A 2D Surgical Simulation Framework for Tool-Tissue Interaction

Yunhai Han
Advanced Robotics and Controls Lab
Contextual Robotics Institute
University of California San Diego
www.ucsdarclab.com

UC San Diego
Jacobs School of Engineering

Introduction

Challenges of surgical simulation for autonomy

Introduction

Challenges of surgical simulation for autonomy

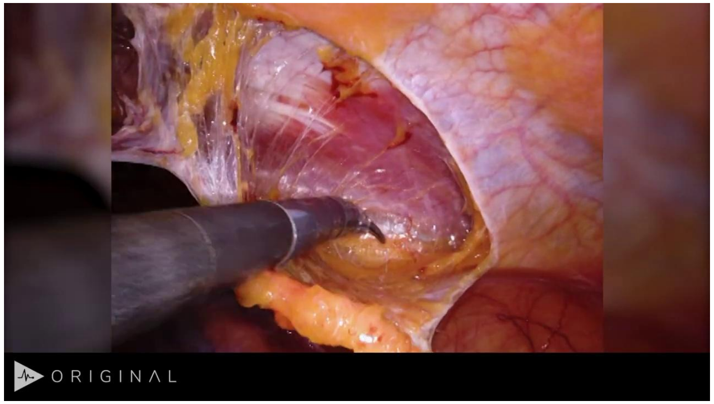


Figure captured from youtube.com/watch?v=glkhWjzjCHM&t=1043s

Introduction

Challenges of surgical simulation for autonomy

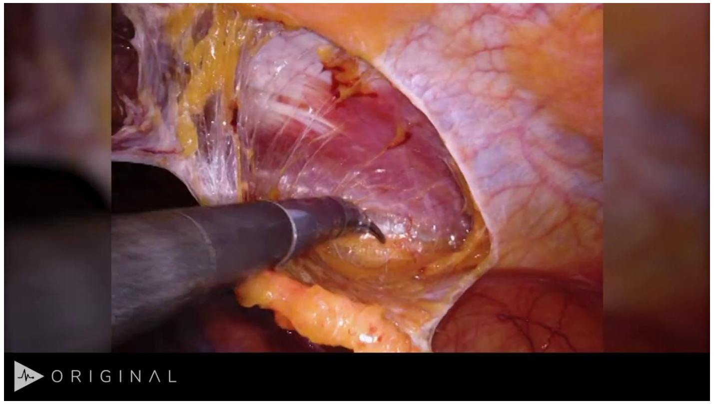


Figure captured from youtube.com/watch?v=glkhWjzjCHM&t=1043s

Simulation

Introduction

Challenges of surgical simulation for autonomy

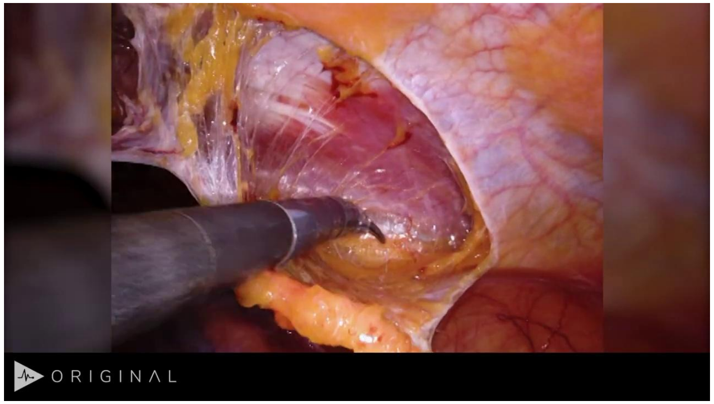


Figure captured from youtube.com/watch?v=glkhWjzjCHM&t=1043s

Simulation

- Physical Modeling

Introduction

Challenges of surgical simulation for autonomy

