

基于GPU的非线性有限元柔体形变仿真模拟

GPU-based Soft Body Deformation Simulation with Nonlinear Finite Element Method

答辩人：郭煜¹

导师：王平安教授²

¹中国科学院深圳先进技术研究院

²香港中文大学

硕士毕业论文答辩，2013



论文大纲

第一章 绪论

虚拟手术系统结构以及柔体变形的各种方法简介。

第二章 网格处理

表面网格的简化、细分以及空间网格的剖分。

第三章 非线性有限元形变模型

有限元方程的建立以及求解。

第四章 算法实现

基于CUDA的有限元算法实现以及快速碰撞模型。

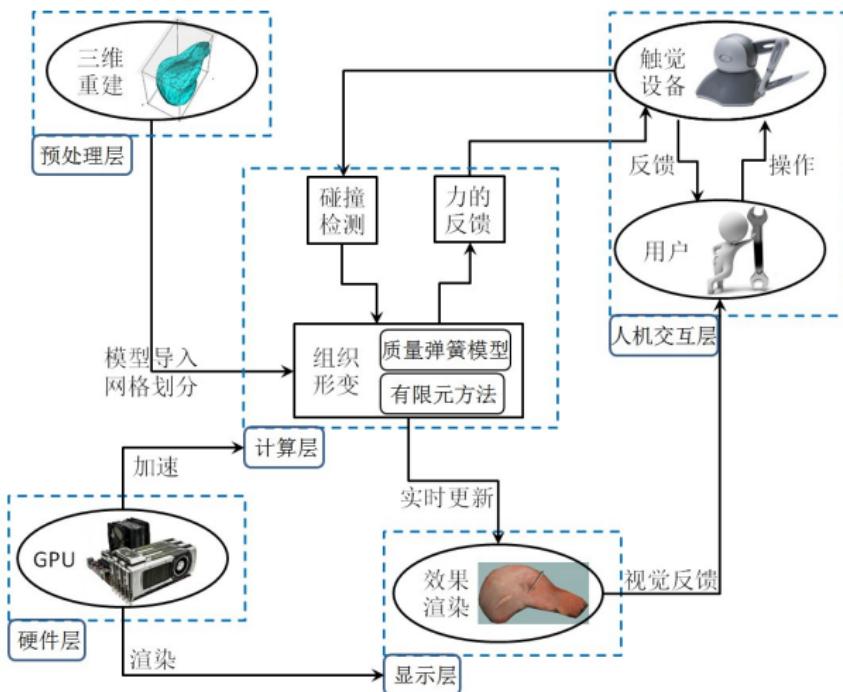
第五章 实验结果分析

对形变结果准确性的一些分析。

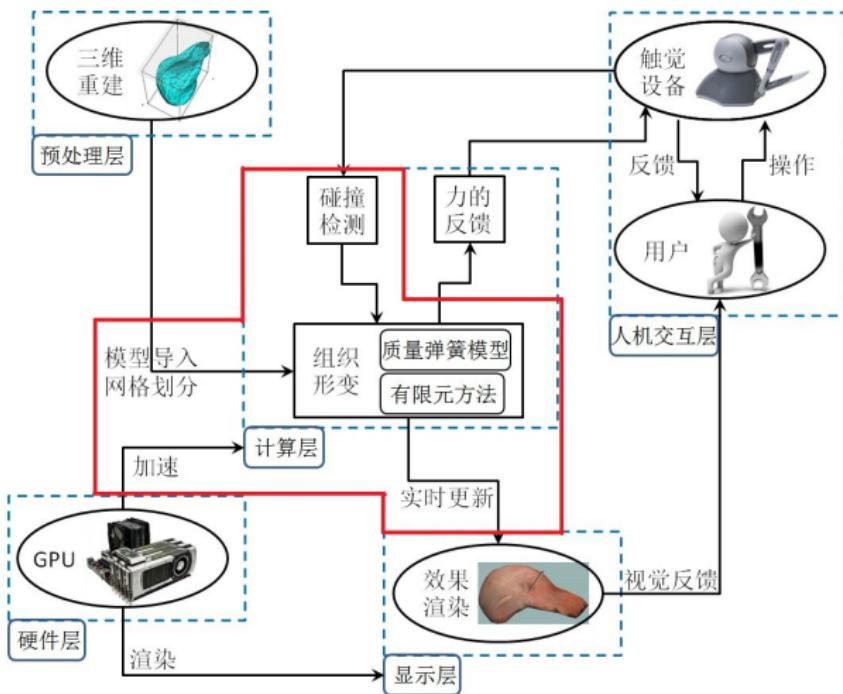
第六章 总结与展望



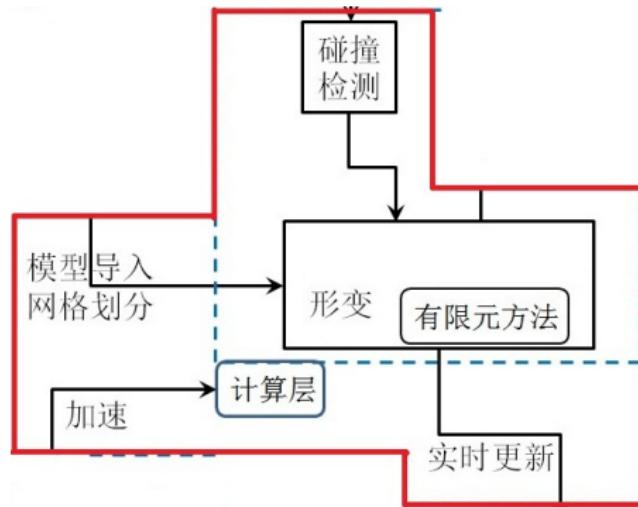
问题的提出



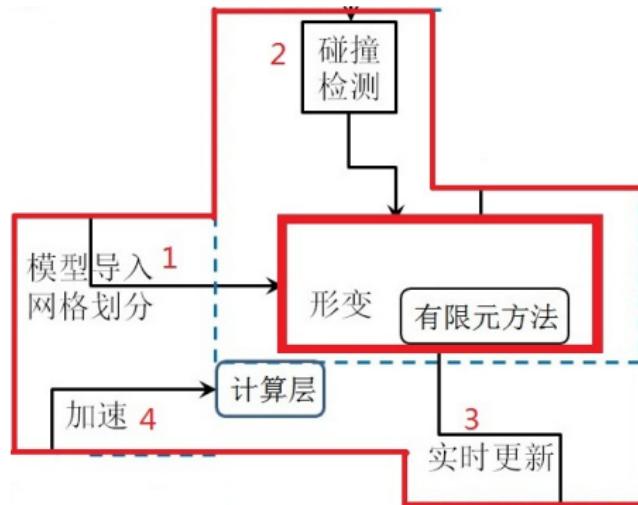
问题的提出



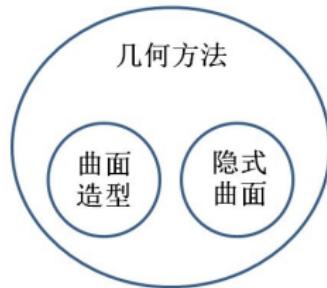
问题的提出



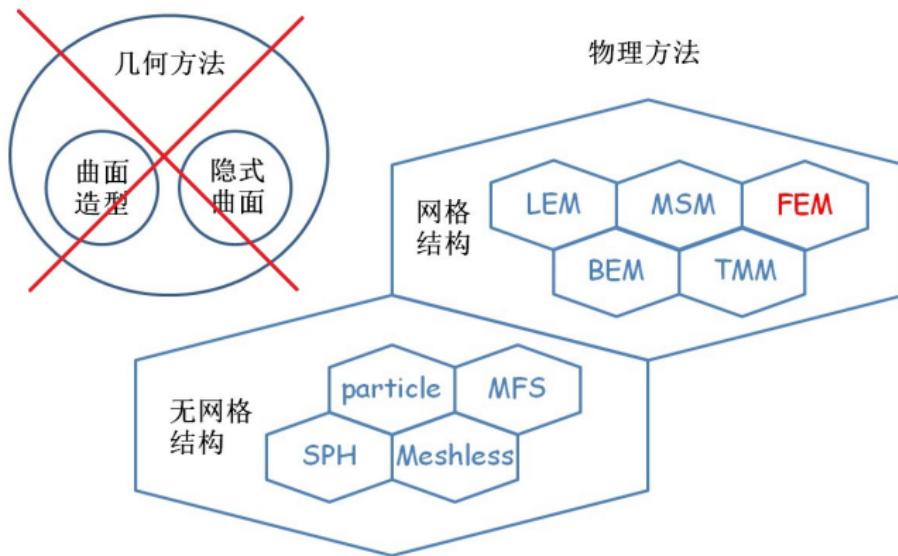
问题的提出



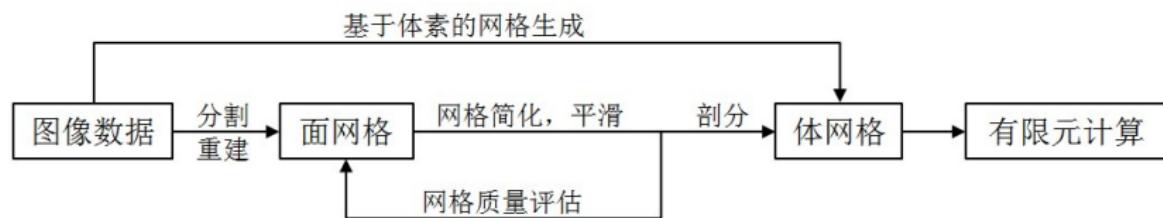
形变的计算方法



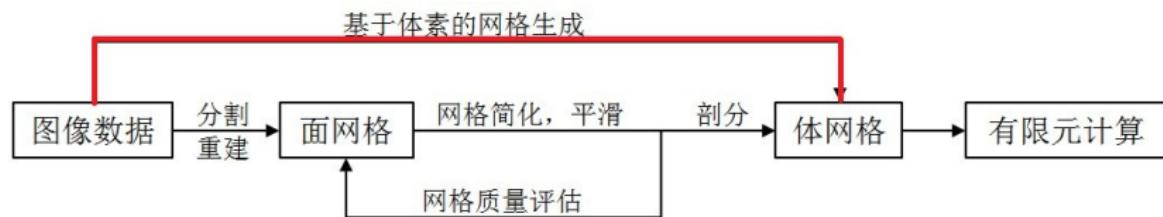
形变的计算方法



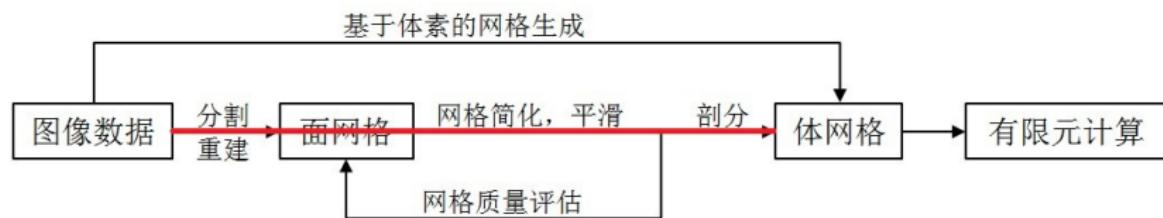
网格处理



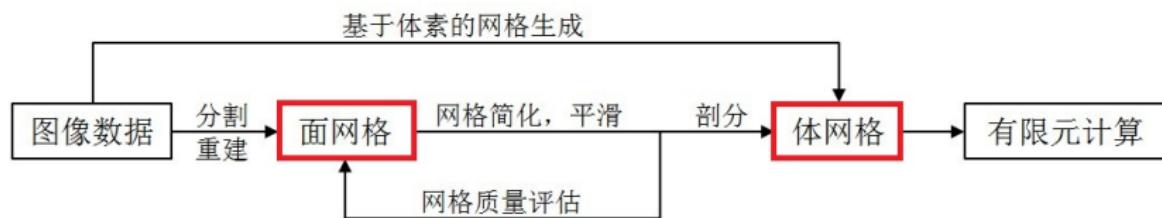
网格处理



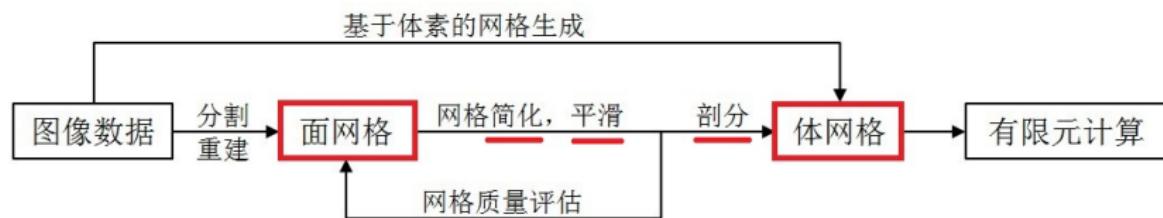
网格处理



网格处理

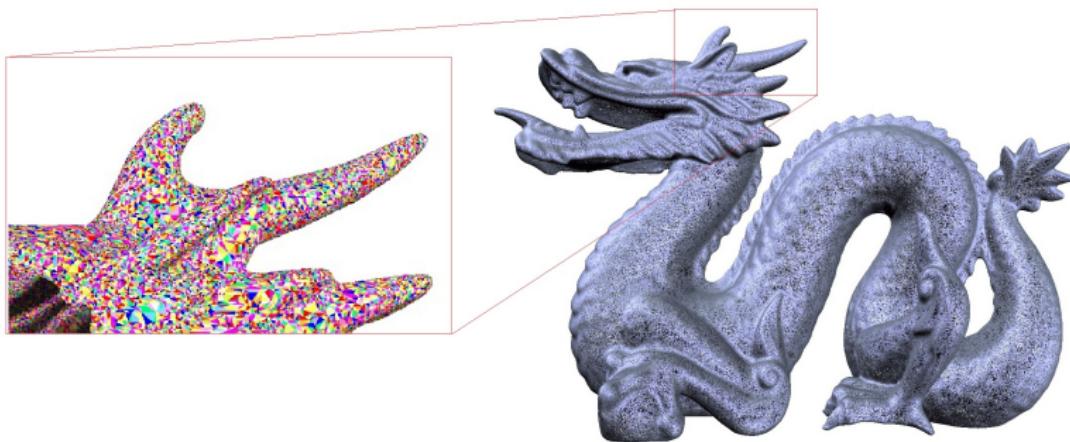


网格处理



网格简化

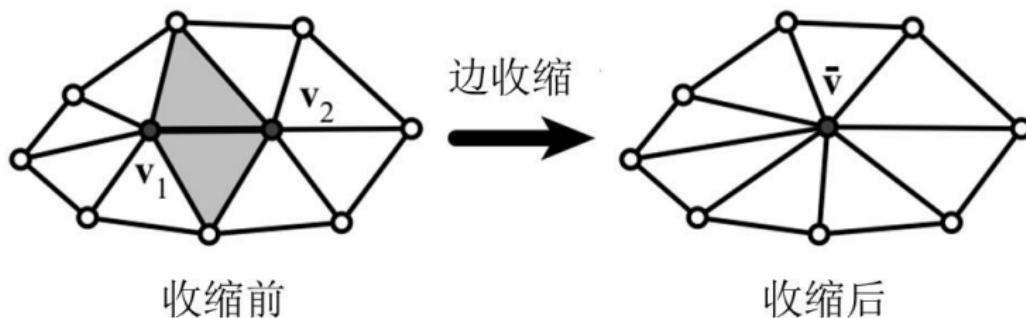
冗余



点437645，面871414

网格简化

二次误差测度 (QEM)



