

有限元理论基础及 **Abaqus** 内部实现方式研究系列 3:

S4 壳单元剪切自锁和沙漏控制

V2017-1017

作者: SnowWave02 From www.jishulink.com

email: snowwave02@qq.com

1 概述

本系列文章研究成熟的有限元理论基础及在商用有限元软件的实现方式。我们关注 CAE 中的结构有限元, 所以选择了商用结构有限元软件中文档相对较完备的 **Abaqus** 来研究内部实现方式。

以往的系列文章:

第一篇: S4 壳单元刚度矩阵研究

<http://www.jishulink.com/content/post/338859>

第二篇: S4 壳单元质量矩阵研究

<http://www.jishulink.com/content/post/343905>

本文为第三篇: S4 壳单元的剪切自锁和沙漏控制。介绍 **Abaqus** 的 S4 单元如何来消除剪切自锁以及 S4R 如何来抑制沙漏的。

1.1 研究方法

在实际应用过程中, 商用 CAE 软件在传统的理论上会做相应的修正以解决工程中遇到的不同问题, 且各家软件的修正方法都不一样, 商用软件对外就是一个黑盒子, 除了开发人员, 使用人员只能在黑盒子外猜测内部实现方式, 一方面我们查阅 **Abaqus** 软件手册得到修正方法的说明, 另一方面我们自己编程实现简单的结构有限元求解器, 通过自研求解器和 **Abaqus** 的结果比较结合理论手册如同管中窥豹一般来研究 **Abaqus** 的修正方法, 从而猜测商用有限元软件的内部计算方法。为了理解方便有很多问题在数学上其实并不严谨, 同时由于水平有限也可能有许多的理论错误, 欢迎交流讨论。

1.2 自编有限元求解器介绍

自编有限元求解器取名为 iSolver, 基于 Matlab 语言编写, 使用 **Abaqus/CAE** 做前后处理。

有如下特点:

(1) 作为 Abaqus/CAE 的一个插件，可以直接使用 Abaqus/CAE 做前后处理，调用方式和 Abaqus 自带求解器类似，容易上手。

(2) 软件整体框架灵活，除了内置的各种单元外，支持用户利用 Matlab 编写自定义单元，便于扩展。

(3) 和 Abaqus 结果一键式比较，包括结果、刚度矩阵、质量矩阵的细致比较。

具体介绍和下载请到下面网页：

<http://www.jishulink.com/content/post/337351>

2 第三篇：S4 壳单元剪切自锁和沙漏控制

商用有限元软件的健壮性体现在对各种特殊情况，求解过程和解的正确性依然能得到保证，而这些特殊情况在自编程序中如果没有考虑到，那么结果就可能相差极大。其中剪切自锁和沙漏现象是最常见的会影响正确性的两个特殊情况。这两者具有相似性，所以我们在本文中一起研究 Abaqus 中线性壳单元 S4 针对这两种情况下的内部实现方式。剪切自锁和沙漏现象影响的是刚度矩阵和应力，我们研究方式是在自编程序 iSolver 中根据成熟的消除剪切自锁和沙漏控制的理论实现刚度矩阵的修正，通过比较同一模型的 Abaqus 的刚度矩阵结果，结合帮助文档猜测 Abaqus 软件单元消除剪切自锁和控制沙漏的内部实现方法。

2.1 剪切自锁介绍

剪切自锁发生在完全积分单元中，这里的积分指的是求解刚度矩阵时采用的积分形式，且线性单元最明显，同时，不是所有的载荷都会发生剪切自锁的，只在受弯下才会发生自锁现象。剪切自锁时会导致单元刚度变大，收敛比较慢，就算对一个细网格，结果和真实值之间也可能有很大误差。这里我们只讨论完全积分的壳单元 S4。

