其他数值方法

江见鲸 陆新征
清华大学土木工程系
2005

有限元法

- 最经典最成熟的方法
- 从工程角度的理解
  - 把实际结构划分成很多细小的单元
  - 用比较简单的数学模型近似单元内部的应力和变形
  - 将这些单元组合，通过求解刚度矩阵得到整体结构的反应

有限元法的优点

- 对于线弹性问题，当实际结构位移场函数连续光滑时，能够得到收敛解
- 对于任意复杂结构，理论上总是可以通过细分单元的方法获得足够近似的模拟
- 刚度矩阵系数带状，在结构不出现软化的时候还是对称正定的，求解方便
- 长期大量工程应用，积累了丰富的经验

有限元法的问题

- 基于连续介质力学
  - 如何处理界面？
  - 界面问题永远是结构分析中的关键问题
  - 存在界面就存在着跨尺度，不同尺度内问题有着完全不同的性态

界面问题

- 微观的界面
  - 混凝土骨料和砂浆
  - 钢筋和混凝土的界面
  - ……
- 宏观的界面
  - 结构的各种留缝
  - 钢结构的节点接触
  - 上部结构和下部地基
  - ……

当界面问题占主导时

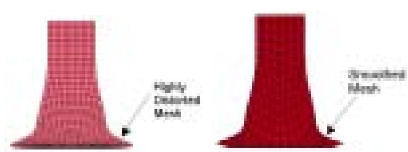
  - 岩石力学中岩石的相互作用
  - 结构倒塌过程中结构构件的相互碰撞
- 有限元分析遇到很多的问题
- 有限元的解决方法
  - 接触算法

有限元的问题

- 基于单元
  - 如果单元网格出现较大变化，如何处理
  - 大变形、裂缝
- 网格重划分
  - 工作量极大
  - 很多时候无法保证可以得到满意的单元网格

其他算法

- 欧拉网格
- 任意拉格朗日欧拉网格(ALE)



收敛性问题

- 传统的隐式分析有限元模型往往遇到各种收敛的难题

其他数值方法

- 显式有限元方法
- 离散单元法
- 刚体弹簧元法
- 无网格法
- 并行计算及其他

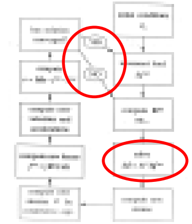
常用的隐式有限元程序

$$\boxed{K(u)}\Delta u = F_{\text{ext}} - \boxed{F_{\text{int}}(u)} - M\ddot{u}$$

刚度矩阵求逆

需要迭代求解

隐式有限元法



显式有限元分析

1. 对时间中心差分

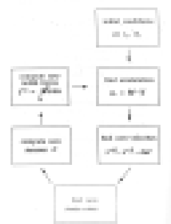
$$\ddot{u}(t) = M^{-1}[P(t) - F_{\text{int}}(t)] \qquad \boxed{K(u)\Delta u = F_{\text{ext}} - F_{\text{int}}(u) - M\ddot{u}}$$

$$\dot{u}(t+\frac{\Delta t}{2}) = \dot{u}(t-\frac{\Delta t}{2}) + \ddot{u}(t)\cdot\Delta t$$

$$u(t+\Delta t) = u(t) + \dot{u}(t+\frac{\Delta t}{2})\cdot\Delta t$$

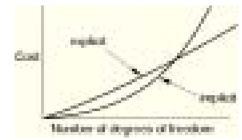
2. 计算单元应变增量
3. 计算内力
4. 回到1

显式有限元法



特色

- 无需对刚度矩阵求逆，只需对质量矩阵求逆，而质量矩阵往往可以简化为对角阵
- 没有增量步内迭代收敛问题，可以一直计算下去



优势

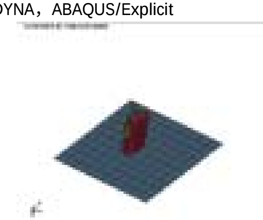
- 不需要求解刚度矩阵
  - 节省存储空间
  - 节省计算时间
- 算法简单，可以模拟各种复杂的非线性现象和本构关系
- 在加载时步足够小的情况下是稳定的
- 易于实现并行算法
- Toublefree, if it doesn’t go unstable, it will run
- 问题：对误差等的估计存在困难

隐式显式计算的优点和缺点

- 隐式计算
  - 时间步长增量较大
  - 每个荷载步都能控制收敛，避免误差累积
  - 存在迭代不收敛的问题
  - 计算量随计算规模增大而成超线性增长
- 显式计算
  - 时间步长很小
  - 误差累积
  - 不存在迭代不收敛的问题
  - 计算量随计算规模基本为线性增长

