct manipulation of the mesh for intuitive design

$$\| PX - X' \|^2$$



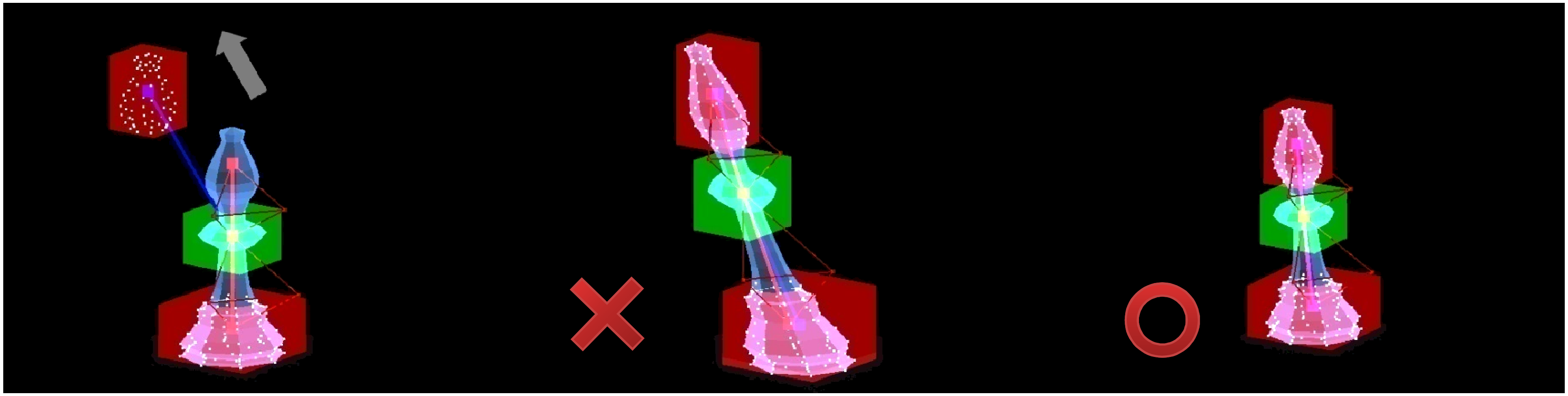
System

V-step

- **Length constraint** *[Zhou et al. 2007]*
  - Control the length of the “bones”

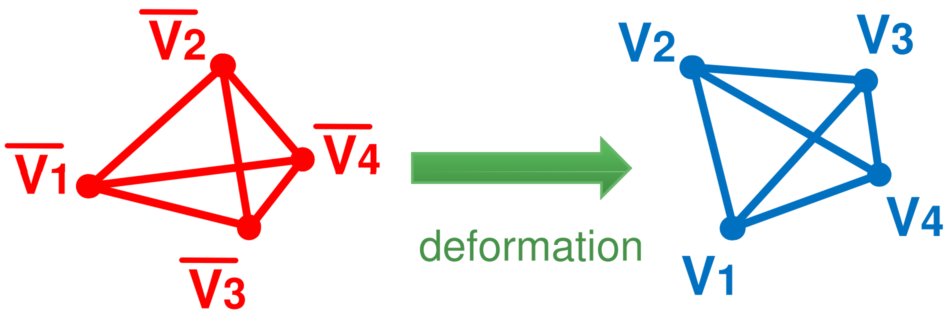
$$\sum_{(i,j)\in \text{bones}} \left(\|\mathbf{v}_i - \mathbf{v}_j\| - \boxed{L_{ij}}\right)^2$$

Original length of bone

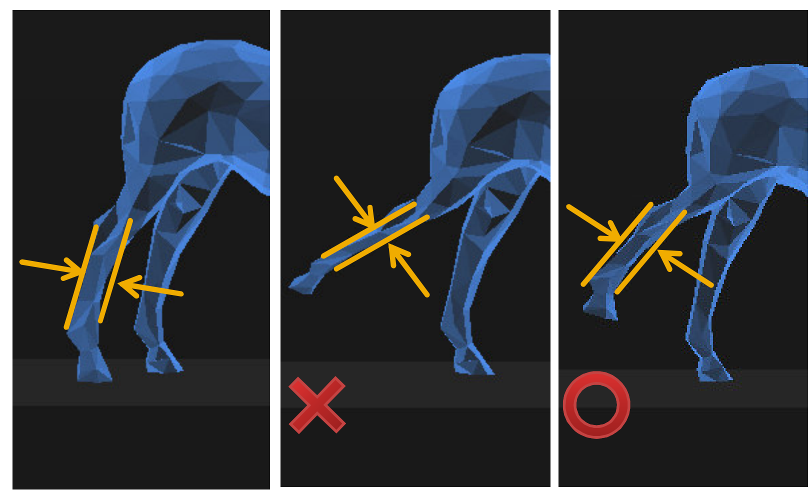


System V-step

- **Rigidity constraint** *[Zhou et al. 2007]*
  - Force near-rigid deformation of skin around bones



$$\sum_{(i,j)\in tetra(b)} \left(\left\| \mathbf{v}_i - \mathbf{v}_j \right\| - l_{ij} \right)^2$$

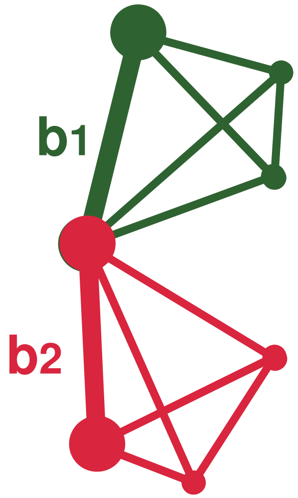


- v_i, v_j : the position of tetravertices i , j
- l_{ij} : the distance between tetravertices i , j

System

V-step

- **Joint angle limit constraint** *[Zhou et al. 2007]*
  - Restrict the range of joint angles for added realism