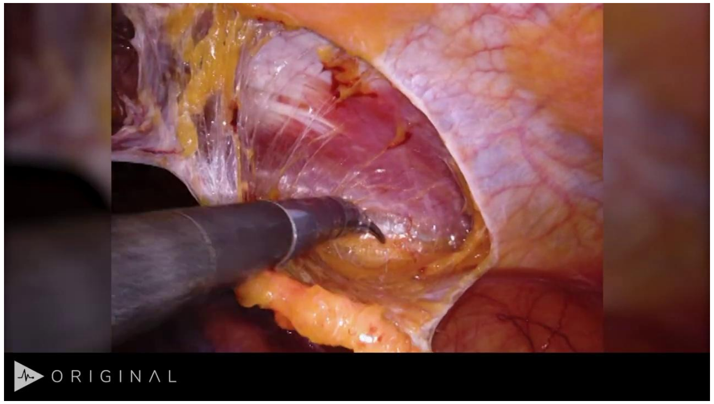


Figure captured from youtube.com/watch?v=glkhWjzjCHM&t=1043s

Simulation →

- Physical Modeling
- Visual Reconstruction

Introduction

Challenges of surgical simulation for autonomy

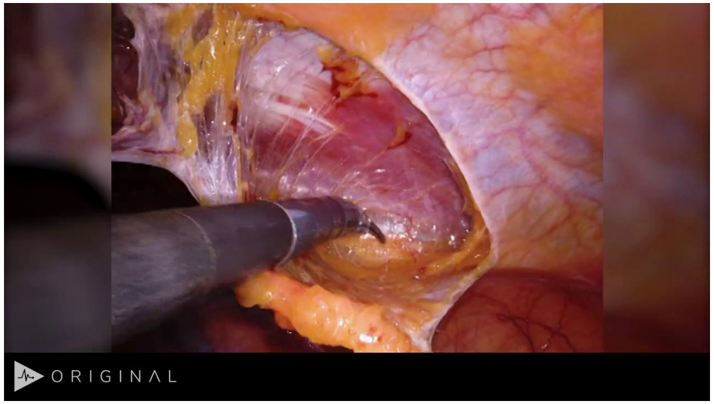


Figure captured from youtube.com/watch?v=glkhWjzjCHM&t=1043s

Simulation →

- Physical Modeling
- Visual Reconstruction
- Combined System

Introduction

Challenges of surgical simulation for autonomy

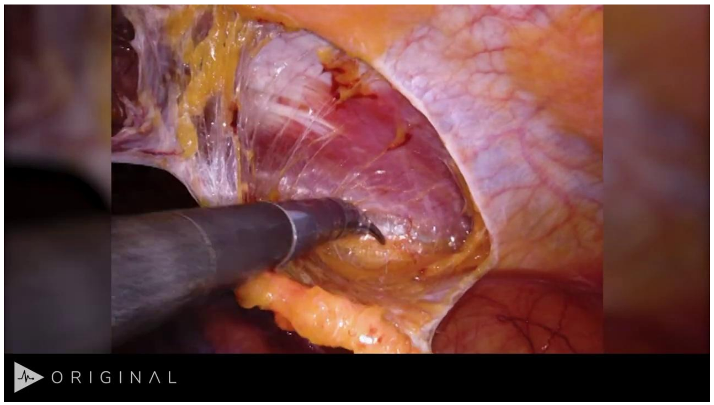


Figure captured from youtube.com/watch?v=glkhWjzjCHM&t=1043s

Simulation →

- Physical Modeling
- Visual Reconstruction
- Combined System

Goals :

Introduction

Challenges of surgical simulation for autonomy

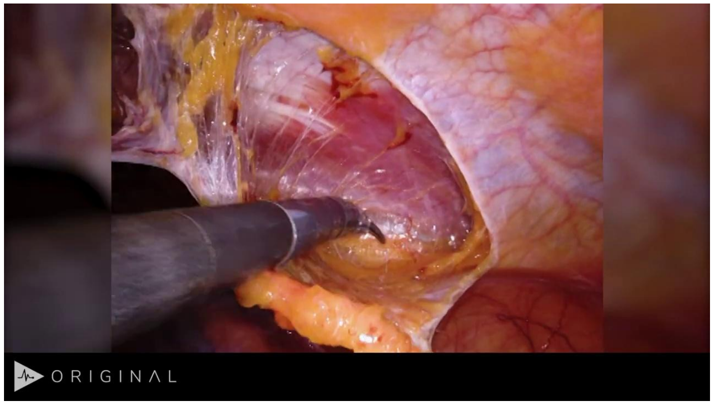


Figure captured from youtube.com/watch?v=glkhWjzjCHM&t=1043s

Simulation →

- Physical Modeling
- Visual Reconstruction
- Combined System

Goals :