网格简化



Stanford 兔子。从左到右点/面数依次
是: 35947/69451; 3596/6945; 362/693



网格简化



人的大脑。从左到右点/面数依次
是：14780/28390；7439/14195；1639/2839



网格简化

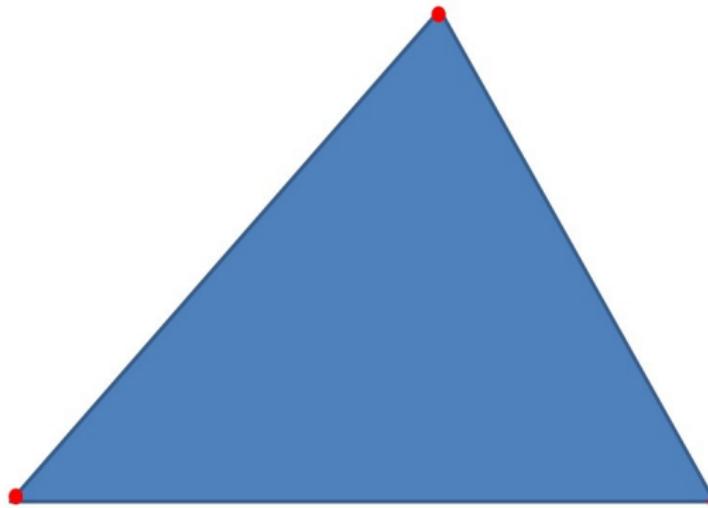


人的牙齿。从左到右点/面数依次
是：116604/233204; 1238/2331; 119/233



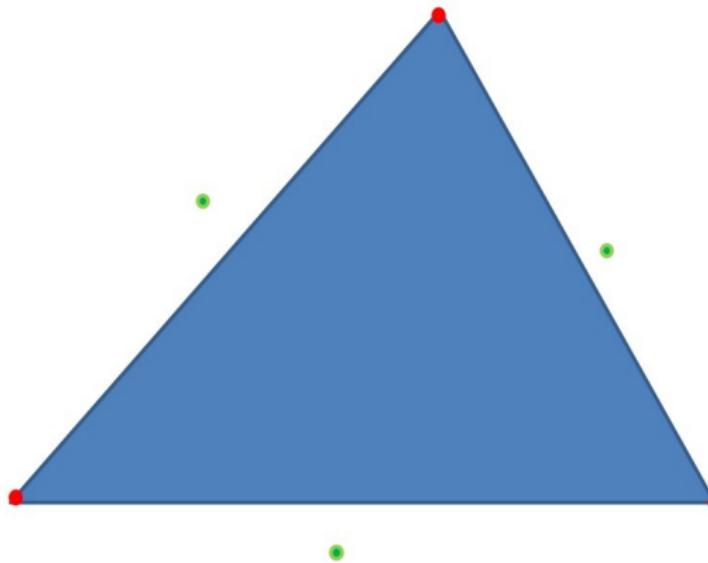
网格细分

Loop细分



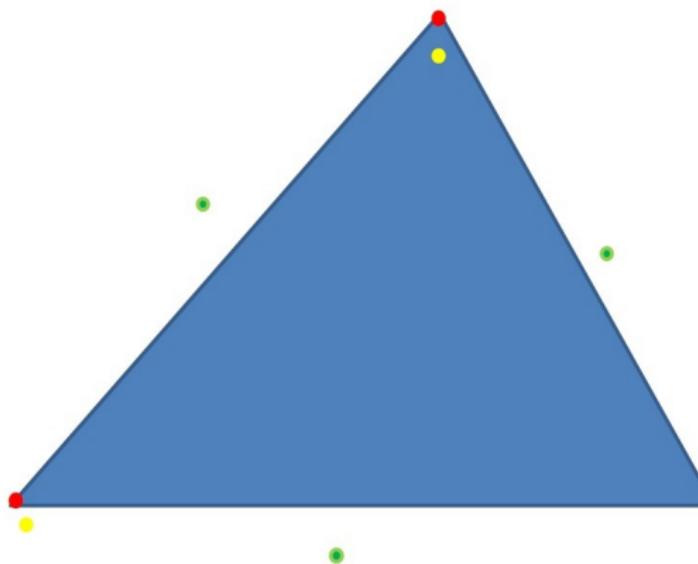
网格细分

Loop细分



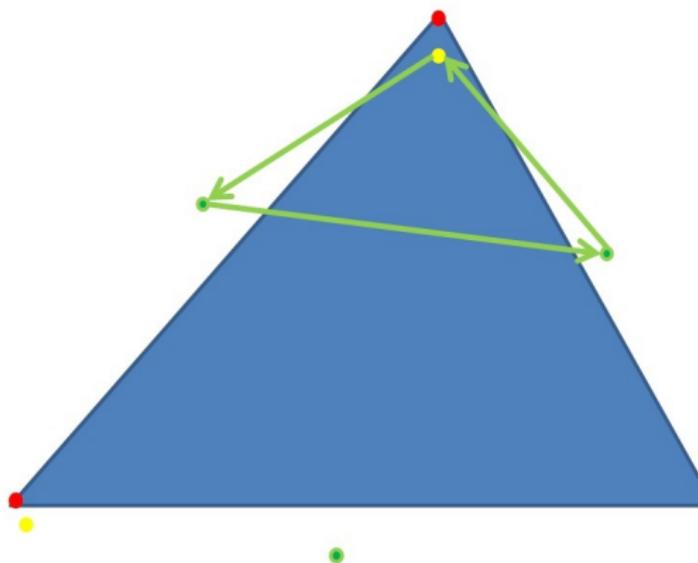
网格细分

Loop细分



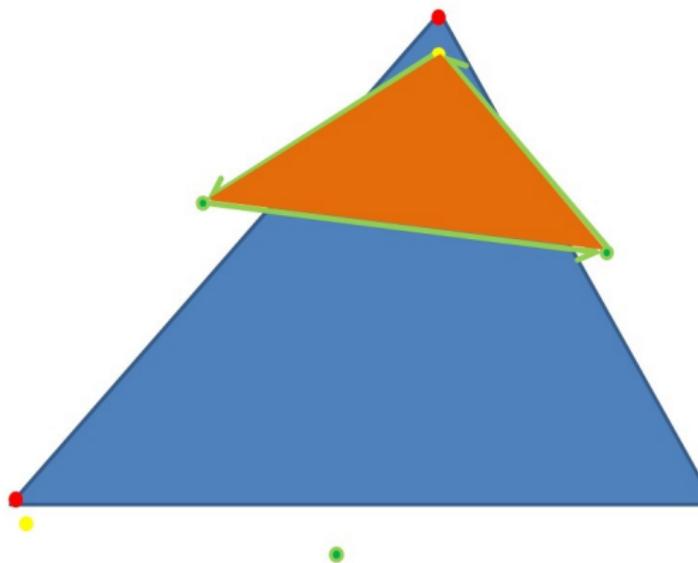
网格细分

Loop细分



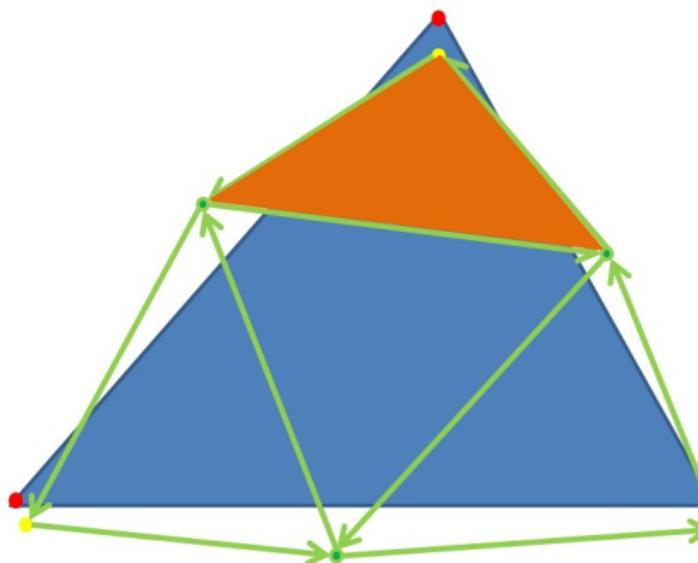
网格细分

Loop细分



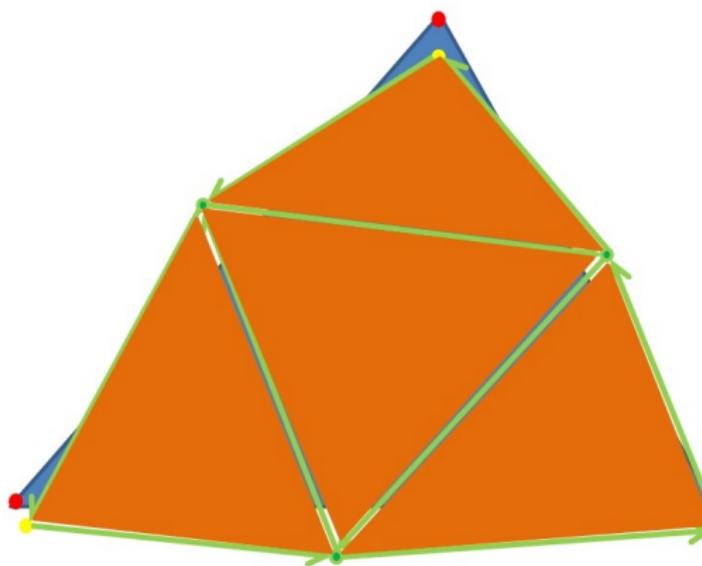