2.1.1 剪切自锁例子

Example 1: 薄膜自锁例子

一个悬臂梁，左边固支约束，右端加力，此时梁将主要受弯曲作用，如果完全积分的线性单元没有考虑剪切自锁，那么将可能发生上面所说的现象。



参数如下：

尺寸：X 方向长度 $L=800$ ，Y 方向宽度 $b=20$ ，Z 方向厚度 $h=2$ 。

材料：Young's Modulus $E=11000$ 。

点载荷： $F=0.01$

理论值：悬臂梁的理论计算公式右端最大的 $U = (F \cdot L^3) / (3 \cdot E \cdot I_z) = 1.16e-1$ 。

其中 E 指的是杨氏模量（弹性系数）， L 指的是悬臂梁的长度， $I_z = hb^3/12$ 。

仿真值：在 **Abaqus** 中采用面网格模拟上述问题，平面位于 **XY** 平面，**Z** 方向为厚度方向，网格类型分别选为平面应力单元 **CPS4** 和线性壳单元 **S4**，这两个都是完全积分的线性单元。其中 **X** 方向和 **Y** 方向网格数目分别设置，如下表 $m \times n$ 的前一个数 m 表示 **Y** 方向网格数，后一个数 n 表示 **X** 方向网格数目。

分析结果如下：（相对值=有限元解/理论解）

单元	1X6 网格	2X12	4X12	8X24
CPS4	5.013e-3 相对值 0.043	1.776e-2 相对值 0.15	1.776e-2 相对值 0.15	4.874e-2 相对值 0.42
S4	1.028e-1 相对值 0.88	1.096e-1 相对值 0.94	1.077e-1 相对值 0.93	1.100e-1 相对值 0.95

可以发现 **CPS4** 由于剪切自锁，网格即使很小为 **8X24** 时，结果依然只有正确值的 42%，而 **S4** 很快收敛到了正确值，说明 **S4** 单元消除了剪切自锁。

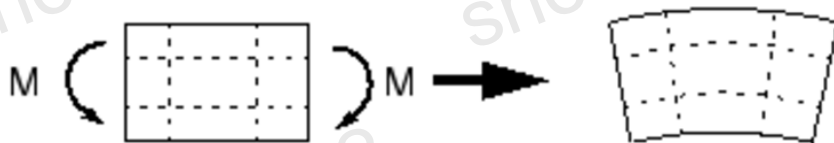
2.1.2 剪切自锁产生的原因

下面简单介绍一下剪切自锁产生的原因，具体可看 **Abaqus** 的帮助手册和有限元理论文章[1]。

壳单元可能受到两种形式的弯曲，分别为面内弯曲和面外弯曲。

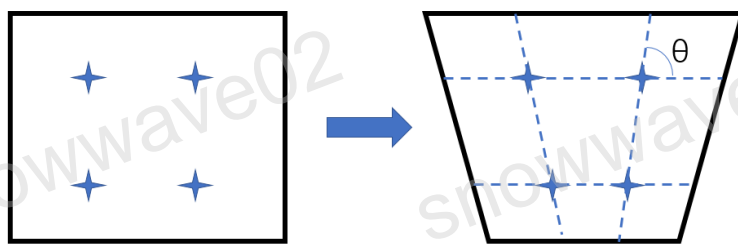
2.1.2.1 面内弯曲

单元在面内受弯时，正常形状如下图：



理论值：在纯弯矩作用下，单元内每个点的两个方向的夹角都维持 90 度不变，导致剪切应变理论上为 0。

仿真值：对线性壳单元，位移在整个单元内是线性的，在边界上的形状也为直线。也就是对线性壳单元受弯情况下变形为梯形，在有限元分析中，计算的值都是积分点上的结果，如果完全积分，那么为 2X2 的积分点，将通过积分点的连线画出来，由于线性单元，那么连线也是线性的，弯曲后的形状如下图。