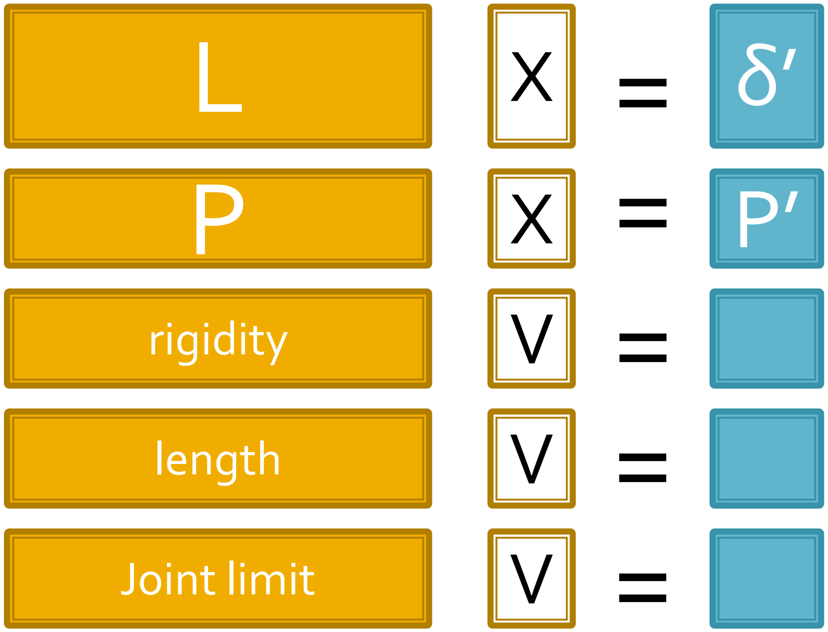


$$\sum_{(i,j)\in \text{pairs}(b_1,b_2)} \left\| (\mathbf{v}_i - \mathbf{v}_j) - \theta_{ij} \right\|^2$$

- v_i, v_j : the position of tetravertices i , j
- Θ_{ij} : the target vector between tetravertices i , j

System V-step

- Combine all constraints



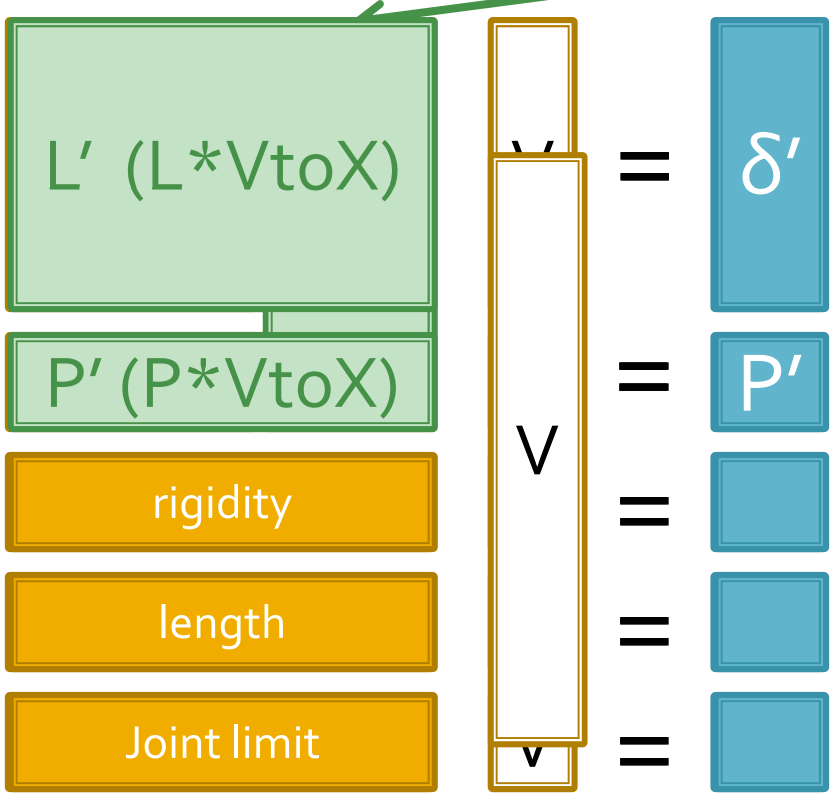
- V-step : Optimization of V
  - Method 1 : Only solve V then get X
  - Method 2 : Solve V and X at the same time

System V-step

Method 1 :
Only solve V

$$X = V\bar{V}^{-1}W\bar{X}$$

Initial W



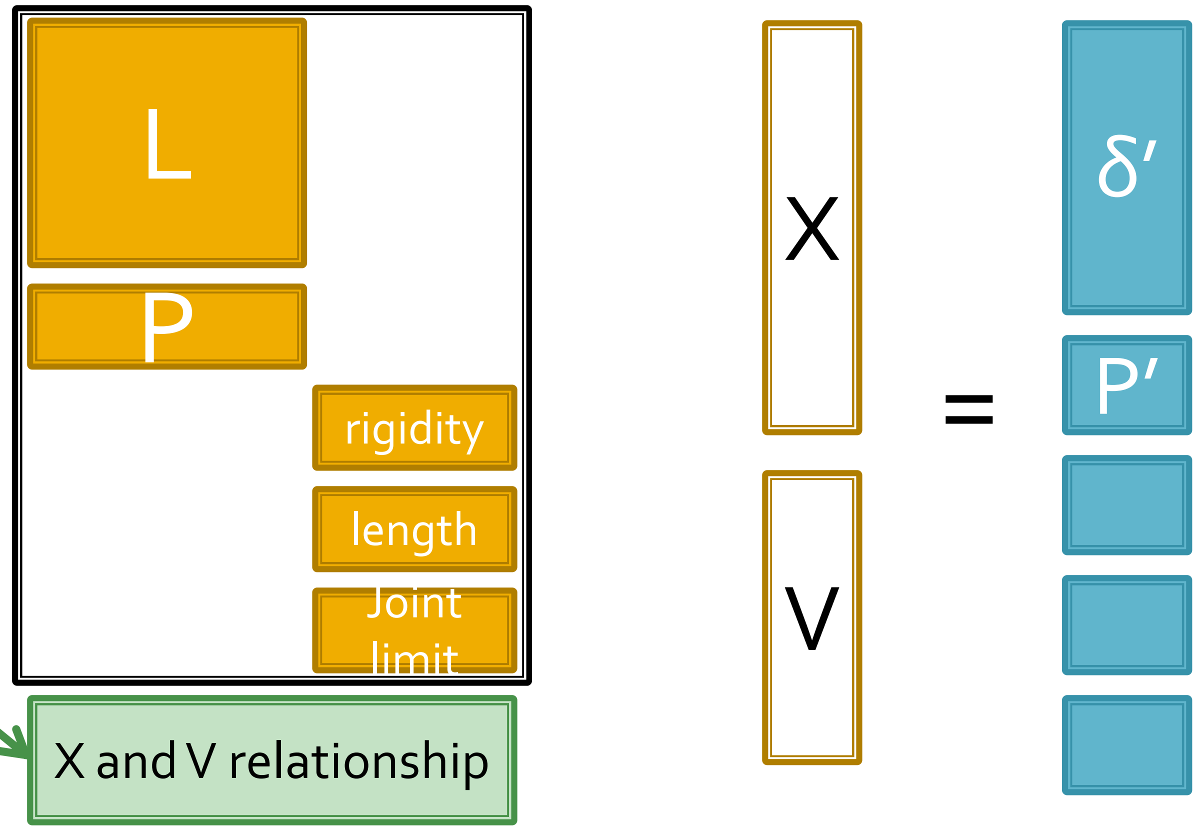
→get V → $X = V\bar{V}^{-1}W\bar{X}$ → get X !!