- tool tracking

Introduction

Challenges of surgical simulation for autonomy

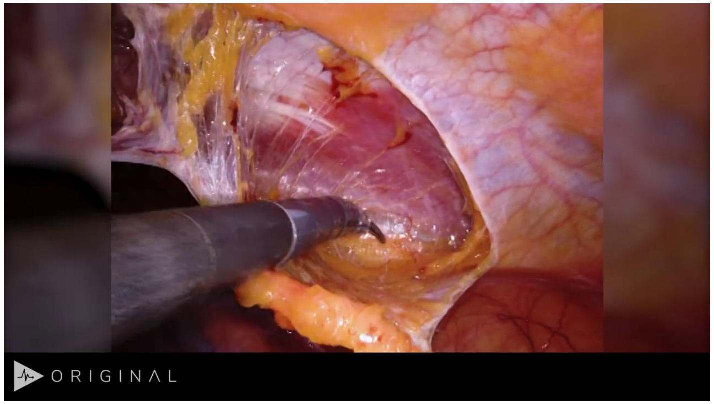


Figure captured from youtube.com/watch?v=glkhWjzjCHM&t=1043s

Simulation →

- Physical Modeling
- Visual Reconstruction
- Combined System

Goals :

- tool tracking
- deal with tool-tissue interaction

Challenges of surgical simulation for autonomy

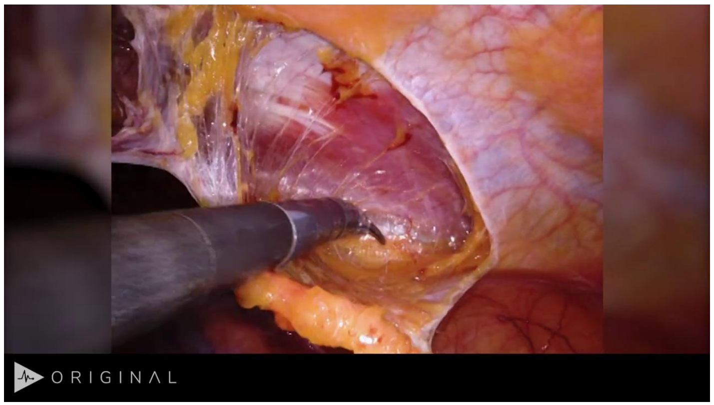


Figure captured from youtube.com/watch?v=glkhWjzjCHM&t=1043s

Simulation →

- Physical Modeling
- Visual Reconstruction
- Combined System

Goals :

- tool tracking
- deal with tool-tissue interaction
- fast, real-time

Introduction

Challenges of surgical simulation for autonomy

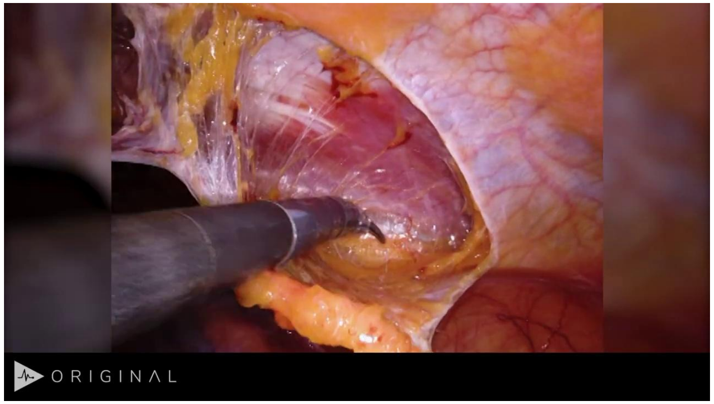


Figure captured from youtube.com/watch?v=glkhWjzjCHM&t=1043s

Simulation →

- Physical Modeling
- Visual Reconstruction
- Combined System

Goals :

- tool tracking
- deal with tool-tissue interaction
- fast, real-time
- can be embedded into closed-loop control

Introduction

Contribution of our work

A novel 2D simulation framework includes four modules:

Introduction

Contribution of our work

A novel 2D simulation framework includes four modules:

1. Mesh generation using 2D tissue images
2. Position-based dynamics methods for tissue simulation
3. Collision detection method for tool-tissue interaction
4. Implicit Euler energy computation