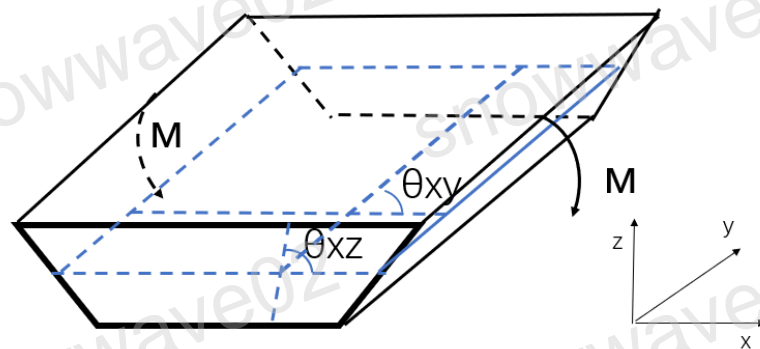
针对积分点上右上角点，横向和纵向连线夹角设为 θ ，由剪切应变的定义

$$\text{面内剪切应变} = G \cdot \Delta\theta = G \cdot (\theta - \pi/2)$$

上述 4 个点明显 θ 不为 90° ，导致剪切应变结果不为 0，与理论矛盾，相当于增加了一部分人工非真实的应变能，导致刚度变大。

2.1.2.2 面外弯曲

面外弯曲和面内弯曲类似，在受面外弯曲载荷作用下，如果完全积分，2X2 的积分点，弯曲后的形状如下图：



在受面外弯曲载荷时，刚度由两部分组成，一部分是面外弯曲刚度，此时得到的剪切应变为面内应变 ϵ_{xy} 与角度 θ_{xy} 很正比，由图上可知，在面外纯弯曲载荷下，这个值一直为 90 度，所以没有剪切应变 ϵ_{xy} ，也就是没有剪切自锁现象。

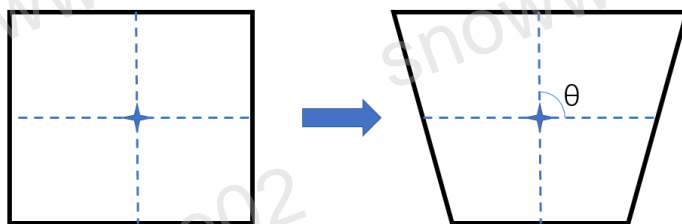
另一部分是面外横向剪切刚度，此时得到的剪切应变为 ϵ_{xz} ，与角度 θ_{xz} 成正比，由上图可知 θ_{xz} 与初始值 90 度不同，得到 ϵ_{xz} 不为 0，与理论不符合。同样产生了剪切自锁。

2.1.3 消除剪切自锁的方法

有剪切自锁不代表不收敛，仅仅只是收敛的慢，通过在长度方向加密网格也可以让结果收敛，但网格的增加会导致计算效率下降很多，所以需要特殊处理消除剪切自锁现象，一般采用下面两种方法：

2.1.3.1 减缩积分

如果采用减缩积分，此时单元在受纯弯曲作用下，发生变形如下：



唯一的积分点横向和纵向连线夹角 θ 依然为 90，所以在该点计算出的剪切应变为 0，和理论一致。

对面内薄膜刚度，由面内拉伸刚度和剪切刚度组成，如果拉伸刚度也是用减缩积分，那么受弯时，拉伸应力在中心积分点上的值为 0，与上方拉伸，下方压缩的理论值矛盾，这将导致另一个问题，也就是后面要论述的沙漏问题了。面外弯曲也有类似现象。

在自编程序 iSolver 中使用 S4 单元时，我们对薄膜刚度采用选择积分的方法，也就是拉伸刚度采用完全积分，剪切刚度采用减缩积分。

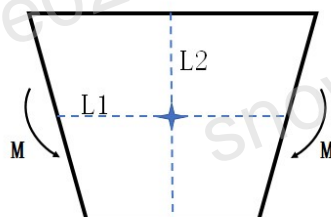
iSolver 计算 Example1 的悬臂梁的结果如下：（相对值=有限元解/理论解），可见采用选择积分能消除剪切自锁。

单元	1X6 网格	2X12	4X12	8X24
iSolver 的 S4 单元	1.156e-1 相对值 0.99	1.162e-1 相对值 0.99	1.162e-1 相对值 0.99	1.164e-1 相对值 0.99

2.1.3.2 假设应变

假设应变的方式是设计合适的应变场，使得剪切自锁消失。

在壳单元受纯弯曲载荷时，可以发现中间线 L2 的两个方向的夹角依然都是 90 度，也就是剪切应变都是 0，那么只要是这条线上的点在纯弯曲下剪切应变都是正确的。假定应变的方法就是剪切应变假定由这条线上两个点插值得到，既然 L2 上随意两个点剪切应变都是 0，那么插值必然也是 0，和理论一致。