网格细分

Loop细分



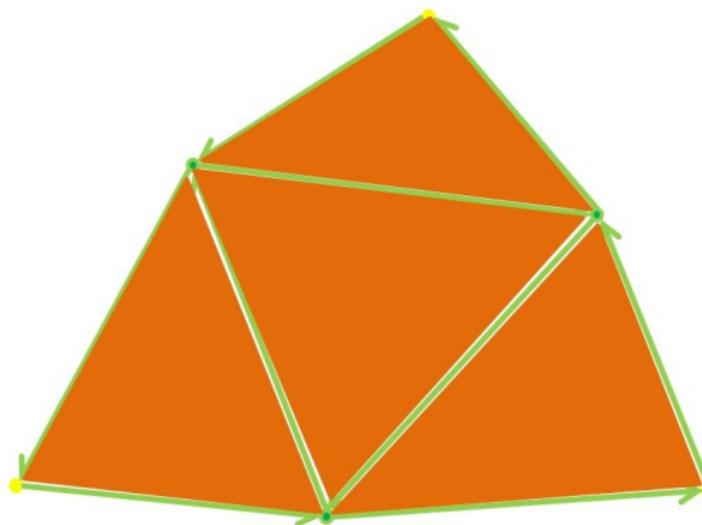
网格细分

Loop细分

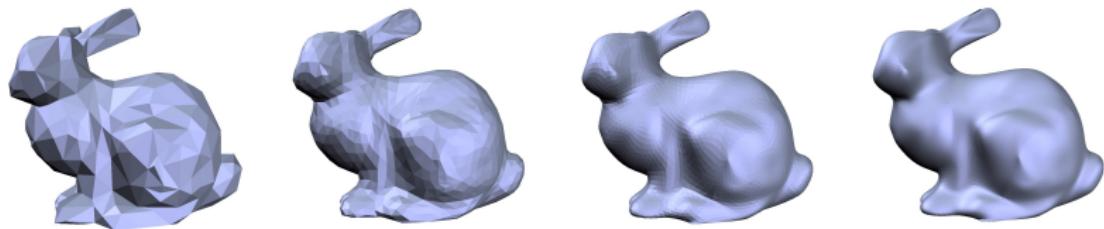


网格细分

Loop细分



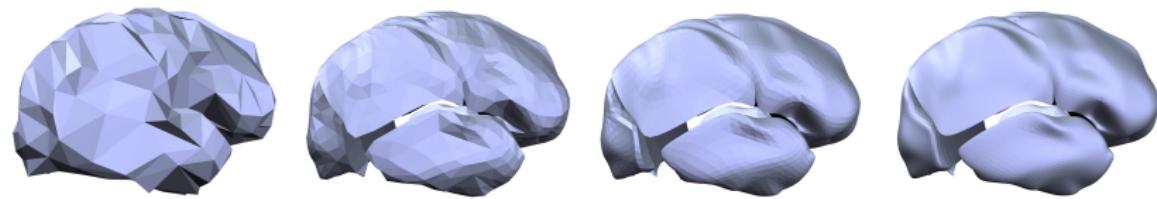
网格细分



Stanford 兔子。点/面数依次：
362/693; 1416/2772; 5603/11088; 22293/44352



网格细分

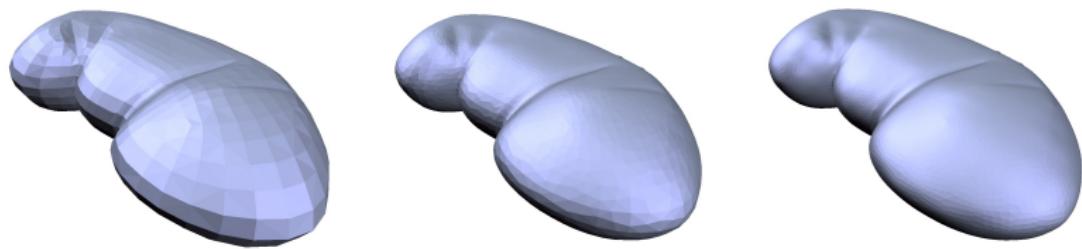


人的大脑。点/面数依

次: 509/928; 1937/3712; 7574/14848; 29972/59392



网格细分

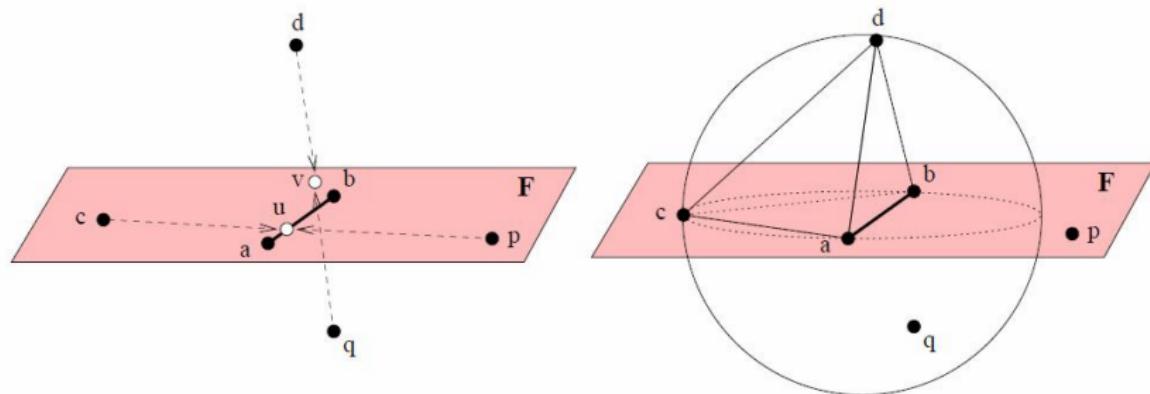


人的肾脏。点/面数依次：1212/2420; 4832/9680; 19302/38720

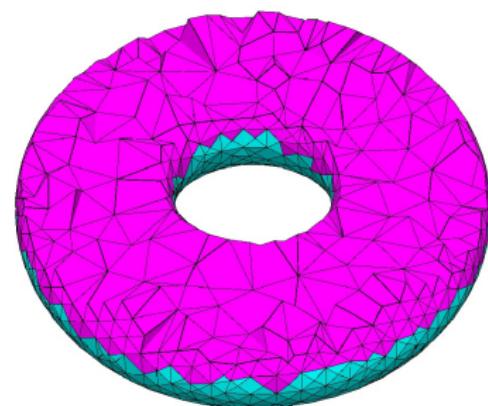
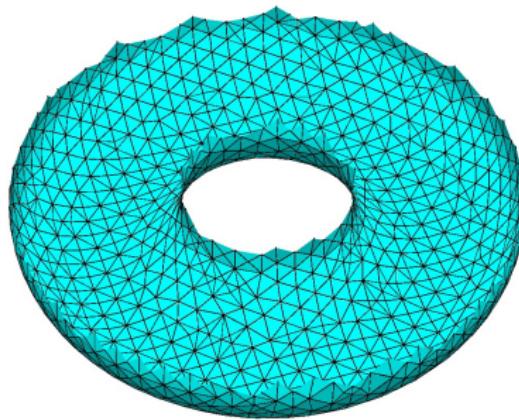