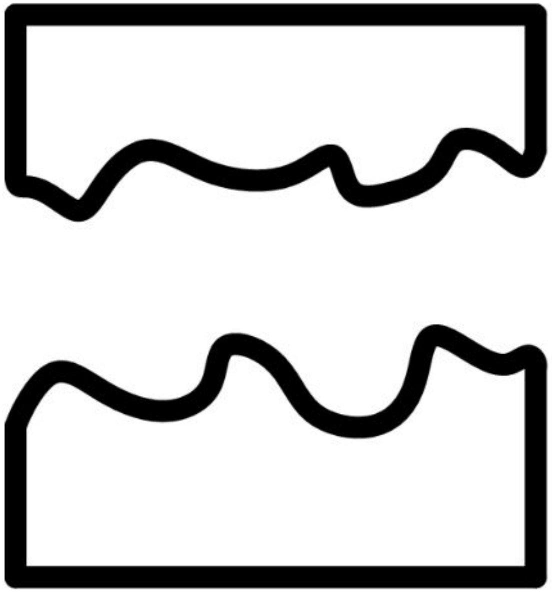
Methodology

Mesh Generation

Methodology

Mesh Generation

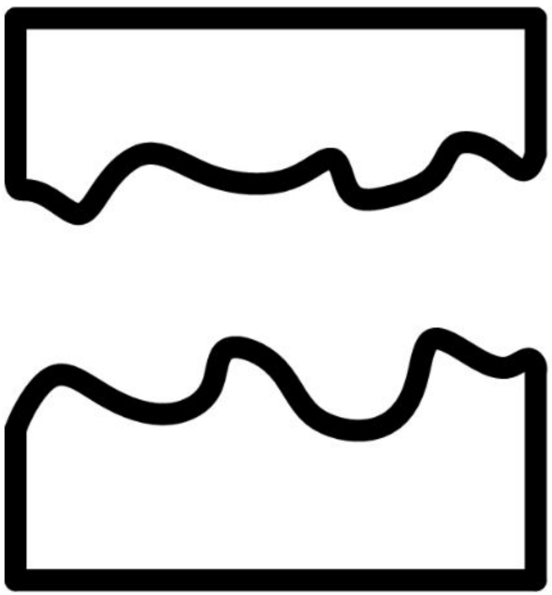
Image



Methodology

Mesh Generation

Image

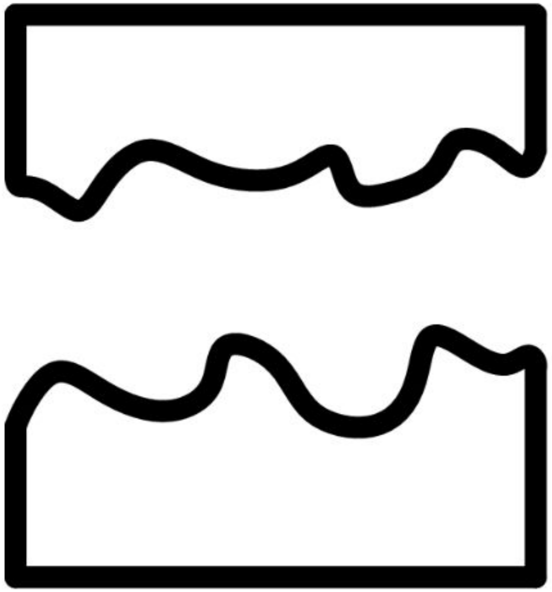


Contour points detection

Methodology

Mesh Generation

Image



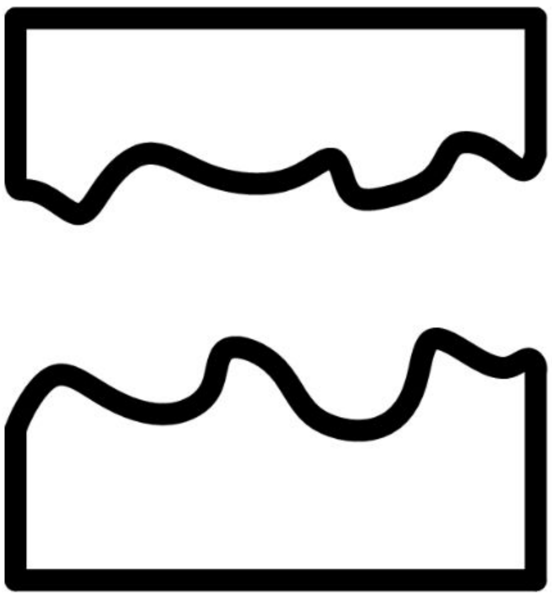
Contour points detection

Delaunay Triangulator

Methodology

Mesh Generation

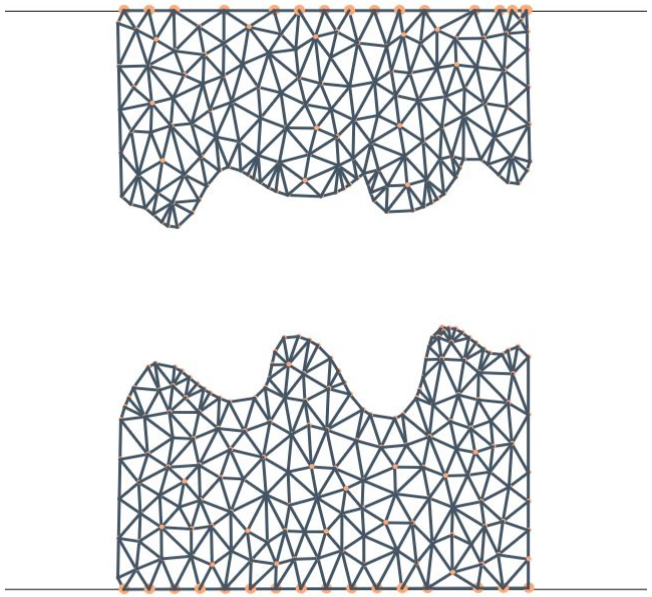
Image



Contour points detection

Delaunay Triangulator

Mesh



Methodology

Algorithm: Simulation Process (Position-based dynamics Simulation) [1]

[1] J. Bender, M. Müller, and M. Macklin, "Position-based simulation methods in computer graphics," in EUROGRAPHICS 2017 Tutorials. Eurographics Association, 2017.

Methodology

Algorithm: Simulation Process (Position-based dynamics Simulation) [1]

$$\mathbf{x}^* = \mathbf{x}^t + \Delta t \mathbf{v}^t + \Delta t^2 \mathbf{M}^{-1} \mathbf{f}_{ext}(\mathbf{x}^{\mathbf{t}})$$ ▷ **prediction step**

[1] J. Bender, M. Müller, and M. Macklin, “Position-based simulation methods in computer graphics,” in EUROGRAPHICS 2017 Tutorials. Eurographics Association, 2017.

Methodology

Algorithm: Simulation Process (Position-based dynamics Simulation) [1]

$\mathbf{x}^{*} = \mathbf{x}^{t} + \Delta t \mathbf{v}^{t} + \Delta t^{2} \mathrm{M}^{-1} \mathbf{f}_{ext}(\mathbf{x}^{\mathbf{t}})$ ▷ **prediction step**

While *iter < SolverIterations* **do**

  for *constraint C* $\in$ *M* **do**

    Compute $\Delta \mathbf{x}$ ▷ **constraint solving step**

    $\mathbf{x} = \mathbf{x}^{*} + \Delta \mathbf{x}$

  end

end