System V-step

Method 2 :

Solve V & X

$$X = V\boxed{\bar{V}^{-1} W \bar{X}}$$



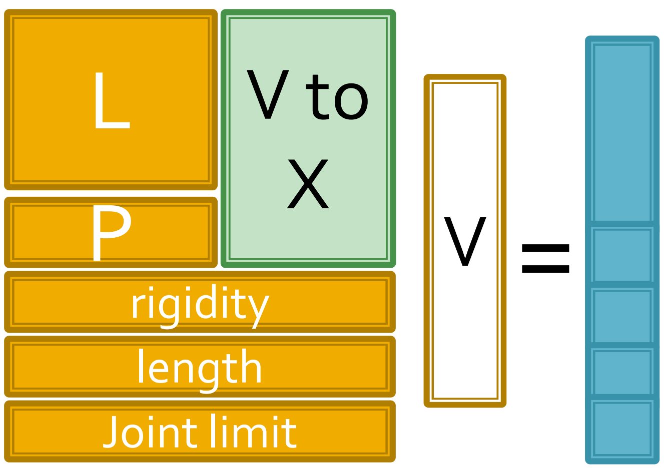
→ get (X & V)

→ get X !!

System V-step

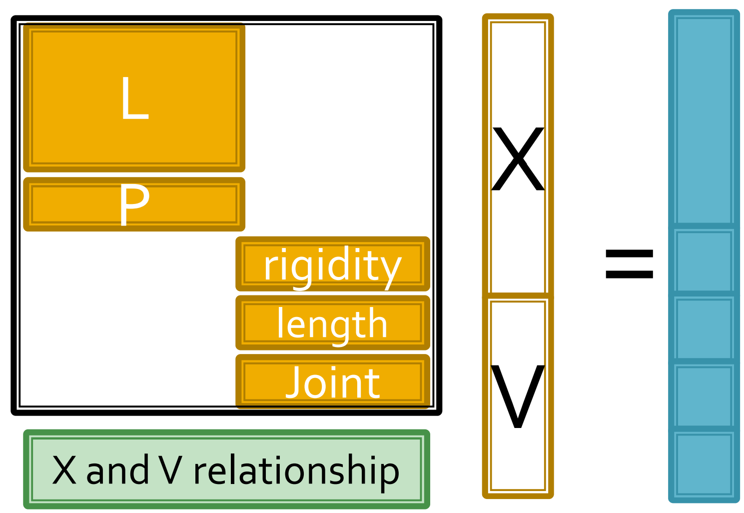
Method 1

2*nB +nJ



Method 2

X.size + 2*nB +nJ



$$V = (A^T A)^{-1} A^T b$$

(2*nB +nJ) * (2*nB +nJ)

small

(X.size+2*nB +nJ) * (X.size+2*nB +nJ)

large

System V-step

Compare

	Method 1	Method 2
Dimension	**Small**	**Large**
Result	**Worse**	**Better**
Cost time	**0.6 sec**	**1.2 sec (more)**
W-step	**necessary**	**optional**
Large model	**efficient**	**slower**

System W-step

- **W-step**
  - Constraints on Vertex Weights only
    - Smooth constraint

$$\varepsilon = \| LX - LX' \|^2 + \| PX - X' \|^2$$ (Laplacain & Position constraint)

$$+ \sum_{(i,j) \in pairs(b_i, b_j)} \left(W_{bi} - \frac{1}{|N(i)|} \sum_{j \in N_i} W_{bj}\right)^2$$ (Smooth term)

$$+ \sum_{i \in [1 \dots n]} \left(\sum_{b \in B} W_{bj} - 1\right)^2$$ (Normalization term)

System

W-step

W-step times

0

0

1

2

3

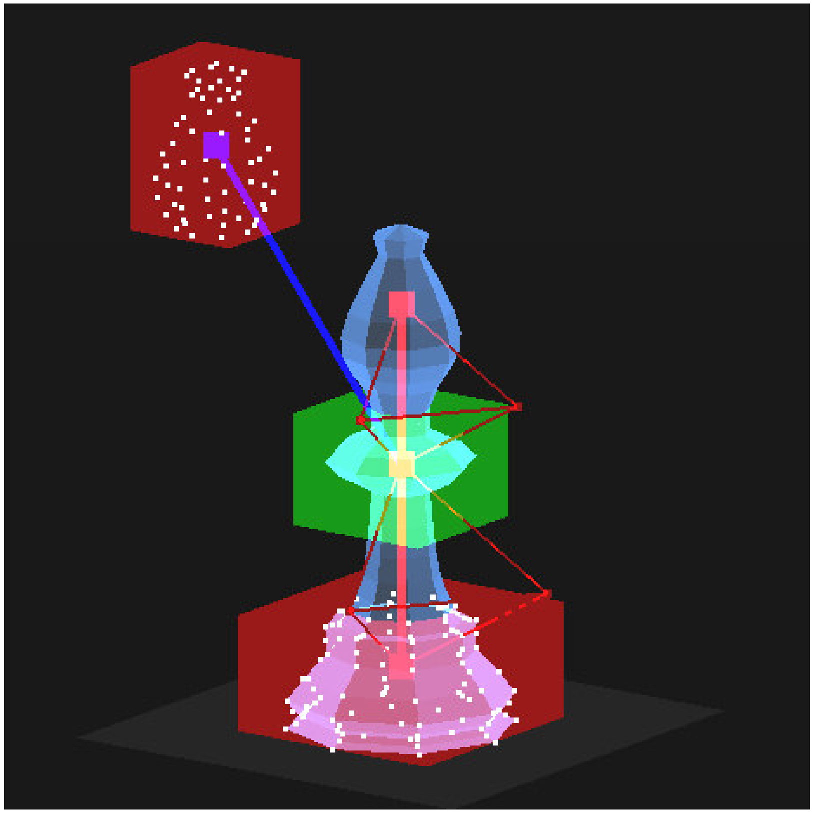
Outline

- Introduction
- Related Work
- System
- **Results**
- Conclusions and Future Work