代表性软件

- DYNA，ABAQUS/Explicit



离散单元法

- 离散单元法也被称为散体单元法，最早是1971年由Cundall提出的一种不连续数值方法模型，这种方法的优点是适用于模拟节理系统或离散颗粒组合体在准静态或动态条件下的变形过程。
- 不是建立在最小势能变分原理上，而是建立在最基本的牛顿第二运动定律上。它以每个刚体的运动方程为基础，建立描述整个破坏过程的显式方程组后，通过动力松弛迭代求解

离散单元法的特点

- 岩体或颗粒组合体被模拟成通过角或边的相互接触而产生相互作用
- 块体之间边界的相互作用可以体现其不连续性和节理的特性
- 使用显式积分迭代算法，允许有大的位移、转动和使用

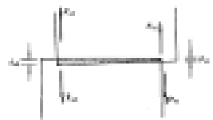
基本过程

- 首先将分析对象按照其物理界面划分成若干块非连续的刚体，刚体之间相互镶嵌排列，在空间有其固定的位置，处于平衡状态。
- 如果外力或边界条件发生变化时，某些块体在重力或者外力的作用下将产生一定的加速度和相应的位移，使块体的空间状态发生变化。
- 位移后块体与块体之间会接触而产生“叠合”，也可能会脱离接触。这时，需要根据力一位移关系，计算出块体间新的作用力状态。。

基本过程

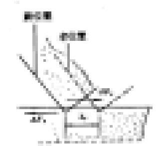
- 根据各个块体受到的作用力，根据牛顿运动定律，计算出块体当前的速度、加速度等运动参数。
- 对块体的运动参数进行积分，得到块体新的位置状态，接着再判断块体间的相互作用力。
- 重复上述步骤，迭代或逐步积分，得到所有块体运动变化的全过程

法向力一位移关系



$$F_n = K_n \cdot \delta_n$$

切向力一位移关系



$$|F_s| \le C + F_n \tan\varphi$$

$$K_n = \frac{B \cdot E \cdot A}{2\Delta l}$$

$$K_s = \frac{K_n}{2(1+\nu)}$$

运动方程

$$\frac{\partial^2 u}{\partial t^2} = \frac{F}{m}$$

$$\frac{\partial^2 u}{\partial t^2} = \frac{\dot{u}(t+\Delta t/2) - \dot{u}(t-\Delta t/2)}{\Delta t}$$

$$\dot{u}(t+\Delta t/2) = \dot{u}(t-\Delta t/2) + \frac{F(t)}{m}\Delta t$$

$$u(t+\Delta t) = u(t) + \dot{u}(t+\Delta t/2)\Delta t$$

存在多个力作用

$$\dot{u}(t+\Delta t/2) = \dot{u}(t-\Delta t/2) + \frac{\sum F(t)}{m}\Delta t$$

$$\dot{\theta}(t+\Delta t/2) = \dot{\theta}(t-\Delta t/2) + \frac{\sum M(t)}{I}\Delta t$$

$$u(t+\Delta t) = u(t) + \dot{u}(t+\Delta t/2)\Delta t$$

$$\theta(t+\Delta t) = \theta(t) + \dot{\theta}(t+\Delta t/2)\Delta t$$

阻尼

$$\frac{\dot{u}(t+\Delta t/2) - \dot{u}(t-\Delta t/2)}{\Delta t} = \frac{F(t)}{m} - \alpha \frac{\dot{u}(t+\Delta t/2) + \dot{u}(t-\Delta t/2)}{2}$$

$$D_s = \Delta u_s K_s' = \Delta u_s \beta K_s$$

$$D_n = \Delta u_n K_n' = \Delta u_n \beta K_n$$

迭代时间步长

$$u^{t+\Delta t}+[(k/m)(\Delta t)^2-2]u^t+u^{t-\Delta t}=0$$

$$\Delta t\le 2\sqrt{m/k}$$

清华大学研究生课程——《钢筋混凝土有限元》

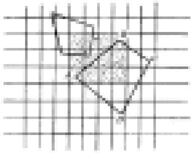
接触判断算法

- 离散元通过块体之间的相互接触判断得到相互之间的作用力，进而形成运动方程。因此，快速而准确的接触算法对离散元方法非常重要。
- 由于离散元计算过程中块体往往会发生较大位移，使得原有的块体间的空间拓扑关系发生变化，使接触判断变得更加复杂。
- 目前离散元对二维问题的接触分析已经比较成熟，但对于三维问题则应用比较有限，其中的重要原因就是三维接触判断过于复杂，特别是允许出现大位移的三维接触，目前还是一个有待进一步研究的问题。