网格剖分

Delaunay四面体剖分



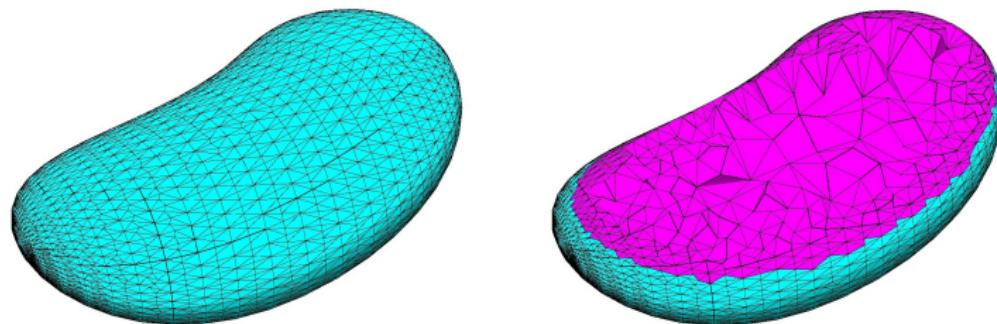
网格剖分



圆环。左：剖分前，点1729，面3458；右：剖分后，点2145，四面体7456



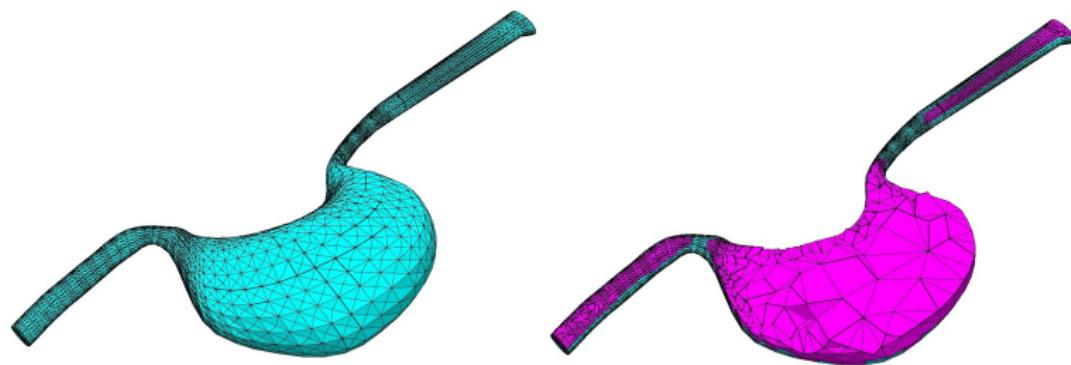
网格剖分



人的肾脏。左：剖分前，点1202，面2400；右：剖分后，
点5922，四面体24369



网格剖分



人的胃。左：剖分前，点1226，面2448；右：剖分后，点9714，
四面体37609



有限元分析

输入：外力

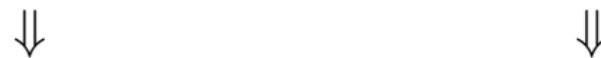


输出：位移



有限元分析

输入：外力 \implies 应力



输出：位移 \iff 应变



有限元分析

宏观

微观

输入：外力

⇒

应力



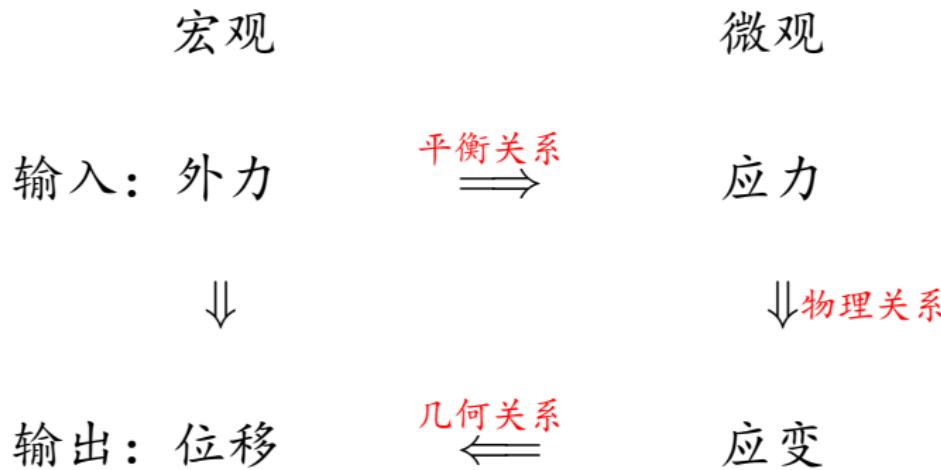
输出：位移

⇐

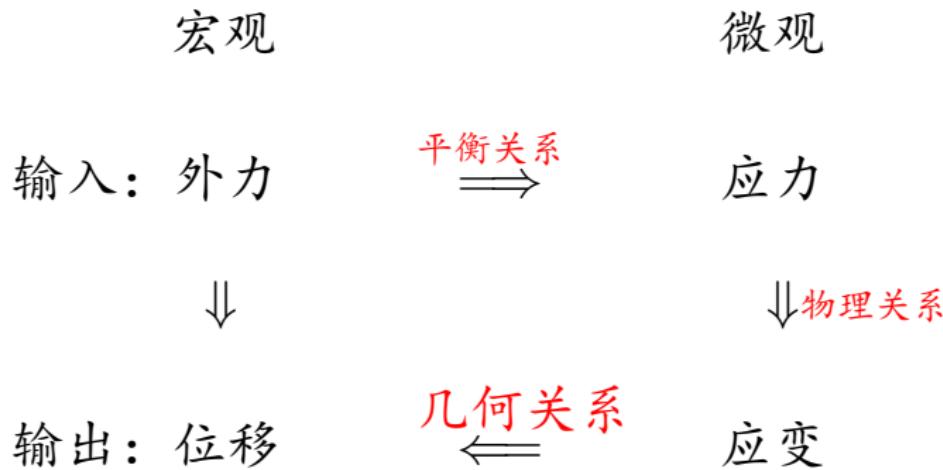
应变



有限元分析

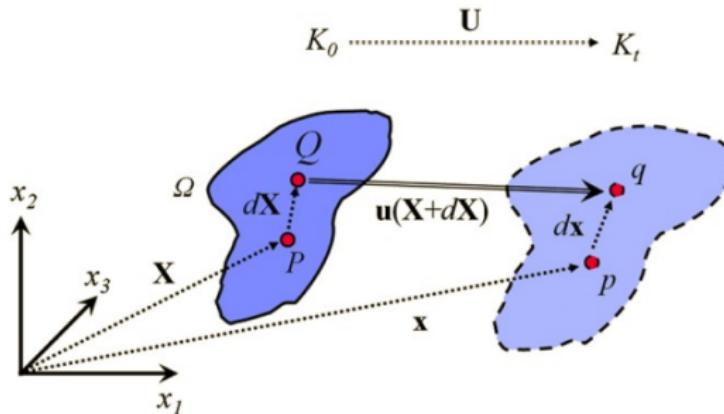


几何方程



几何方程

位移： \mathbf{U}



$$d\mathbf{x} = \left(\mathbf{I} + \frac{\mathbf{u}(\mathbf{X} + d\mathbf{X}) - \mathbf{u}(\mathbf{X})}{d\mathbf{X}} \right) d\mathbf{X} = (\mathbf{I} + \nabla_{\mathbf{X}} \mathbf{u}) d\mathbf{X}$$



几何方程

应变： \mathbf{E}

$$\begin{aligned}\mathbf{E} &= \frac{1}{2}[(\mathbf{I} + \nabla \mathbf{u})^T(\mathbf{I} + \nabla \mathbf{u}) - \mathbf{I}] \\ &= \frac{1}{2}(\nabla \mathbf{u} + \nabla \mathbf{u}^T + \nabla \mathbf{u}^T \nabla \mathbf{u})\end{aligned}$$



几何方程

应变： \mathbf{E}

$$\begin{aligned}\mathbf{E} &= \frac{1}{2}[(\mathbf{I} + \nabla \mathbf{u})^T(\mathbf{I} + \nabla \mathbf{u}) - \mathbf{I}] \\ &= \frac{1}{2} \left(\underline{\nabla \mathbf{u} + \nabla \mathbf{u}^T} + \underline{\nabla \mathbf{u}^T \nabla \mathbf{u}} \right)\end{aligned}$$

线性项 非线性项



线性几何方程

$$\boldsymbol{\varepsilon} = \frac{1}{2}(\nabla \mathbf{u} + \nabla \mathbf{u}^T)$$

$$\hat{\boldsymbol{\varepsilon}} = \begin{bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \varepsilon_{zz} \\ \gamma_{xy} \\ \gamma_{xz} \\ \gamma_{yz} \end{bmatrix} = \begin{bmatrix} \frac{\partial}{\partial X_1} & 0 & 0 \\ 0 & \frac{\partial}{\partial X_2} & 0 \\ 0 & 0 & \frac{\partial}{\partial X_3} \\ \frac{\partial}{\partial X_2} & \frac{\partial}{\partial X_1} & 0 \\ 0 & \frac{\partial}{\partial X_3} & \frac{\partial}{\partial X_2} \\ \frac{\partial}{\partial X_3} & 0 & \frac{\partial}{\partial X_1} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \mathbf{P}\mathbf{u}$$



非线性几何方程

$$\mathbf{E} = \frac{1}{2}(\nabla \mathbf{u} + \nabla \mathbf{u}^T + \nabla \mathbf{u}^T \nabla \mathbf{u})$$

$$\hat{\mathbf{E}} = \begin{bmatrix} \frac{\partial}{\partial X_1} & 0 & 0 \\ 0 & \frac{\partial}{\partial X_2} & 0 \\ 0 & 0 & \frac{\partial}{\partial X_3} \\ \frac{\partial}{\partial X_2} & \frac{\partial}{\partial X_1} & 0 \\ 0 & \frac{\partial}{\partial X_3} & \frac{\partial}{\partial X_2} \\ \frac{\partial}{\partial X_3} & 0 & \frac{\partial}{\partial X_1} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} + \begin{bmatrix} \frac{1}{2} \sum_{k=1}^3 \left(\frac{\partial u_k}{\partial X_1} \right)^2 \\ \frac{1}{2} \sum_{k=1}^3 \left(\frac{\partial u_k}{\partial X_2} \right)^2 \\ \frac{1}{2} \sum_{k=1}^3 \left(\frac{\partial u_k}{\partial X_3} \right)^2 \\ \sum_{k=1}^3 \left(\frac{\partial u_k}{\partial X_1} \right) \left(\frac{\partial u_k}{\partial X_2} \right) \\ \sum_{k=1}^3 \left(\frac{\partial u_k}{\partial X_2} \right) \left(\frac{\partial u_k}{\partial X_3} \right) \\ \sum_{k=1}^3 \left(\frac{\partial u_k}{\partial X_1} \right) \left(\frac{\partial u_k}{\partial X_3} \right) \end{bmatrix} = \mathbf{P}_X \mathbf{u} + \bar{\mathbf{P}}_X(\mathbf{u})$$



非线性几何方程

$$\mathbf{E} = \frac{1}{2}(\nabla \mathbf{u} + \nabla \mathbf{u}^T + \nabla \mathbf{u}^T \nabla \mathbf{u})$$

$$\hat{\mathbf{E}} = \begin{bmatrix} \frac{\partial}{\partial X_1} & 0 & 0 \\ 0 & \frac{\partial}{\partial X_2} & 0 \\ 0 & 0 & \frac{\partial}{\partial X_3} \\ \frac{\partial}{\partial X_2} & \frac{\partial}{\partial X_1} & 0 \\ 0 & \frac{\partial}{\partial X_3} & \frac{\partial}{\partial X_2} \\ \frac{\partial}{\partial X_3} & 0 & \frac{\partial}{\partial X_1} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} + \begin{bmatrix} \frac{1}{2} \sum_{k=1}^3 \left(\frac{\partial u_k}{\partial X_1} \right)^2 \\ \frac{1}{2} \sum_{k=1}^3 \left(\frac{\partial u_k}{\partial X_2} \right)^2 \\ \frac{1}{2} \sum_{k=1}^3 \left(\frac{\partial u_k}{\partial X_3} \right)^2 \\ \sum_{k=1}^3 \left(\frac{\partial u_k}{\partial X_1} \right) \left(\frac{\partial u_k}{\partial X_2} \right) \\ \sum_{k=1}^3 \left(\frac{\partial u_k}{\partial X_2} \right) \left(\frac{\partial u_k}{\partial X_3} \right) \\ \sum_{k=1}^3 \left(\frac{\partial u_k}{\partial X_1} \right) \left(\frac{\partial u_k}{\partial X_3} \right) \end{bmatrix} = \mathbf{P}_X \mathbf{u} + \boxed{\bar{\mathbf{P}}_X(\mathbf{u})}$$