Results

Length constraint

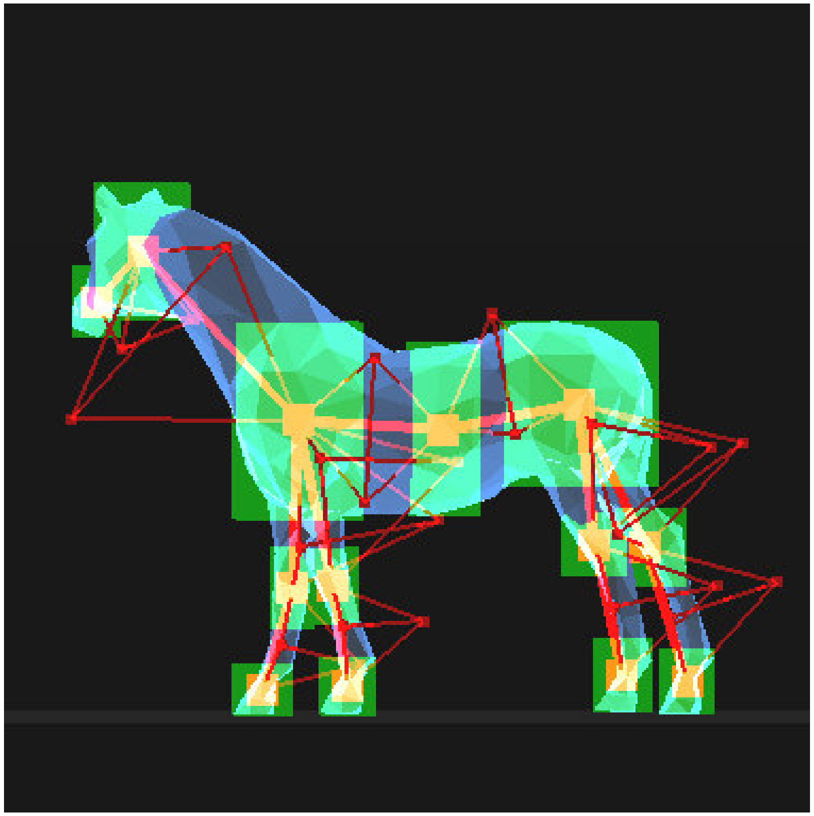
Without Length constraint



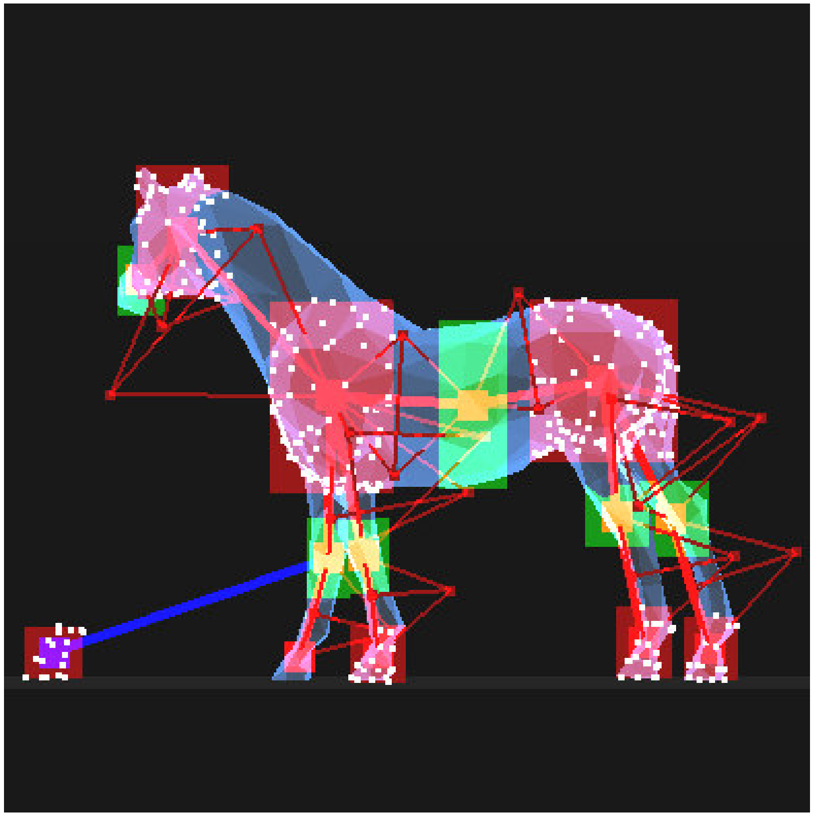
With Length constraint

Results

Length constraint



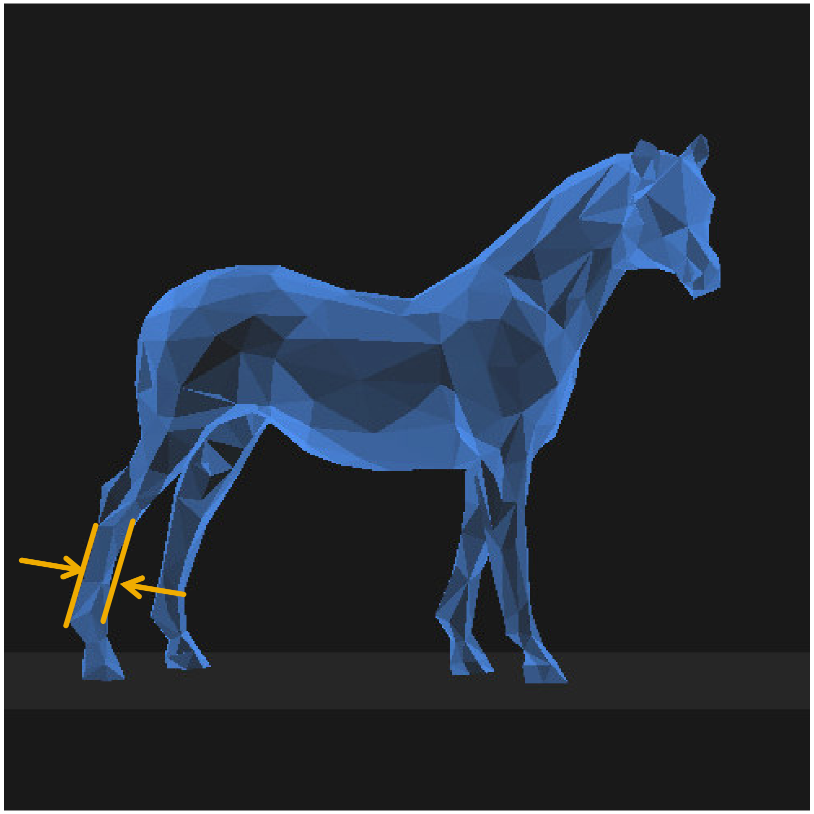
Without Length constraint



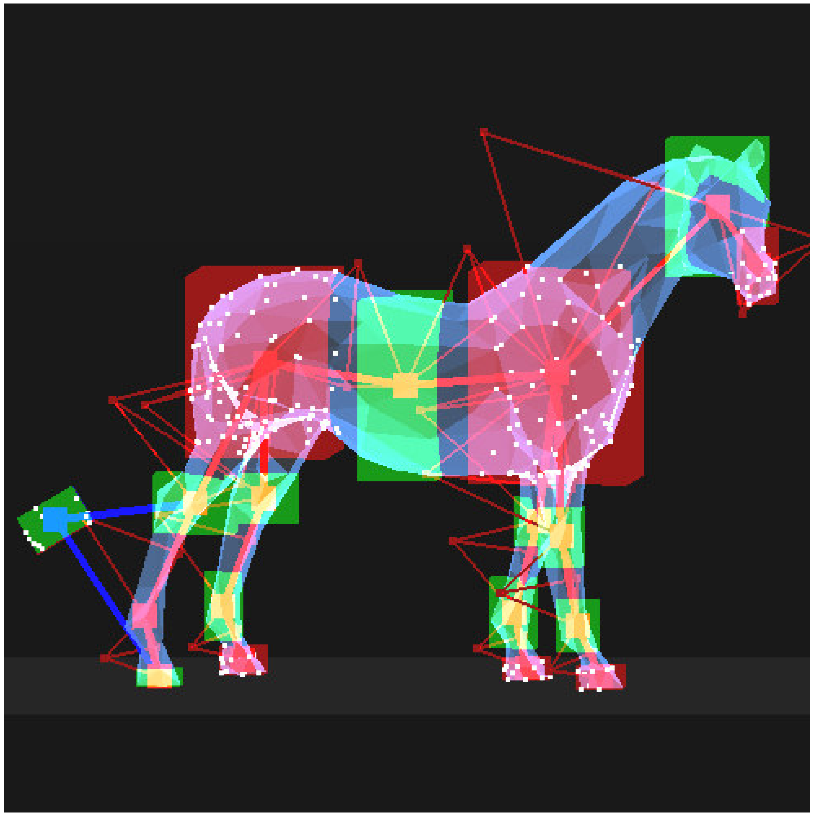
With Length constraint

Results

Rigidity constraint



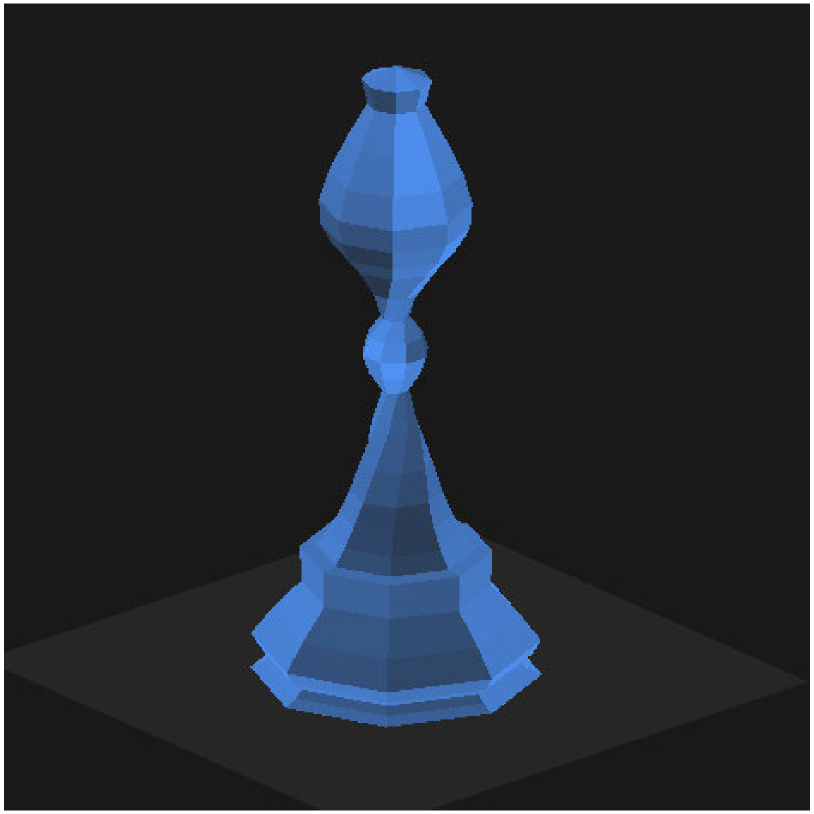
Without Rigidity constraint



With Rigidity constraint

Results

Joint angle constraint

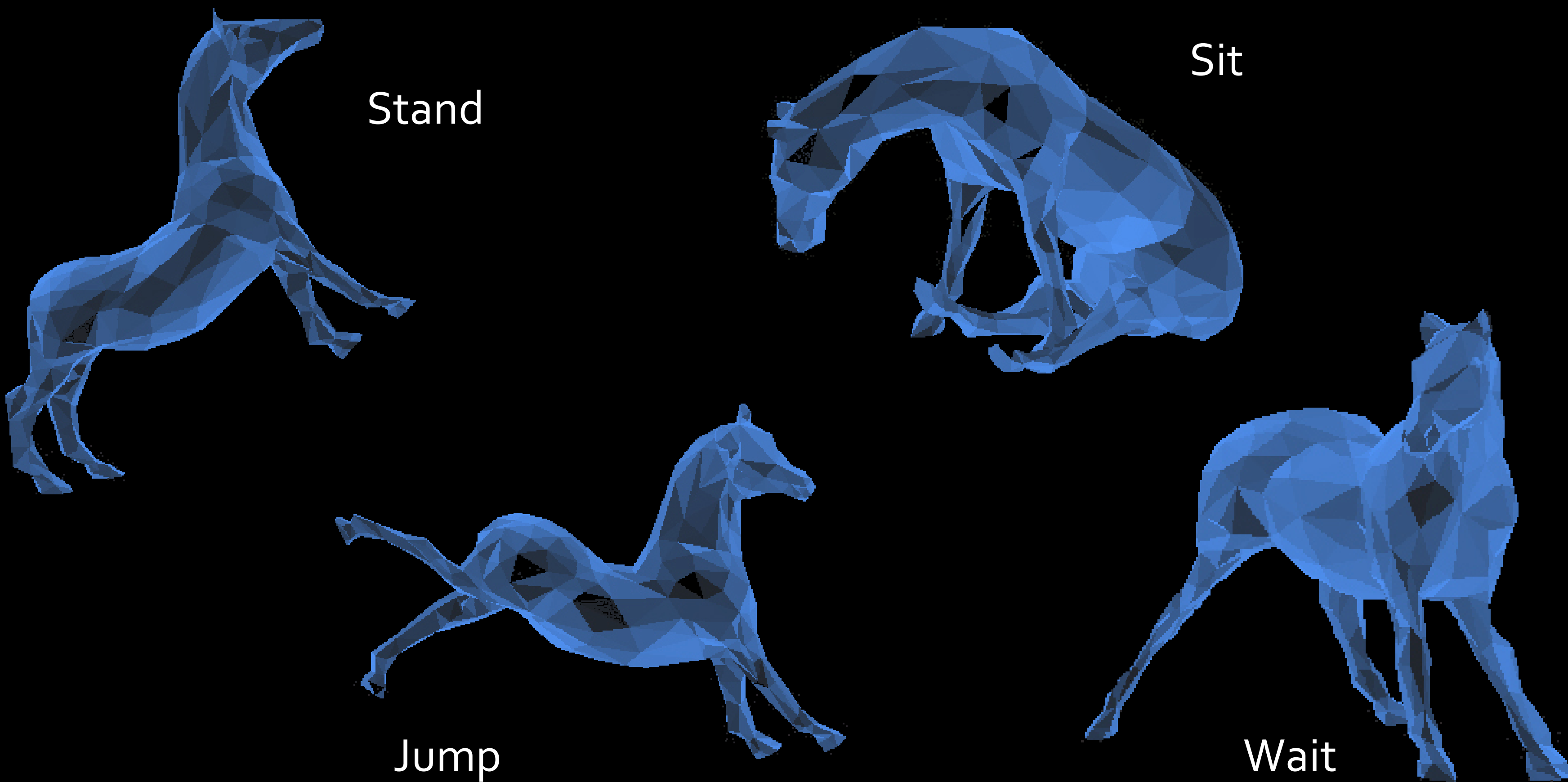


Without Joint angle constraint

With Joint angle constraint

Results

Some Interesting Results



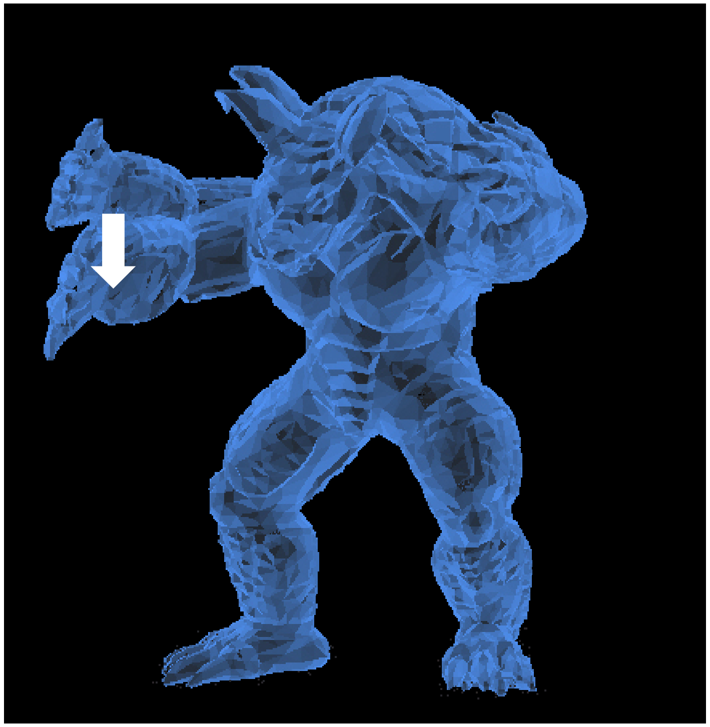
Results

Some Interesting Results

Dinosaur

Raise left leg up

Raise tail up