清华大学研究生课程——《钢筋混凝土有限元》

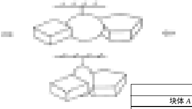
分格接触检索算法

- 将分析区域划分成相互重叠的若干分区
- 在每个分区内检索各个角点和边的关系
- 形成接触对



清华大学研究生课程——《钢筋混凝土有限元》

公共面法



与公共面接触的角点数		接触类型
块体 A	块体 B	
0	0	—
1	1	角—角
1	2	角—边
1	>2	角—面
2	1	边—角
2	2	边—边
2	>2	边—面
>2	1	面—角
>2	2	面—边
>2	>2	面—面

清华大学研究生课程——《钢筋混凝土有限元》

对三维接触的简化

- 从以上分析可以看出，三维离散元的接触判断和接触作用算法是非常复杂的。因此，考虑大位移的三维空间离散元分析，尤其是接触分析，还是当前研究的一个热点问题。
- 另外，一些离散元程序，将块体运动限制在有限位移范围内，这样在计算过程中，块体的拓扑关系不会发生大的变化，进而使得接触的判断和处理都得到简化。
- 圆形/球体接触算法

清华大学研究生课程——《钢筋混凝土有限元》

颗粒离散元法

- 由于多边形块体组成的离散元体系接触判断过于复杂，因此，很多研究者想到使用圆形（二维）或球形（三维）块体来代替多边形，从而使接触判断大大简化。
- 当球形块体足够多时，也可以较好地近似模拟实际块体的行为。
- 这种离散元体系被称为颗粒离散元、颗粒散体元或扩展离散元。

清华大学研究生课程——《钢筋混凝土有限元》

基本思路

- 将体系离散为由一列球形或圆形块体组成体系；
- 块体运动服从牛顿运动定律；
- 体系的变形和内力由块体间相互作用控制；
- 初始状态块体件可以通过链杆或弹簧单元相互联系，随着荷载的增加，块体间的连接单元破坏，块体之间主要靠接触传递荷载。

清华大学研究生课程——《钢筋混凝土有限元》

质点—桁架模型

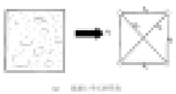
- 轴向变形等效

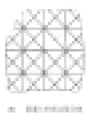
$$k_1=\frac{1}{2(1+\nu)}Et$$

$$k_2=\frac{\nu}{1-\nu^2}Et$$

- 切向变形等效

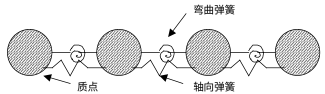
$$k_2=\frac{1}{2(1+\nu)}Et$$



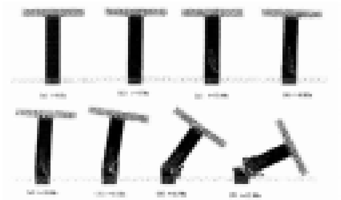


清华大学研究生课程——《钢筋混凝土有限元》

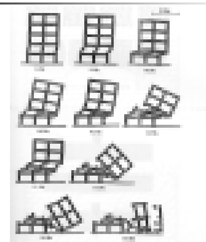
由颗粒单元组成的杆件单元



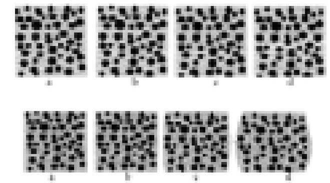
应用举例



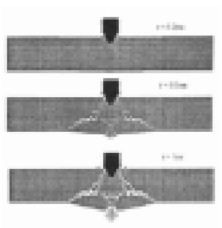
应用举例



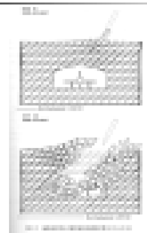
应用举例



应用举例



应用举例



刚体一弹簧元法

- 刚体弹簧元法(Rigid Body Spring Method, RBSM)最早由Kawai于1976年提出，当初提出的意图是以较少的自由度来求解结构问题。
- 它把体系分解为一些由均布在接触面上的弹簧系统联系起来的刚性元，刚性元本身不发生弹性变形，因此结构的变形能仅能储存在接触面的弹簧系统中。

刚体一弹簧元法

- 由于刚体弹簧元中任意一点的位移完全由单元型心的刚体位移来描述，因此单元刚度矩阵决不会超过6×6（平面单元为3×3），而且总刚的半带宽及体积也比传统的有限元要小。
- 由于刚体弹簧元单元间的作用力通过单元界面上弹簧传递，可以直接得到界面的作用力，因此在极限分析等领域也有着较好的应用。