[1] J. Bender, M. Müller, and M. Macklin, "Position-based simulation methods in computer graphics," in EUROGRAPHICS 2017 Tutorials. Eurographics Association, 2017.

Methodology

Algorithm: Simulation Process (Position-based dynamics Simulation) [1]

$$\mathbf{x}^* = \mathbf{x}^t + \Delta t \mathbf{v}^t + \Delta t^2 \mathbf{M}^{-1} \mathbf{f}_{ext}(\mathbf{x}^t)$$ ▷ **prediction step**

While *iter < SolverIterations* **do**

  for *constraint C* ∈ *M* **do**

    Compute $\Delta \mathbf{x}$ ▷ **constraint solving step**

    $$\mathbf{x} = \mathbf{x}^* + \Delta \mathbf{x}$$

  end

end

$$\mathbf{x}^{t+1} = \mathbf{x}$$ ▷ **update position**

$$\mathbf{v}^{t+1} = \left(\mathbf{x}^{t+1} - \mathbf{x}^t\right) / \Delta t$$ ▷ **update velocity**

[1] J. Bender, M. Müller, and M. Macklin, "Position-based simulation methods in computer graphics," in EUROGRAPHICS 2017 Tutorials. Eurographics Association, 2017.

Methodology

Position-based dynamics Simulation

Solver: The position correction ΔX in each iteration can be computed on based on the constraint violation and its derivatives through Gauss-Siedel Method.

Methodology

Position-based dynamics Simulation

Solver: The position correction ΔX in each iteration can be computed on based on the constraint violation and its derivatives through Gauss-Siedel Method.

- Distance constraint:

$$C_{\text{spring}}(\mathbf{x}_1, \mathbf{x}_2) = |\mathbf{x}_1 - \mathbf{x}_2| - d_0$$

Methodology

Position-based dynamics Simulation

Solver: The position correction ΔX in each iteration can be computed on based on the constraint violation and its derivatives through Gauss-Siedel Method.