Armadillo

Outline

- Introduction
- Related Work
- System
- Results
- **Conclusions and Future Work**

Conclusions and Future Work

- Conclusions
  - Convenience of manipulation on rigging and deformation
  - High-level constraints, and more natural and realistic deformed mesh
  - Potential of the system
    - An interactive deformation platform
    - Various applications
      - Deformation transfer
      - Motion retargeting

Conclusions and Future Work

- Future Work
  - Balance constraint
    - Mesh Puppetry [*Zhou et al. 2007*]
  - Auto-skeleton extraction
    - Domain Connected Graph: the Skeleton of a Closed 3D Shape for Animation *[Wu et al. 2006]*
  - Implement on multi-core processor

Demo film

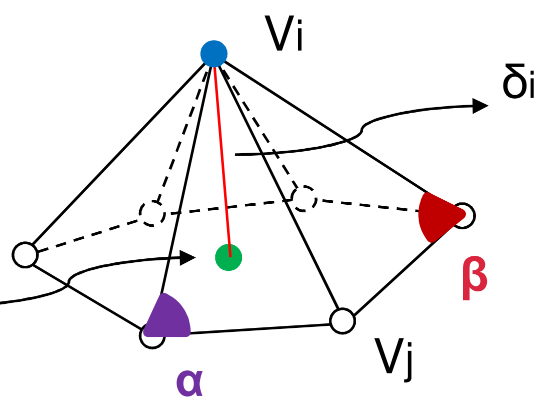
Thank you

jacktsai@cmlab.csie.ntu.edu.tw

Laplacain coordinates

- **Laplacain coordinates**

$$\frac{1}{d_i}\sum_{j\in N_i} V_j$$



$$\delta_i = V_i - \frac{1}{d_i}\sum_{j\in N_i} V_j$$

- d_i = degree of V_i (uniform weights)
- $d_i = \cot\alpha + \cot\beta$ (cotangent weights)

Laplacain coordinates

- **If we consider the rotation …**

$$\begin{bmatrix} V_i.x & V_i.y & V_i.z & 1 \\ V_1.x & V_1.y & V_1.z & 1 \\ V_2.x & V_2.y & V_2.z & 1 \\ & \vdots & & \\ V_n.x & V_n.y & V_n.z & 1 \end{bmatrix} \begin{bmatrix} \text{Rotation} \\ \text{matrix} \end{bmatrix} = \begin{bmatrix} V_i'.x & V_i'.y & V_i'.z & 1 \\ V_1'.x & V_1'.y & V_1'.z & 1 \\ V_2'.x & V_2'.y & V_2'.z & 1 \\ & \vdots & & \\ V_n'.x & V_n'.y & V_n'.z & 1 \end{bmatrix}$$