基本原理

$$\begin{Bmatrix} u \\ v \end{Bmatrix} = \begin{bmatrix} 1 & 0 & -(y-y_G) \\ 0 & 1 & x-x_G \end{bmatrix} \begin{Bmatrix} u_G \\ v_G \\ \theta_G \end{Bmatrix}$$

$$\{\mathbf{u}\} = [\mathbf{Q}]\{\mathbf{U}_G\}$$

$$\{\mathbf{u}\} = \{u_1 \quad v_1 \quad u_2 \quad v_2\}^T$$

$$\{\mathbf{Q}\} = \begin{bmatrix} 1 & 0 & -(y-y_{G1}) & & & \\ 0 & 1 & x-x_{G1} & & & \\ & & & 1 & 0 & -(y-y_{G2}) \\ & & & 0 & 1 & x-x_{G2} \end{bmatrix}$$

$$\{\mathbf{u}_G\} = \{u_{G1} \quad v_{G1} \quad \theta_{G1} \quad u_{G2} \quad v_{G2} \quad \theta_{G2}\}^T$$

坐标变换

$$\{\boldsymbol{\delta}\} = \begin{Bmatrix} \delta_n \\ \delta_s \end{Bmatrix} = [\mathbf{M}][\mathbf{R}][\mathbf{Q}]\{\mathbf{U}_G\} = [\mathbf{B}]\{\mathbf{u}_G\}$$

$$[\mathbf{M}] = \begin{bmatrix} -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix}$$

$$[\mathbf{R}] = \frac{1}{l_{AC}} \begin{bmatrix} y_A-y_C & x_C-x_A & & \\ x_A-x_C & y_C-y_A & & \\ & & y_A-y_C & x_C-x_A \\ & & x_A-x_C & y_C-y_A \end{bmatrix}$$

$$\{\boldsymbol{\sigma}\} = \begin{Bmatrix} \sigma_n \\ \tau \end{Bmatrix} = \begin{bmatrix} k_n & 0 \\ 0 & k_s \end{bmatrix} \begin{Bmatrix} \delta_n \\ \delta_s \end{Bmatrix} = [\mathbf{D}]\{\boldsymbol{\delta}\}$$

弹簧参数

$$\begin{Bmatrix}\varepsilon_n\\ \gamma\end{Bmatrix}=\frac{1}{h_1+h_2}\begin{Bmatrix}\delta_n\\ \delta_s\end{Bmatrix}$$

$$\begin{cases}\sigma_n=E_1\varepsilon_n=\dfrac{(1-\nu)E}{(1-2\nu)(1+\nu)}\varepsilon_n\\ \tau=E_2\gamma=\dfrac{E}{1+\nu}\gamma\end{cases}\qquad\begin{cases}k_n=\dfrac{(1-\nu)E}{(1-2\nu)(1+\nu)}\dfrac{1}{h_1+h_2}\\ k_s=\dfrac{E}{1+\nu}\dfrac{1}{h_1+h_2}\end{cases}$$

$$\begin{cases}\sigma_n=E_1\varepsilon_n=\dfrac{E}{1-\nu^2}\varepsilon_n\\ \tau=E_2\gamma=\dfrac{E}{1+\nu}\gamma\end{cases}\qquad\begin{cases}k_n=\dfrac{E}{1-\nu^2}\dfrac{1}{h_1+h_2}\\ k_s=\dfrac{E}{1+\nu}\dfrac{1}{h_1+h_2}\end{cases}$$

应力应变关系

$$\{\boldsymbol{\delta}\}=\begin{Bmatrix}\delta_n\\ \delta_s\end{Bmatrix}=[\mathbf{M}][\mathbf{R}][\mathbf{Q}]\{\mathbf{U}_G\}=[\mathbf{B}]\{\mathbf{u}_G\}$$

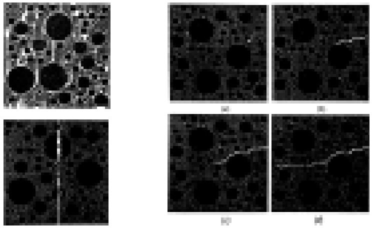
$$\{\mathbf{F}_G\}=\{X_{G1}\quad Y_{G1}\quad M_{G1}\quad X_{G2}\quad Y_{G2}\quad M_{G2}\}^T$$

$$\{\delta\mathbf{u}_G\}=\{\delta u_{G1}\quad \delta v_{G1}\quad \delta\theta_{G1}\quad \delta u_{G2}\quad \delta v_{G2}\quad \delta\theta_{G2}\}^T$$