- Distance constraint:

$$C_{\text{spring}}\left(\mathbf{x}_1, \mathbf{x}_2\right) = \left|\mathbf{x}_1 - \mathbf{x}_2\right| - d_0$$

- Area conservation:

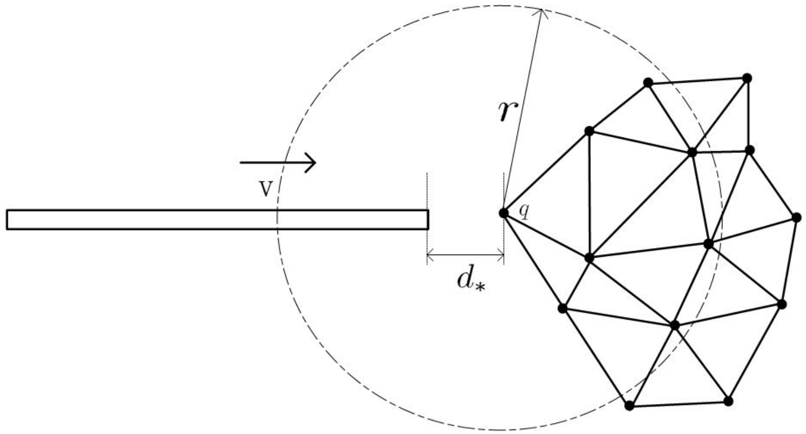
$$C_{\text{area}}\left(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\right) = \frac{1}{2}\left|\left(\mathbf{x}_2 - \mathbf{x}_1\right) \times \left(\mathbf{x}_3 - \mathbf{x}_1\right)\right| - A_0$$

Methodology

Collision Detection for Tool-Tissue Interaction

Collision Detection for Tool-Tissue Interaction

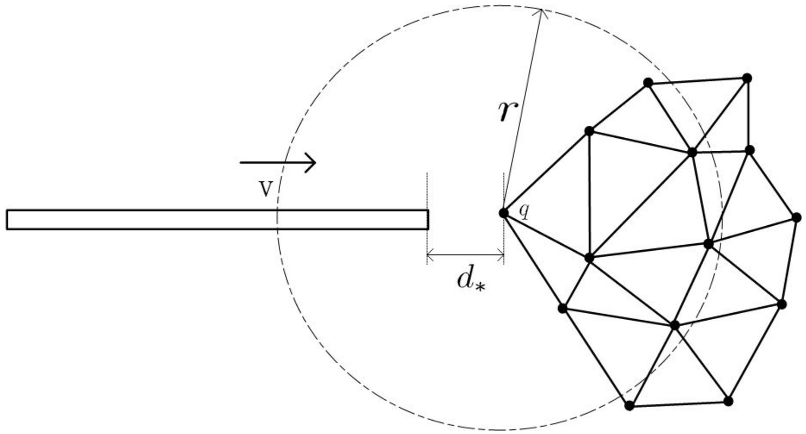
The collision point is defined as the particle *q* on the mesh when the distance d_* between it and the tool is smaller than a fixed threshold α.



Methodology

Collision Detection for Tool-Tissue Interaction

The collision point is defined as the particle *q* on the mesh when the distance d_* between it and the tool is smaller than a fixed threshold *α*.



Position update for each particle *i*:

$$x_i^{\mathrm{update}} = \begin{cases} \frac{r - \|\mathbf{x}_q - \mathbf{x}_i\|}{r} d_* \frac{\mathbf{v}_{\mathrm{mani}}}{\|\mathbf{v}_{\mathrm{mani}}\|}, & \text{if } r - \|\mathbf{x}_q - \mathbf{x}_i\| \geq 0 \\ 0, & \text{otherwise} \end{cases}$$

Methodology

Implicit Euler Energy Computation

Contains both the inertial and potential terms:

$$E(x) = \frac{1}{2} \left\| \mathbf{x}^{t+1} - \mathbf{x}^* \right\|_{\mathbf{M}}^2 + \Delta t^2 E_p(\mathbf{x}^{t+1})$$

Implicit Euler Energy Computation

Contains both the inertial and potential terms:

$$E(x) = \frac{1}{2}\left\|\mathbf{x}^{t+1} - \mathbf{x}^{*}\right\|_{\mathbf{M}}^{2} + \Delta t^{2} E_{p}(\mathbf{x}^{t+1})$$

Potential energy = spring elastic energy + area conservation energy

$$E_i = \frac{1}{2}\mathbf{C}_i^T(\mathbf{x})\mathbf{K}_i\mathbf{C}_i(\mathbf{x})$$

Methodology

Implicit Euler Energy Computation

Contains both the inertial and potential terms:

$$E(x) = \frac{1}{2}\left\|\mathbf{x}^{t+1} - \mathbf{x}^*\right\|_{\mathbf{M}}^2 + \Delta t^2 E_p(\mathbf{x}^{t+1})$$