$$\mathbf{V} \quad \mathbf{R} \quad = \quad \mathbf{V'}$$

$$\mathbf{R = (V^T V)^{-1} V^T V'}$$

Laplacain coordinates

- If we consider the rotation …

$$L\,V' = L\,VR$$

$$= L\,V\,(V^TV)^{-1}\,V^T\,V'$$

$$V\,R = V'$$

$$R = (V^TV)^{-1}\,V^T\,V'$$

$$L\,(V' - V\,(V^TV)^{-1}\,V^T\,V'\,) = 0$$

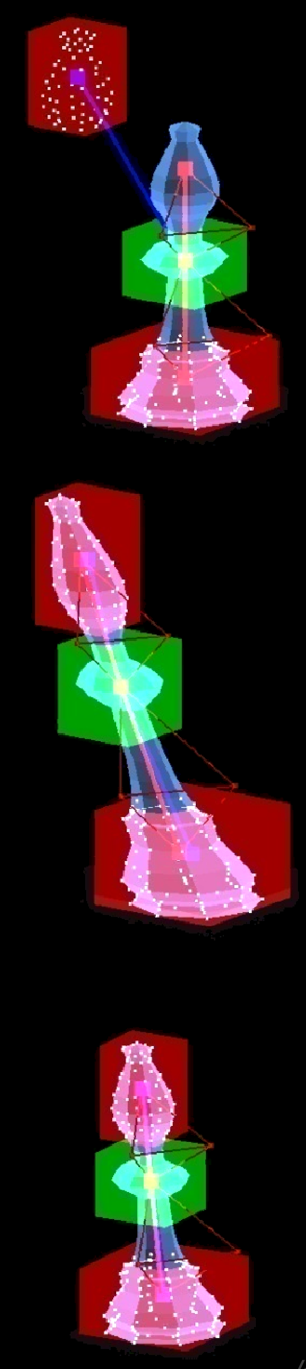
$$L\,(1 - V\,(V^TV)^{-1}\,V^T)\,V' = 0$$

Length constraint

- **Length constraint**
  - Control the length of the "bones"

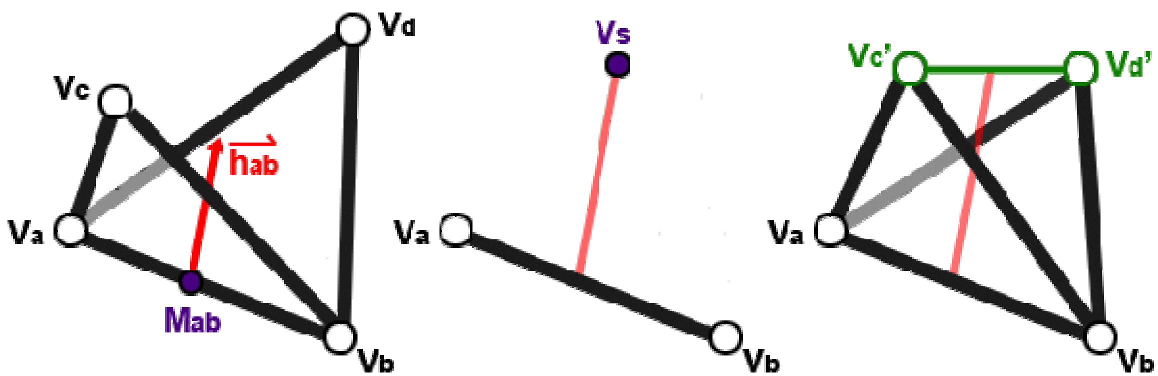
$$\sum_{(i,j)\in \text{bones}} \left(\|\mathbf{v}_i - \mathbf{v}_j\| - L_{ij} \right)^2$$

$$\left\| (\mathrm{v}_i - \mathrm{v}_j) - \frac{\mathrm{v}_i{}' - \mathrm{v}_j{}'}{\|\mathrm{v}_i{}' - \mathrm{v}_j{}'\|} \mathrm{L}_{ij} \right\|^2$$



After Rigidity constraint

- **Rigidity constraint**
  - After deformation, we rebuild new tetrabones to be reused in next times