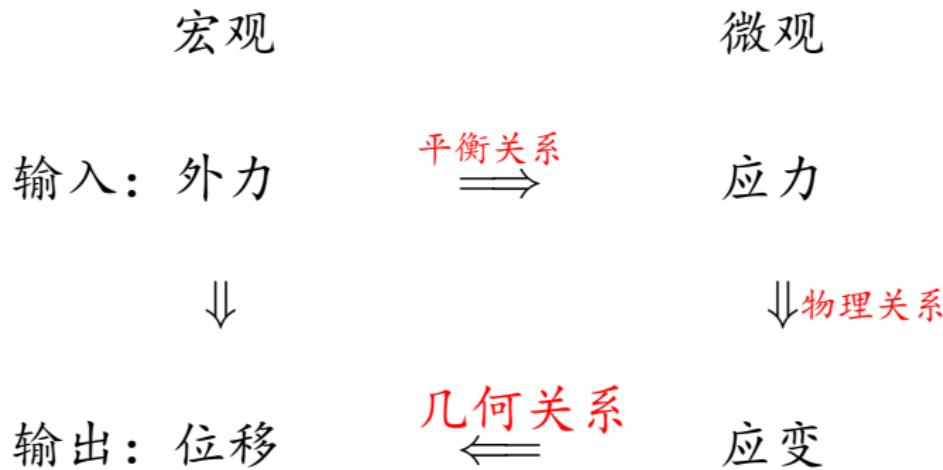
非线性几何方程

$$\mathbf{P}_X \mathbf{u} + \bar{\mathbf{P}}_X(\mathbf{u}) \approx \mathbf{P}_X \mathbf{F}^T \mathbf{u} = \mathbf{P}_x \mathbf{u}$$

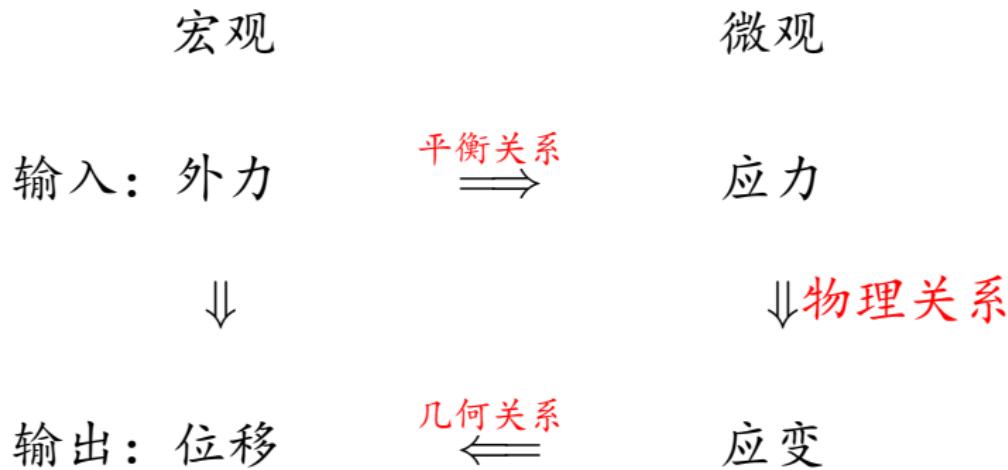
$$[F_{ij}] = \left[\frac{\partial x_i}{\partial X_j} \right] = \begin{bmatrix} \frac{\partial x_1}{\partial X_1} & \frac{\partial x_1}{\partial X_2} & \frac{\partial x_1}{\partial X_3} \\ \frac{\partial x_2}{\partial X_1} & \frac{\partial x_2}{\partial X_2} & \frac{\partial x_2}{\partial X_3} \\ \frac{\partial x_3}{\partial X_1} & \frac{\partial x_3}{\partial X_2} & \frac{\partial x_3}{\partial X_3} \end{bmatrix}$$



几何方程

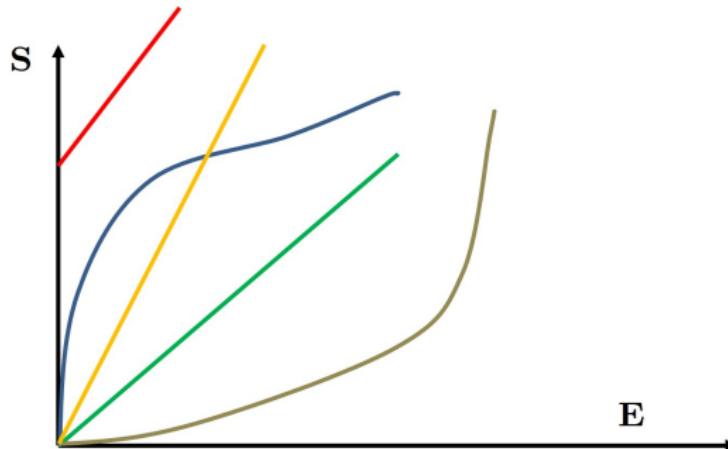


物理方程



物理方程

应力： \mathbf{S}



$$\mathbf{S} \propto \mathbf{E}$$



线性物理方程

各向同性

$$\sigma = \mathcal{C}\varepsilon$$

$$\sigma = \lambda(\text{tr}(\varepsilon))\mathbf{I} + 2\mu\varepsilon$$

$$\hat{\sigma} = \begin{bmatrix} \lambda + 2\mu & \lambda & \lambda & 0 & 0 & 0 \\ & \lambda + 2\mu & \lambda & 0 & 0 & 0 \\ & & \lambda + 2\mu & 0 & 0 & 0 \\ & & & \mu & 0 & 0 \\ & sym & & & \mu & 0 \\ & & & & & \mu \end{bmatrix} \hat{\varepsilon} = \mathcal{C}\hat{\varepsilon}$$



非线性物理方程

本构方程： $\mathbf{S} = \mathcal{C}(\mathbf{E})$

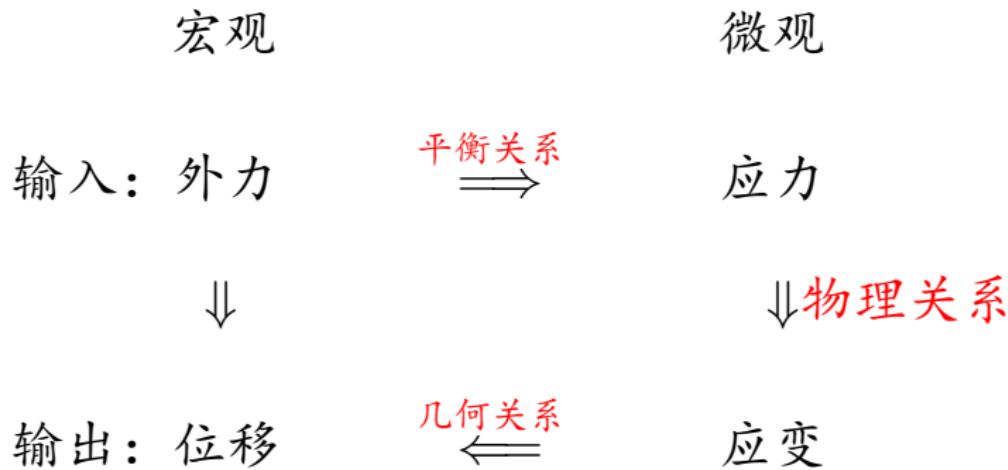
$$\mathbf{S} = \frac{\partial W(\mathbf{E})}{\partial \mathbf{E}}$$

Neo-Hookean

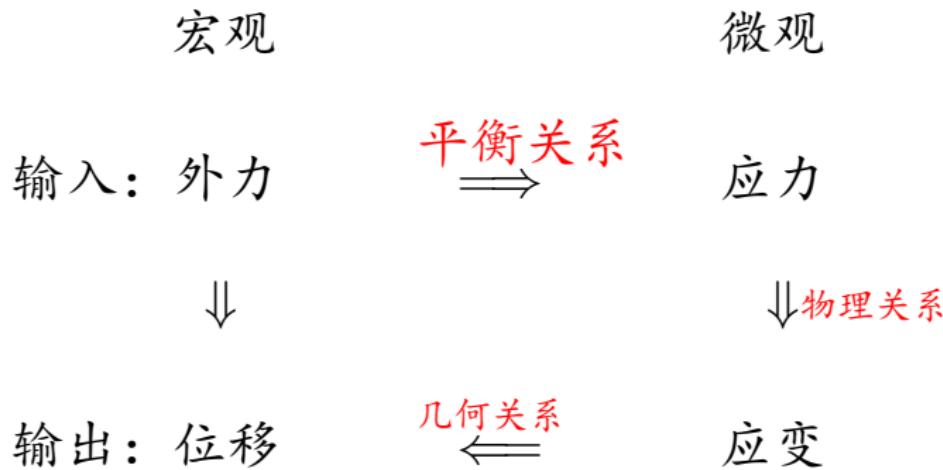
Mooney-Rivlin



物理方程



平衡方程



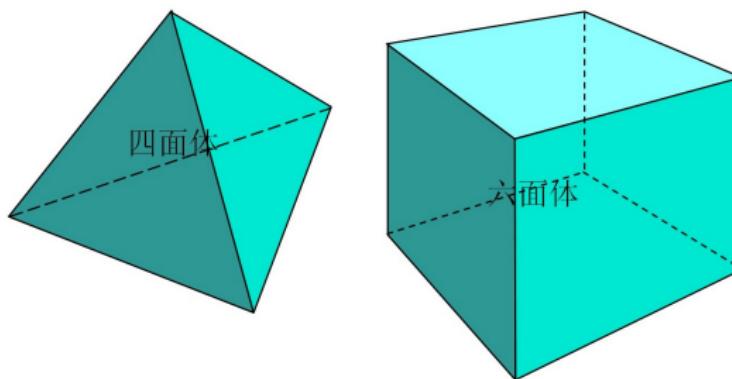
平衡方程

虚功方程：

$$\Pi = \underbrace{\int_{\Omega} \frac{1}{2} \mathbf{E}^T \mathbf{S} dV}_{\text{Energy}} - \underbrace{\int_{\Omega} \mathbf{u}^T \mathbf{f}_0 dV - \int_{\Gamma} \mathbf{u}^T \mathbf{t}_0 dA}_{\text{Boundary Conditions}}$$



离散单元



$$\hat{\mathbf{E}}_e = \mathbf{P}_X \mathbf{F}^T \mathbf{u} = \mathbf{P}_X \mathbf{F}^T \mathbf{N}_e \mathbf{u}_e = \mathbf{B} \mathbf{u}_e = \mathbf{B}(\mathbf{u}_{e_1} : \mathbf{u}_{e_2} : \mathbf{u}_{e_3} : \mathbf{u}_{e_4})^T$$