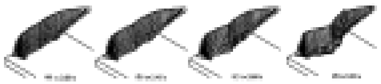
$$\{\mathbf{F}_G\}=\int[\mathbf{B}]^T[\mathbf{D}][\mathbf{B}]\mathrm{d}s\{\mathbf{u}_G\}$$

$$[\mathbf{K}]=\int[\mathbf{B}]^T[\mathbf{D}][\mathbf{B}]\mathrm{d}s$$

应用举例



应用举例



评述

- 离散单元、刚体弹簧元、有限元三者在定义和应用上很多时候存在矛盾或者重复
- 可以理解：离散单元在块体自身变形较小而块体相对作用较大时适用；刚体弹簧元适用于界面变形不大而对界面强度计算有要求的情况；
- 有限元法最为通用，带界面的显式有限元模型可以部分或者完全代替上述计算方法

无网格法

- 传统有限元需要构造特定的单元网格来形成位置插值函数
- 是否可以让计算机根据节点信息来“自动”形成位移插值函数？
- 对函数的要求：
  - 光滑连续
  - 影响的节点有限

无网格法的主要类型

- 插值函数
  - 移动最小二乘
  - 核函数
  - 径向基函数
- 整体方程
  - 配点法
  - 最小二乘法
  - 伽辽金法

无网格伽辽金法（EFGM）

- Element Free Galerkin Method
- 利用移动最小二乘法形成插值函数
- 利用伽辽金法形成整体方程
- 应用最广、最稳定的无网格法之一

无网格伽辽金方法

1、用移动最小二乘法构造某点附近的场函数的近似函数

$$u^*(x)\approx u^h(x)=\sum_{j=1}^{m}p_j(x)a_j(x)\equiv\mathbf{p}^T(x)\mathbf{a}(x)$$

EFGM—基函数

- 这里P（x）是m维完全多项式，在二维可取：

$$\mathbf{p}^T(x)=[1 \quad x \quad y] \quad \text{(m=3)} \quad \text{线性基} \qquad (2a)$$

$$\mathbf{p}^T(x)=[1 \quad x \quad y \quad x^2 \quad xy \quad y^2] \quad \text{(m=6)} \quad \text{二次基} \qquad (2b)$$

$$\mathbf{p}^T(x)=[1 \quad x \quad y \quad x^2 \quad xy \quad y^2 \quad x^3 \quad x^2y \quad xy^2 \quad y^3] \quad \text{(m=3)} \quad \text{三次基} \qquad (2c)$$

EFGM—局部近似场

式（1）在点附近对应的局部近似为

$$u^h(x,x^*)=\sum_{j=1}^{m} p_j(x^*)a_j(x) \equiv \mathbf{p}^T(x^*)\mathbf{a}(x)$$

EFGM—误差加权范数

系数a(x)根据加权最小二乘来确定，它使得近似函数和原函数在各已知点上取值差别的加权平方范数J最小。

$$J=\sum_{j=1}^{n} w(\|x-x_j\|)[u^h(x,x_j)-u^*(x_j)]^2=\sum_{j=1}^{n} w(\|x-x_j\|)\left[\sum_{i=1}^{m} p_i(x_j)a_i(x)-u^*(x_j)\right]^2$$

EFGM—移动最小二乘法

$$\frac{\partial J}{\partial a_i}=0$$

$$\mathbf{a}(x)=A^{-1}(x)B(x)\mathbf{u}^* \qquad (5)$$

这里

$$A(x)=\sum_{i=1}^{n} w_i\left(\|x-x_j\|\right)p(x_i)\,p^T(x_i) \qquad (5a)$$

$$B(x)=[w_i\left(\|x-x_j\|\right)p(x_1),\ldots\ldots,\ w_n\left(\|x-x_j\|\right)p(x_n)] \qquad (5b)$$

$$\mathbf{u}^*=[u^*(x_1),u^*(x_2),\ldots,u^*(x_n)]^T \qquad (5c)$$

EFGM—形函数及其导数

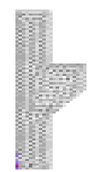
$$u^h(x)=\sum_{i=1}^{n} n_i(x)u_i^*$$

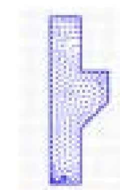
$$n_i(x)=\sum_{j=1}^{m} p_j(x)[A^{-1}(x)B(x)]_{ji}$$

$$n_{i,k}(x)=\sum_{j=1}^{m}\{p_{j,k}(x)[A^{-1}(x)B(x)]_{ji}+p_j(x)[A_{,k}^{-1}(x)B(x)+A^{-1}(x)B_{,k}(x)]_{ji}\}$$