Potential energy = spring elastic energy + area conservation energy

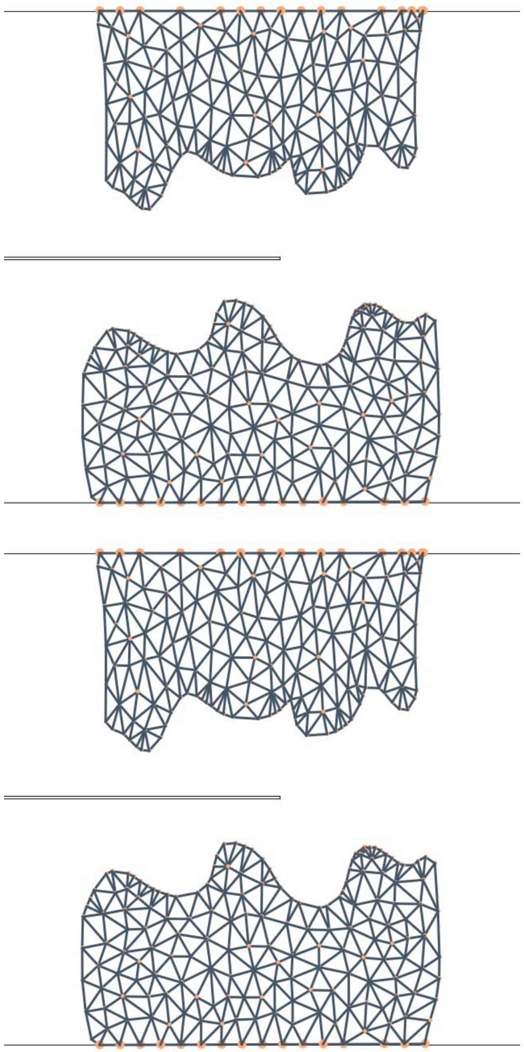
$$E_i = \frac{1}{2}\mathbf{C}_i^T(\mathbf{x})\mathbf{K}_i\mathbf{C}_i(\mathbf{x})$$

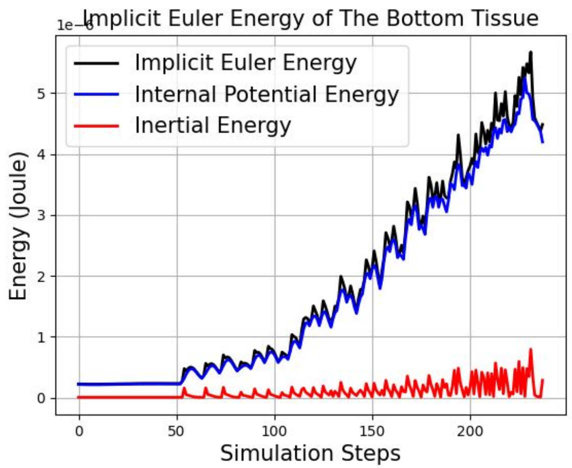
where,

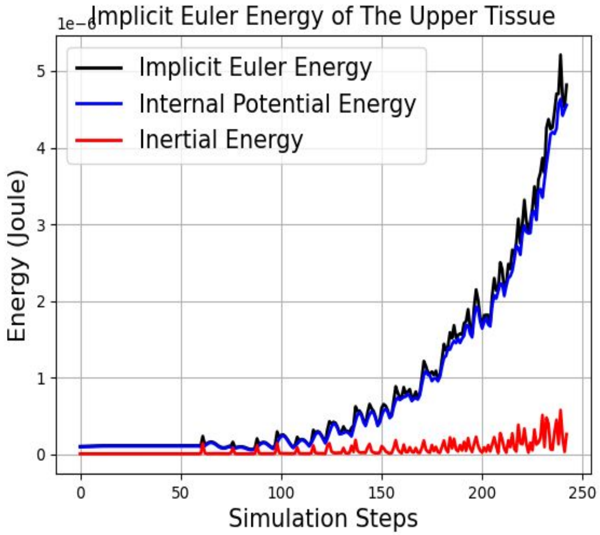
$$\mathbf{C}_i = \left[\mathbf{C}_\mathbf{i}^{\mathrm{Spring}}(\mathbf{x}), \mathbf{C}_\mathbf{i}^{\mathrm{Area}}(\mathbf{x})\right]^T$$