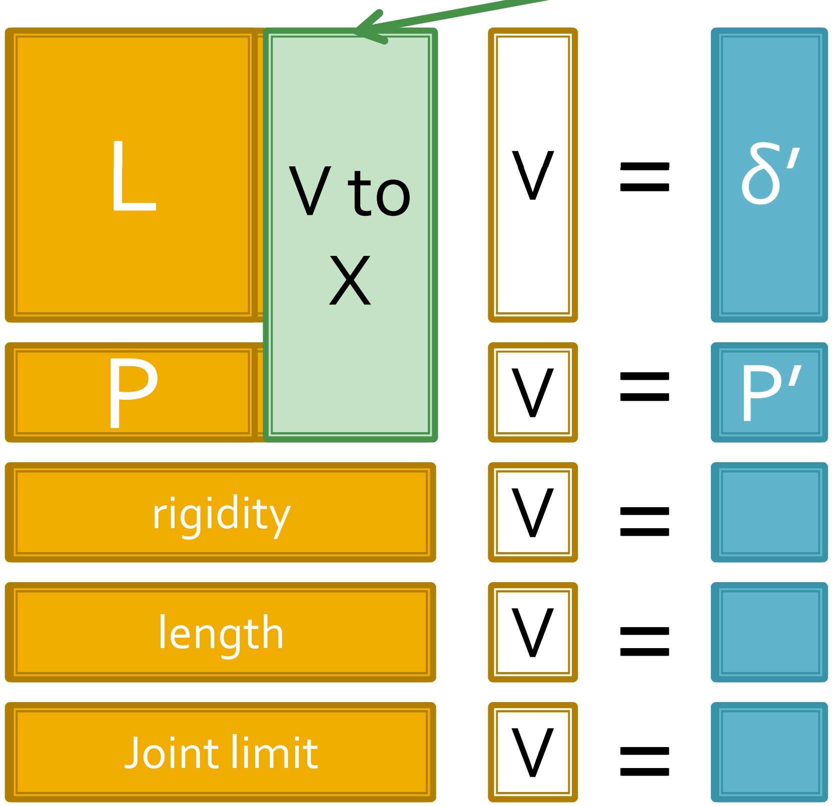


System

Method 1 :
Only solve V

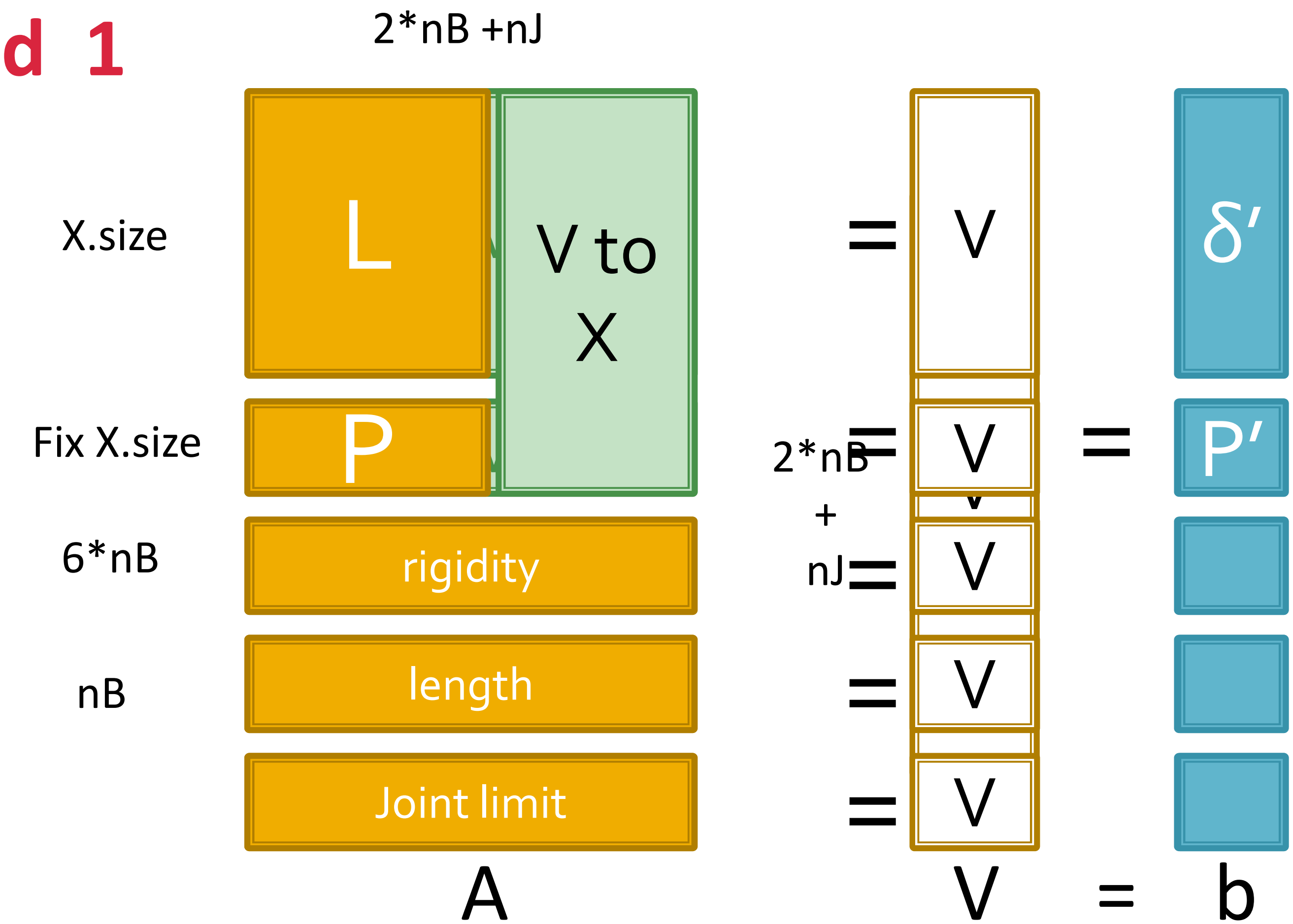
$$X = V\boxed{\overline{V}^{-1} W \overline{X}}$$

Initial W



System

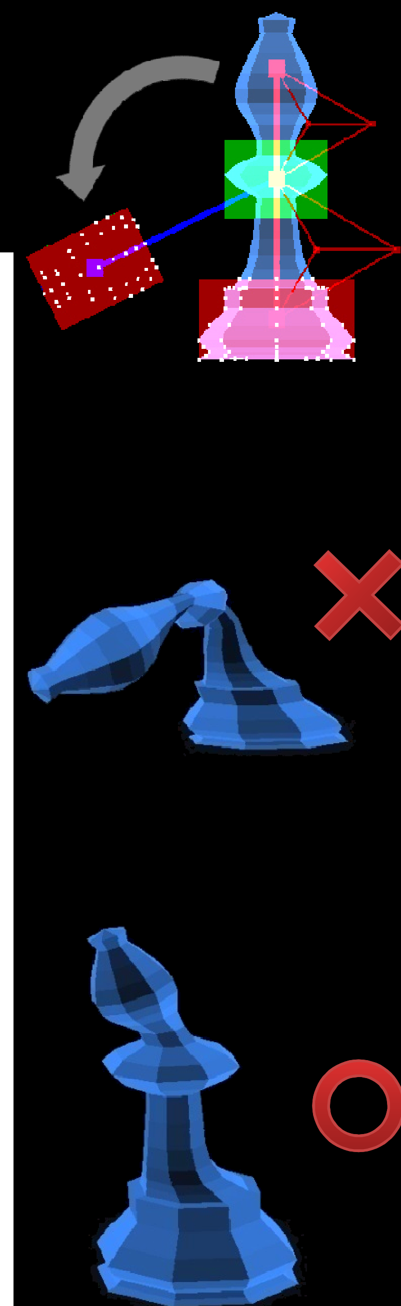
Method 1



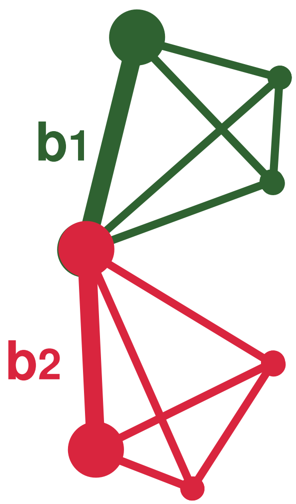
→get V → $X = V\bar{V}^{-1}W\bar{X}$ → get X !!

Joint angle limit constraint

V-step



- **Joint angle limit constraint**
  - Restrict the range of joint angles for added realism



$$\sum_{(i,j)\in \mathrm{pairs}(b_1,b_2)} \left\|(\mathbf{v}_i - \mathbf{v}_j) - \theta_{ij}\right\|^2$$

$$\Downarrow$$

$$\sum_{(i,j)\in \mathrm{pairs}(b_i,b_j)} \left(\|\mathrm{v}_i - \mathrm{v}_j\| - \mathrm{d}_{ikj}\mathrm{l}_{ikj}\right)^2$$

$$\mathrm{l}_{ikj} = \sqrt{\mathrm{l}_{ik}{}^2 + \mathrm{l}_{jk}{}^2 - 2\,\mathrm{l}_{ik}\,\mathrm{l}_{jk}\cos\theta_{ikj}}$$

- v_i, v_j : the position of tetravertices i , j
- Θ_{ij} : the target vector between tetravertices i , j