无网格方法有着“流形法”的特点

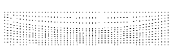
- 所谓“流形法”，就是物理覆盖和数学覆盖可以相互独立
- 可以在任意需要的位置布置节点，节点和积分点关系不固定

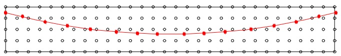




无网格方法有着“流形法”的特点

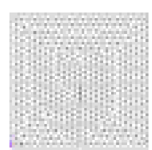
便于钢筋布置

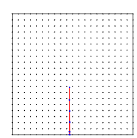




无网格方法有着“流形法”的特点

便于裂缝布置





有限元处理混凝土裂缝的方法

- 分离裂缝法
  - 直接得到单条裂缝的长度、宽度和形状
  - 可以很方便的分析裂面的咬合、接触等问题
  - 分离裂缝模型需要随着裂缝扩展不断修正网格划分，效率低、工作量大
  - 难以跟踪混凝土中大量的微裂缝

有限元处理混凝土裂缝的方法

- 弥散裂缝法
  - 比较简单，易于程序实现
  - 可以处理微裂缝
  - 无法得到单条裂缝的信息。
  - 剪力传递系数影响很大，凭经验确定，缺少试验支持

引入无网格法处理混凝土裂缝

- 与有限元本质相同，有限元的方法可以直接使用 ⟺ 用弥散裂缝模拟大量存在的微裂缝
- 有“流形法”的优点，可以方便修改网格和节点分布 ⟺ 用分离裂缝模型模拟宏观裂缝

影响域的设定和裂缝边界的处理

- 由于裂缝的开展，节点和高斯积分点之间的关系是不断变化的
- 使用“遮蔽”的方法处理裂缝边界

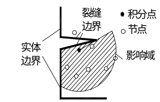


图 1 影响域和边界的处理

Fig.1 Setup the influence domain and its border

裂缝的分类

- 当混凝土的拉应变小于某一数值时，我们认为这时混凝土的裂缝基本上都是肉眼不可见的微裂缝，此时的混凝土仍然可以承受部分的拉应力。因此，对于这些混凝土，可以使用传统的弥散裂缝模型
- 当混凝土的拉应变大于 $\varepsilon_{t,u}$ 时，我们认为此时混凝土中已经存在肉眼可见的宏观裂缝，弥散裂缝模型已经不能很好的描述这些裂缝。因此，我们通过增加节点和边界的方法，对这些裂缝加以几何描述

宏观裂缝的生成

- 本次计算均假设裂缝首先从构件的边缘产生，对于普通钢筋混凝土梁等纯弯或弯剪组合受力构件，该假设一般都是成立的

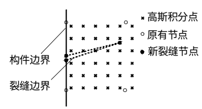


图 2 宏观裂缝的生成

Fig.2 Generation of macro-crack

宏观裂缝的开展

- 如果裂缝尖端节点最大拉应变大于极限应变，则允许裂缝按最大压应变的方向发展
- 添加两个新的裂缝节点

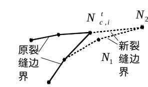


图 3 宏观裂缝开展

Fig.3 Expansion of maro-crack

宏观裂面抗剪单元

- 从分离式宏观裂缝模型可以得到裂缝宽度和裂缝相对剪切位移
- 目前已经有大量关于混凝土界面的试验结果
- 可以在宏观裂缝表面布置抗剪单元，描述骨料咬合作用

混凝土和钢筋的组合

- 由于无网格方法节点配置非常灵活，因而可以在任何配置钢筋的位置增加节点。本文使用传统混凝土有限元中分析中常采用的链杆单元来模拟钢筋，钢筋与混凝土共用节点

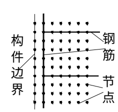


图 4 混凝土和钢筋组合

Fig.4 Combination of concrete and rebar

算例1：受弯破坏混凝土梁

- 素混凝土悬臂梁B1
- 无腹筋混凝土悬臂梁B2

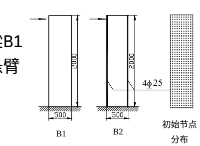
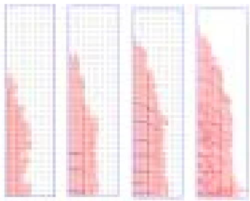