单元平衡方程

单元虚功方程：

$$\begin{aligned}\Pi_e &= \int_{\Omega_e} \frac{1}{2} (\mathbf{B} \mathbf{u}_e)^T \mathcal{C} (\mathbf{B} \mathbf{u}_e) dV_e - \int_{\Omega_e} (\mathbf{N}_e \mathbf{u}_e)^T \mathbf{f}_0 dV_e - \int_{\Gamma} (\mathbf{N}_e \mathbf{u}_e)^T \mathbf{t}_0 dA \\ &= \underbrace{\frac{1}{2} \mathbf{u}_e^T \left(\int_{\Omega_e} \mathbf{B}^T \mathcal{C} \mathbf{B} dV_e \right) \mathbf{u}_e}_{\mathbf{K}_e} - \underbrace{\mathbf{u}_e^T \left(\int_{\Omega_e} \mathbf{N}_e^T \mathbf{f}_0 dV_e + \int_{\Gamma} \mathbf{N}_e^T \mathbf{t}_0 dA \right)}_{\mathbf{r}_e}\end{aligned}$$



变分原理

$$\frac{\partial \Pi_e}{\partial \mathbf{u}_e} = \mathbf{K}_e \mathbf{u}_e - \mathbf{r}_e = \mathbf{0}$$



刚度矩阵的处理

$$\begin{aligned}\mathbf{Ku} &= \left(\sum_e \mathbf{K}_e \right) \mathbf{u} \\ &= \left(\sum_e \int_e \mathbf{B}^T \mathcal{C} \mathbf{B} dV_e \right) \mathbf{u} \\ &= \sum_e \int_e \mathbf{B}^T \mathcal{C} \mathbf{B} \mathbf{u}_e dV_e \\ &= \sum_e \int_e \mathbf{B}^T \mathcal{C} \hat{\mathbf{E}}_e dV_e \\ &= \sum_e \int_e \mathbf{B}^T \hat{\mathbf{S}} dV_e \\ &= \sum_e \mathbf{f}_e \\ &= \mathbf{f}(\mathbf{u})\end{aligned}$$



动力学平衡方程

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{D}\dot{\mathbf{u}} + \mathbf{K}(\mathbf{u})\mathbf{u} = \mathbf{r}$$



动力学平衡方程

$$M\ddot{u} + D\dot{u} + K(u)u = r$$



$$M\ddot{u} + D\dot{u} + f = r$$



动力学平衡方程

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{D}\dot{\mathbf{u}} + \mathbf{K}(\mathbf{u})\mathbf{u} = \mathbf{r}$$



$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{D}\dot{\mathbf{u}} + \mathbf{f} = \mathbf{r}$$



$$\mathbf{M}\ddot{\mathbf{u}}_n + \mathbf{D}\dot{\mathbf{u}}_n + \mathbf{f}_n = \mathbf{r}_n$$



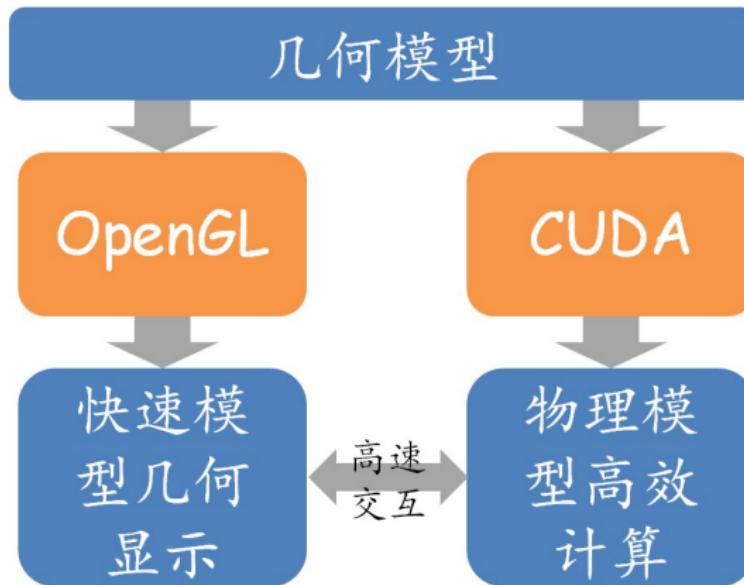
中心差分显式解法

$$^{t+\Delta t} \mathbf{u}_n = \gamma_1 \mathbf{M}_{n \times n}^{-1} ({}^t \mathbf{r}_n - {}^t \mathbf{f}_n) + \gamma_2 {}^t \mathbf{u}_n + \gamma_3 {}^{t-\Delta t} \mathbf{u}_n$$

其中 $\gamma_1 = 2\Delta t^2 / (\alpha\Delta t + 2)$, $\gamma_2 = 4 / (\alpha\Delta t + 2)$, $\gamma_3 = 1 - \gamma_2$ 。



GPU



Kernel 1

模型被剖分成 N_e 个四面体单元，发起 $3 \times N_e$ 个线程

$$\begin{aligned}\mathbf{f}_e &= \int_e \mathbf{B}^T \hat{\mathbf{S}} dV_e \\ &= V_e \mathbf{B}^T \hat{\mathbf{S}}\end{aligned}$$



Kernel 1

模型被剖分成 N_e 个四面体单元，发起 $3 \times N_e$ 个线程

$$\begin{aligned}\mathbf{f}_e &= \int_e \mathbf{B}^T \hat{\mathbf{S}} dV_e \\ &= V_e \mathbf{B}^T \hat{\mathbf{S}}\end{aligned}$$

$$V_e \quad \mathbf{B} \quad \hat{\mathbf{S}}$$



Kernel 2

模型被剖分成 N_n 个节点，发起 $3 \times N_n$ 个线程

$$^{t+\Delta t} \mathbf{u}_n = \gamma_1 \mathbf{M}_{n \times n}^{-1} ({}^t \mathbf{r}_n - {}^t \mathbf{f}_n) + \gamma_2^t \mathbf{u}_n + \gamma_3^{t-\Delta t} \mathbf{u}_n$$



Kernel 2

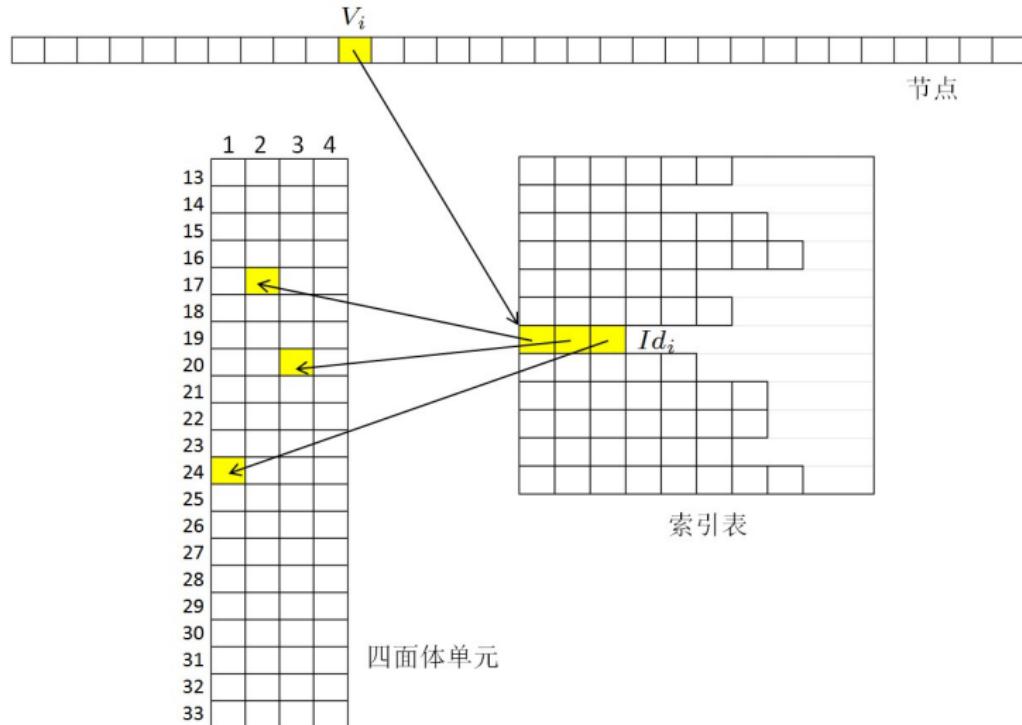
模型被剖分成 N_n 个节点，发起 $3 \times N_n$ 个线程

$$^{t+\Delta t} \mathbf{u}_n = \gamma_1 \mathbf{M}_{n \times n}^{-1} ({}^t \mathbf{r}_n - {}^t \mathbf{f}_n) + \gamma_2^t \mathbf{u}_n + \gamma_3^{t-\Delta t} \mathbf{u}_n$$

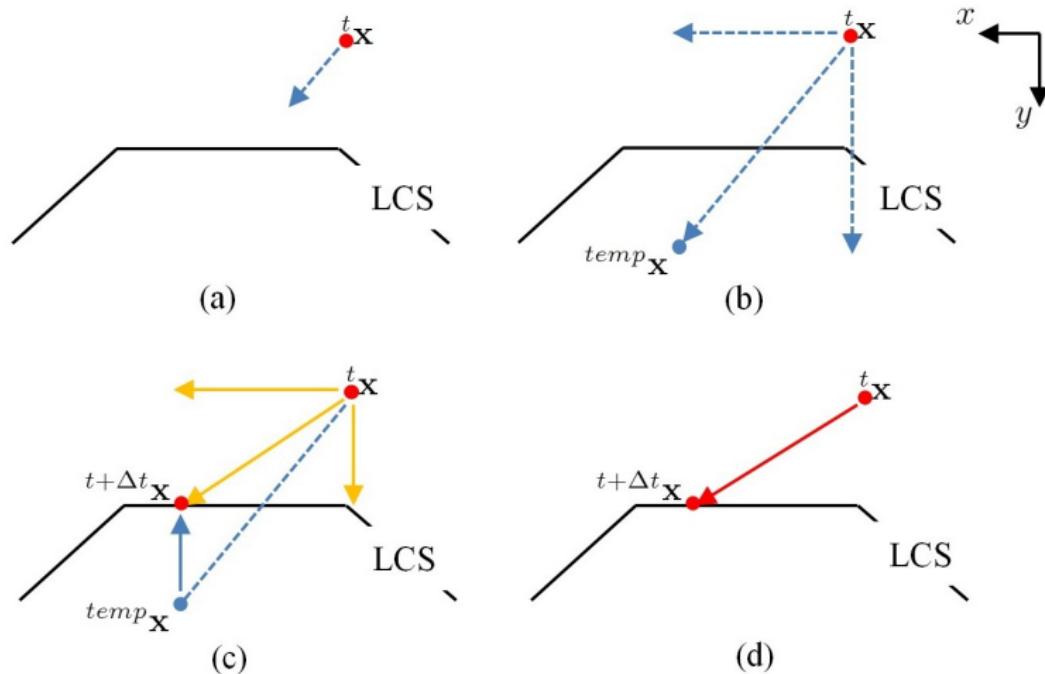
$$\mathbf{M}_{n \times n} \quad \mathbf{r}_n \quad \mathbf{f}_n$$