Experiments

Task1: The tool is approaching the bottom/upper tissue

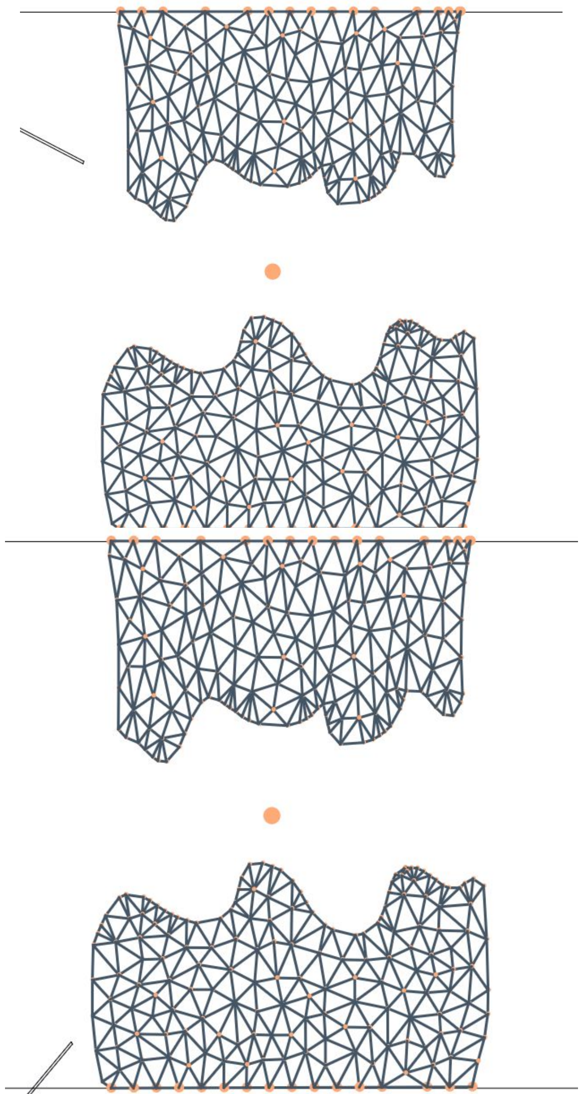


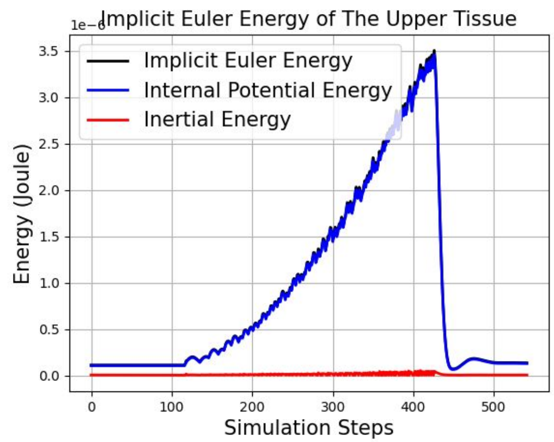


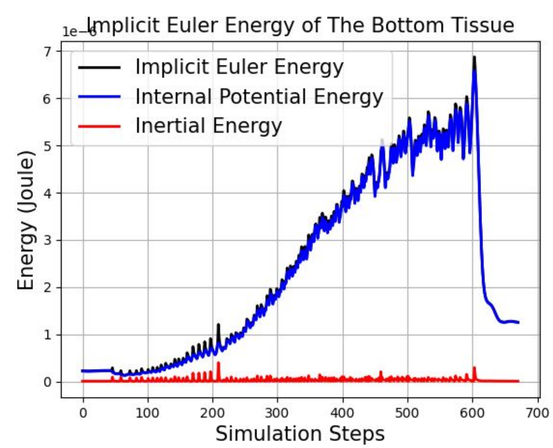


Experiments

Task2: The tool is being inserted from two different angles, aiming at the same target goal.







It is obviously to see the energy variation according to insertion angle.

Conclusion

Summary

Conclusion

- **Simulate the 2D soft tissue deformations**

Summary

Conclusion

- **Simulate the 2D soft tissue deformations**
- **Deal with tool-tissue interactions**

Conclusion

- Simulate the 2D soft tissue deformations
- Deal with tool-tissue interactions
- Energy computation for the control and planning applications

Summary

Conclusion

- Simulate the 2D soft tissue deformations
- Deal with tool-tissue interactions
- Energy computation for the control and planning applications

Future Work

Summary

Conclusion

- Simulate the 2D soft tissue deformations
- Deal with tool-tissue interactions
- Energy computation for the control and planning applications

Future Work

- Extended to 3D surgical environments

Summary

Conclusion

- Simulate the 2D soft tissue deformations
- Deal with tool-tissue interactions
- Energy computation for the control and planning applications

Future Work

- Extended to 3D surgical environments
- Embedded into closed-loop control system

Thanks for Listening!

Any Questions?