图 5 构件几何尺寸及初始节点分布

Fig.5 Dimesion and initial node distribution of specimens

算例1：受弯破坏混凝土梁



算例1：受弯破坏混凝土梁

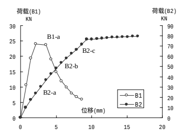


图 6 荷载位移曲线

Fig.6 Load-displacement curves

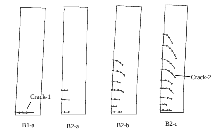


图 7 宏观裂缝发展

Fig.7 Expansion of macro-cracks

算例1：受弯破坏混凝土梁

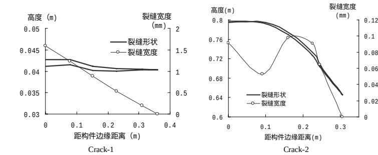
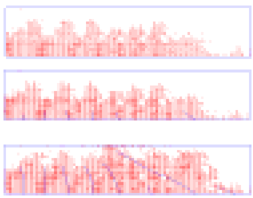


图 8 裂缝形状和宽度

Fig.8 Shape and width of cracks

算例2：斜拉破坏混凝土梁



算例2：斜拉破坏混凝土梁

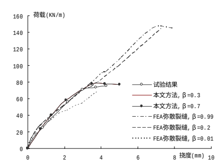


图 5 荷载位移曲线对比

Fig.5 Compare of Load-displacement Curve

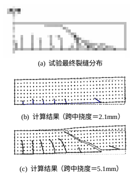
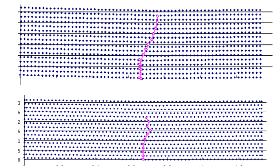


图 6 裂缝分布图

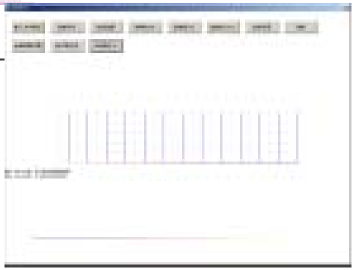
Fig.6 Development of Crack

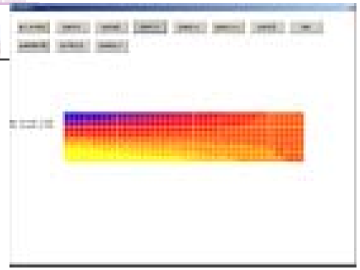
目前存在的问题

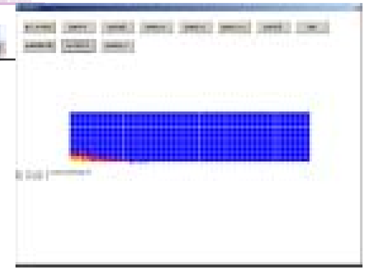
- 拓扑关系自动生成
- 网格不稳定性

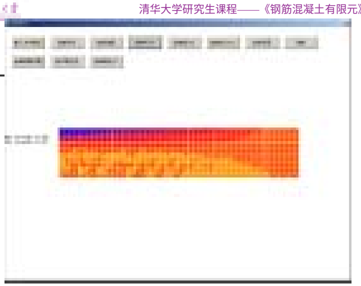


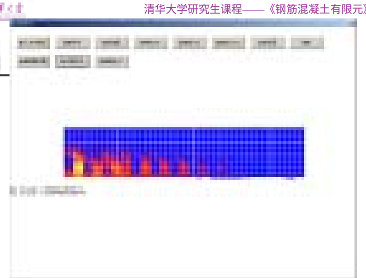
算例



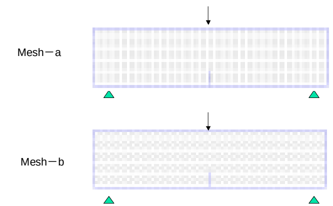




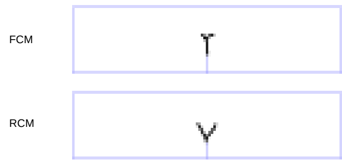




算例对比



Mesh-a



Mesh-b

对FCM和RCM的思考

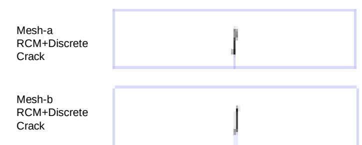
- 在微裂缝阶段，RCM是比较合理的，因为此时的损伤较小，主损伤方向发生一定的转动是合理的；
- 在裂缝发展较大的时候，前期的累计损伤效果不可忽视，主裂缝方向已经基本确定；
- 结论：根据裂缝发展阶段选用不同的裂缝模型从原理上说是合理的；