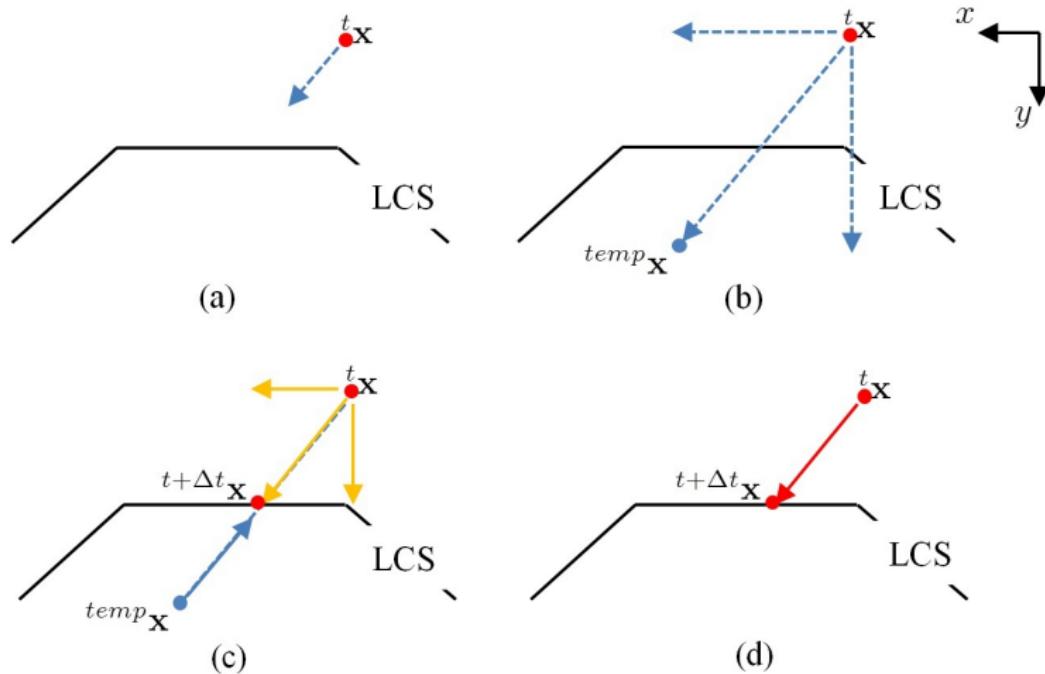
稀疏矩阵的合成



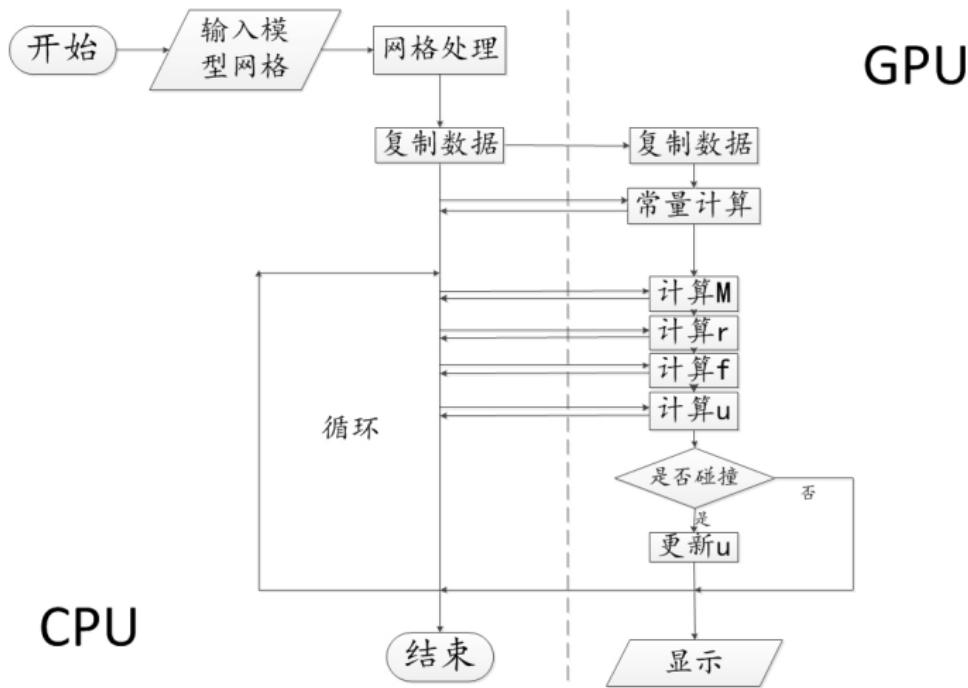
碰撞处理



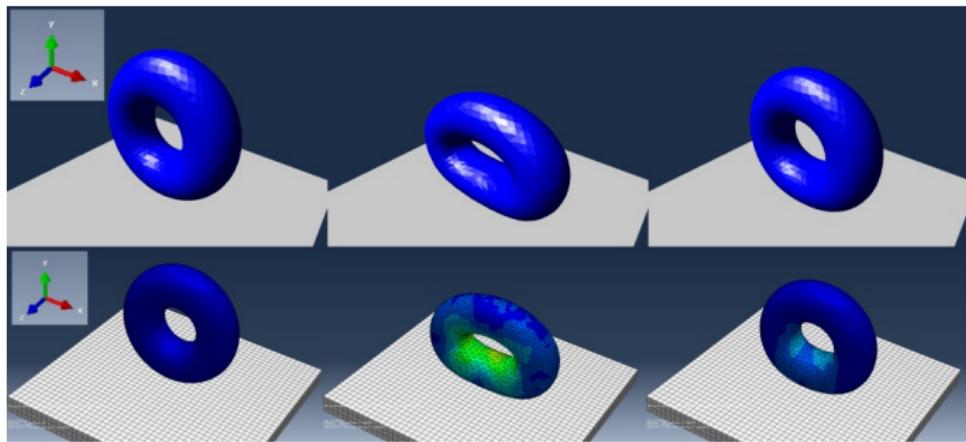
碰撞处理



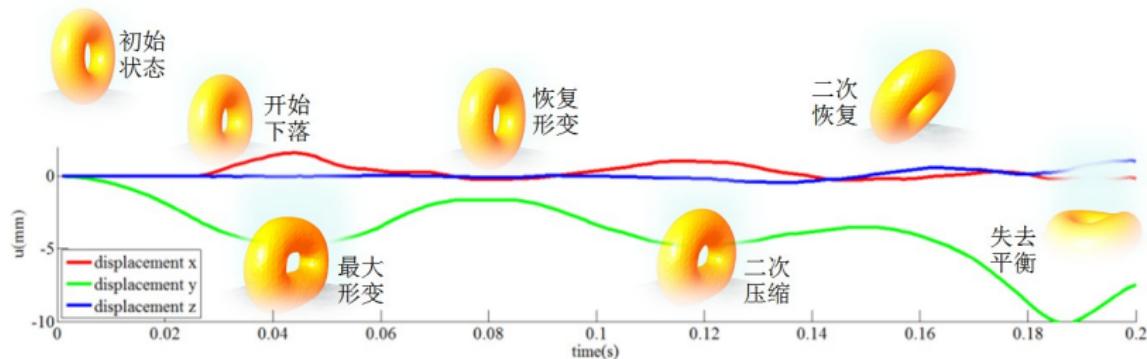
算法流程



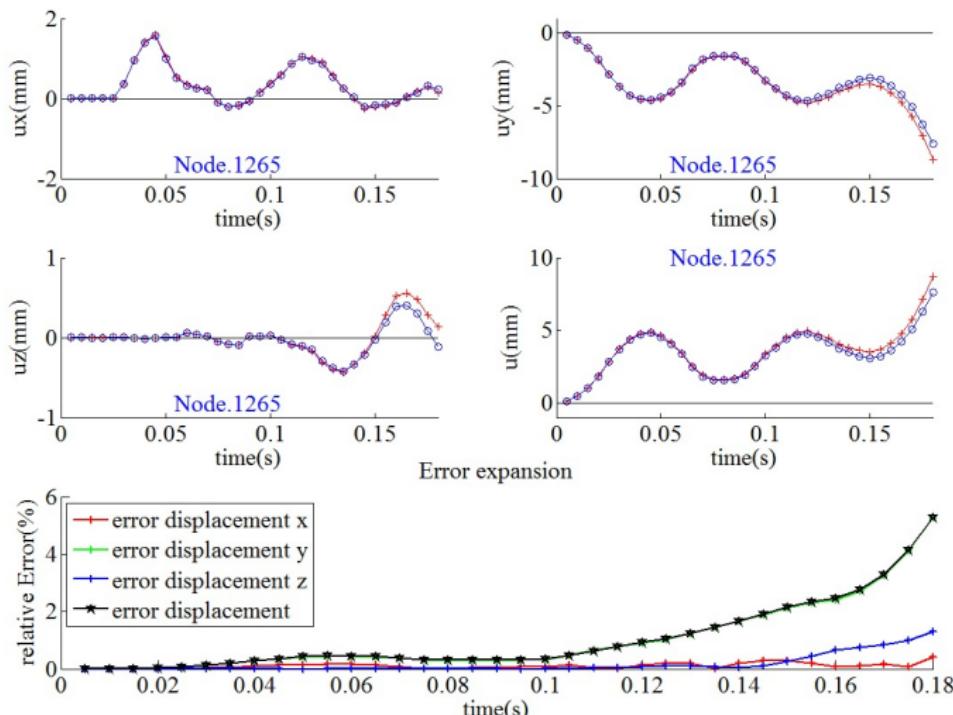
结果分析



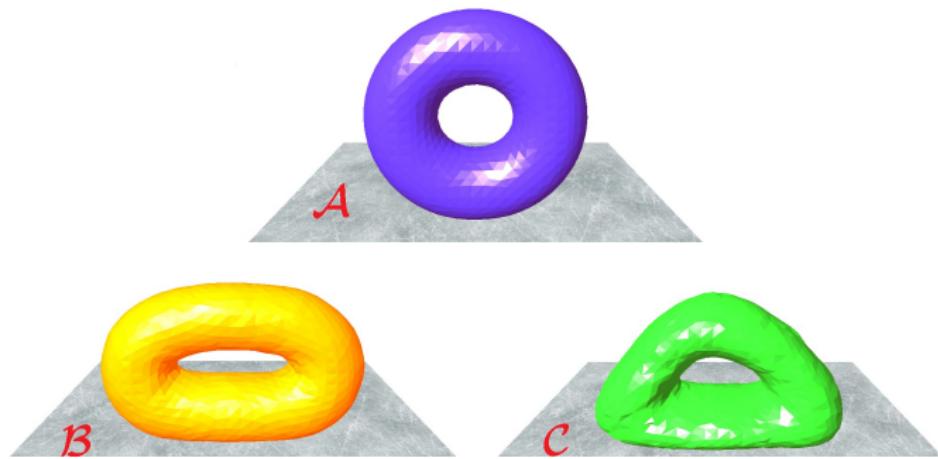
结果分析



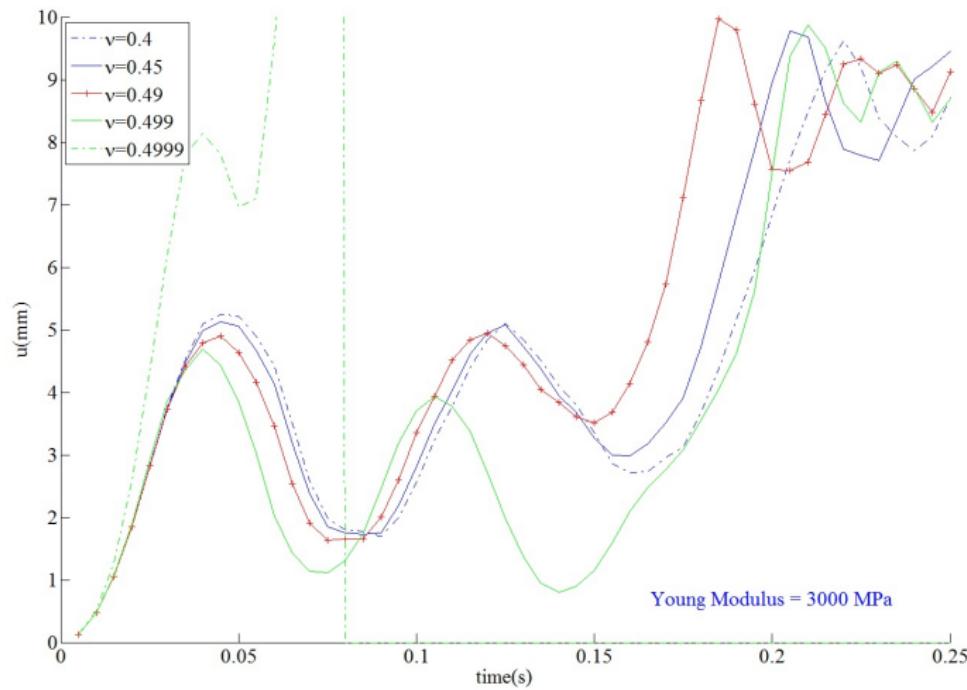
结果分析



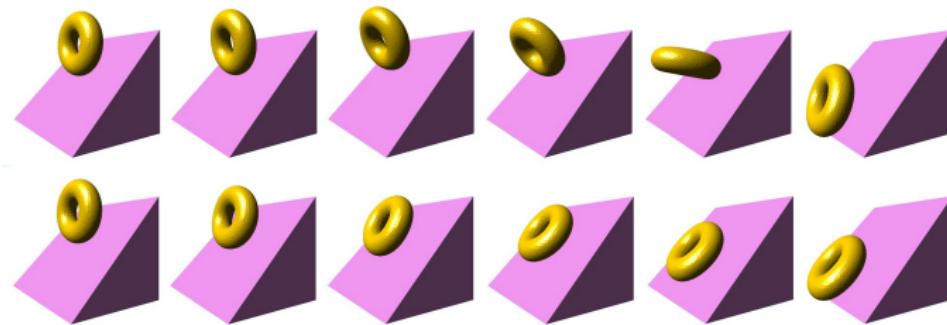
结果分析



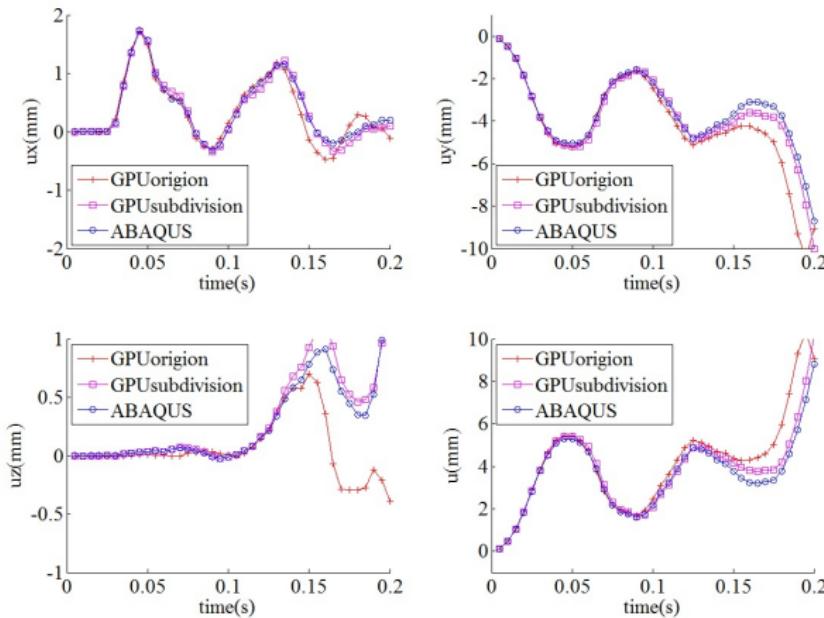
结果分析



结果分析



结果分析



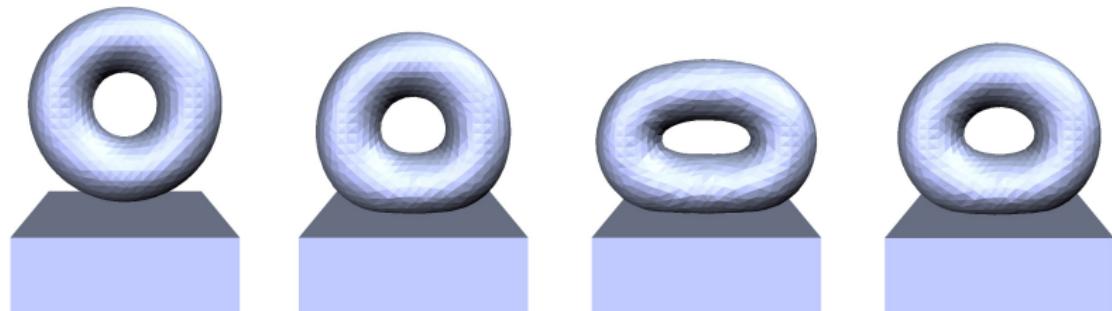
结果分析

8000次迭代，每次迭代步长0.025ms。

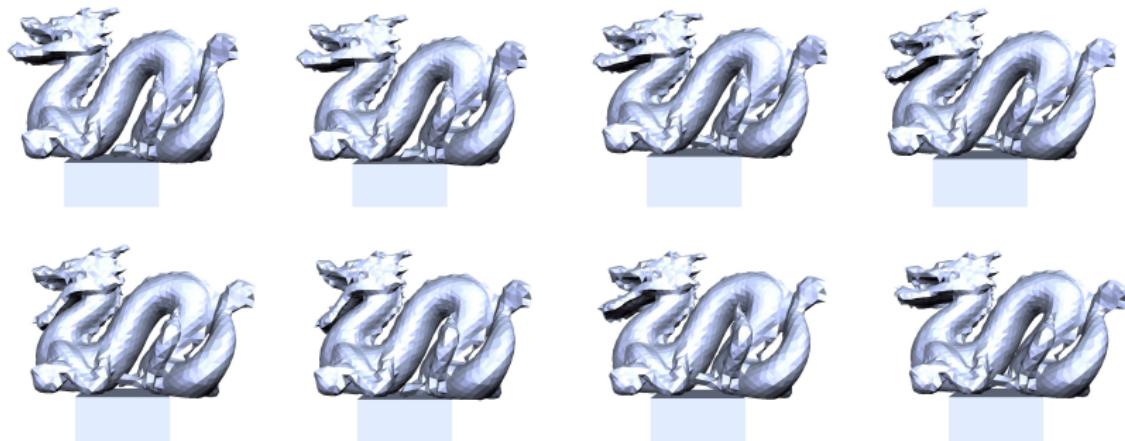
网格 数目	GPU 时间 (ms)					CPU 总时间(ms)
	求M	求f	求u	更新u	总时间	
7456	50.0	2280.0	54.8	500.0	2884.8	298288
12584	102.8	4024.0	101.2	980.0	5208.0	666624
27918	190.0	6604.0	160.4	1788.0	8742.4	1186344