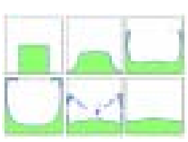
组合裂缝(RS-D-CM)模型

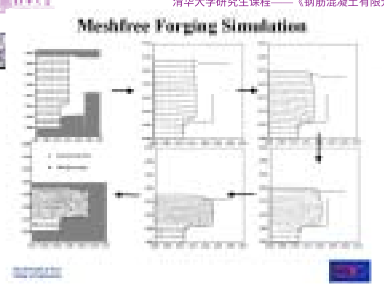
- Rotated Smeared-Discrete-Crack Model
- 基本思想：
- 微裂缝阶段使用RCM分布裂缝模型
- 宏观裂缝阶段使用分离裂缝(Discrete Crack)模型

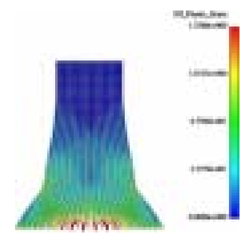
算例（以前的模型计算结果）

算例（RS-D-Crack Model）

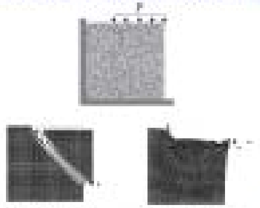




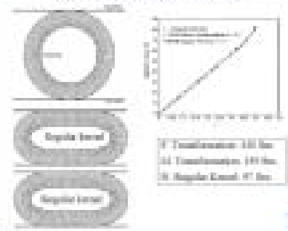




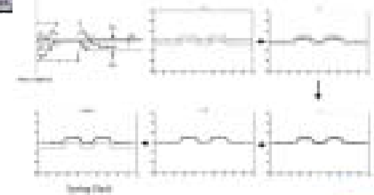
Shear Failure in Vertical Slope Earth



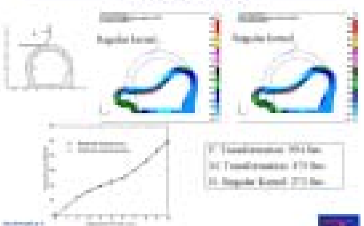
Rubber Ring Compression



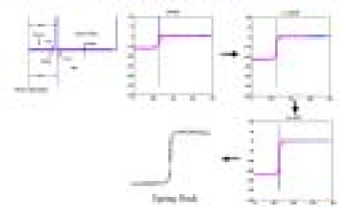
Metal Stamping Simulation



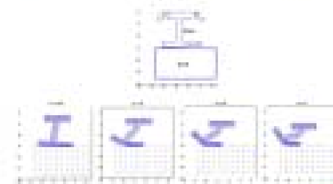
Door Seal Analysis



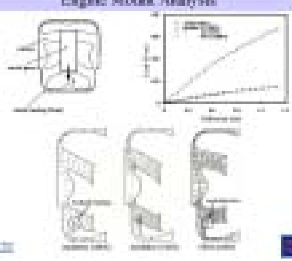
Deep Drawing Simulation



Rubber Blade-Block Contact



Engine Mount Analysis

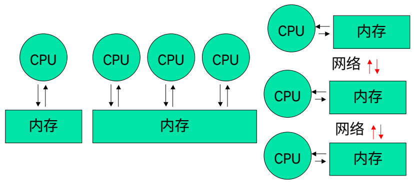


Adaptive Simulation of Slope Instability

并行计算

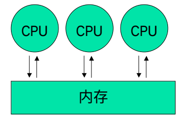
- 常用的并行计算环境
- SMP (Share Memory Parallel)
- MPI (Message Passing Interface)

计算机结构



SMP

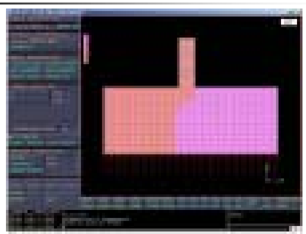
- 共享内存计算
- 例如，ANSYS里面使用如下命令
- /config,nproc,n



MPI

- 多计算机或单计算机多CPU，通过一定的通讯协议（MPI）互相交流数据
- 积分分析过程
  - 对结构分区
  - 分区求解结构反应
  - 利用网络交流计算结果

MARC并行计算示例



计算时间比较

- 用2个CPU 7701s
- 用1个CPU 13147s
- 速度提高了70%

MPI并行计算的最大优点

- 可以多个计算机同时计算一个问题
- 尤其适用于计算自由度过大，内存不足，需要用硬盘当缓存的情况
- 可以极大的提高计算速度，节省时间