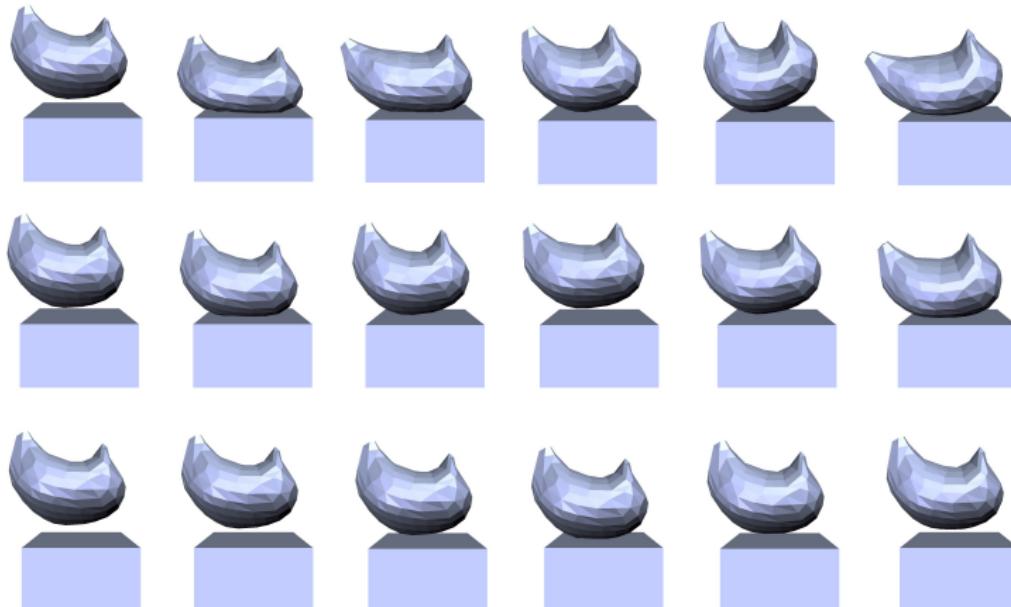
结果分析



结果分析



结果分析



工作总结

1. 采用合适的方法对模型网格进行简化、细分以及剖分，得到适合于本文有限元方法的体网格。
2. 针对有限元中的几何非线性情况，提出一种将其转化成线性函数的方法，以适应GPU加速。
3. 针对有限元中的物理非线性情况，提出对刚度矩阵的求解进行转换，以适应GPU加速。
4. 提出了一种快速碰撞检测方法，可以模拟不同的摩擦情况。
5. 把非线性有限元求解的过程用GPU并行化加速，实现更快的计算求解。



不足与展望

1. 没有和其他方法做比较，包括准确性和速度；之后可以考虑做真实的实验用于结果估计或者参数模拟。
2. 碰撞模型过于简单，不能很好的模拟真实情况；之后可以引入一些高级碰撞模型。
3. 没有达到预期的自适应细分，不能很好的将变形表面的细节展现出来；可以考虑用LOD的方式，或者displacement mapping等方法。



发表论文与毕业论文的关系

1. A master-slave robotic simulator based on GPUDirect.
2. Fall over or sliding down?
3. A Survey on Simulation of Soft Tissue Deformation in Virtual Surgery.
4. Surface Contact Deformation of Soft Body with Nonlinear Finite Element Method.



Thank you!