在 MacNeal[1]第一次提出这个方法时，用的是 X 和 Y 轴上的高斯点位置，后来 Bathe[2]等改进采用边界中点来插值，此时可以更好的模拟斜单元。

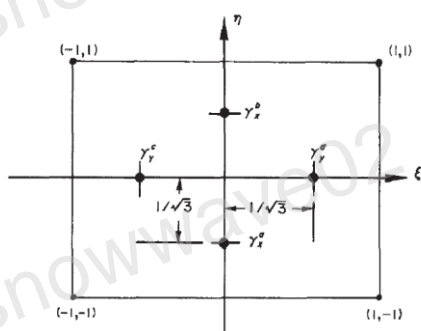
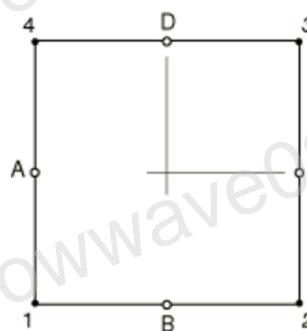


图 1 (a) MacNeal 的插值点



(b) Bathe 的插值点

2.1.4 AbaqusS4 壳单元消除剪切自锁的实现方式

壳单元的剪切自锁只影响薄膜刚度和面外横向剪切刚度。在 Abaqus 的 S4 单元中，都是通过假设应变方法来消除剪切自锁的。

对薄膜刚度：

✧ 详见：Abaqus 6.12 Analysis Manual 29.6.2 Choosing a shell element
Element type S4 is a fully integrated, general-purpose, finite-membrane-strain shell element. The element's membrane response is treated with an assumed strain formulation that gives accurate solutions to in-plane bending problems, is not sensitive to element distortion, and avoids parasitic locking.

对横向剪切刚度：

✧ 详见：Abaqus 6.12 Theory Manual 3.6.5 Finite-strain shell element formulation
Several interpolation schemes have been proposed to avoid shear-locking, which typically arises as the thickness of a plate or shell goes to zero. Here we employ an assumed strain method based on the Hu-Washizu principle.

2.2 沙漏现象介绍

沙漏和剪切自锁可以说是既相同又相反。

和剪切自锁相同的地方在于在线性单元中最明显，且不是所有的载荷都会发生沙漏的，只在受弯下才会发生沙漏现象。

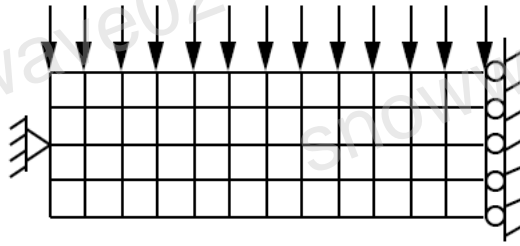
和剪切自锁相反的地方在于剪切自锁发生在完全积分单元中，剪切自锁时会导致单元刚度变大，收敛比较慢，而沙漏发生在减缩积分单元中，且沙漏现象会导致单元刚度变小。这里我们只讨论 S4R 单元。

2.2.1 沙漏现象例子

Example 2: 沙漏例子

一根梁，左端中间节点固支，右端 X 方向约束，上端加力。（取之 Belytchko 的书

[3]）



参数如下：

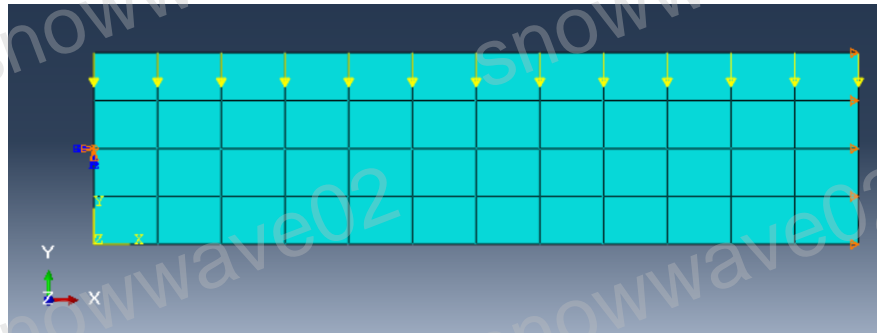
尺寸：X 方向长度 $L=80$ ，Y 方向宽度 $b=20$ ，Z 方向厚度 $h=2$ 。

材料：Young's Modulus $E=11000$ 。

点载荷： $F=10N$ 。

Dynamic, Explicit 分析 1s 的时间。

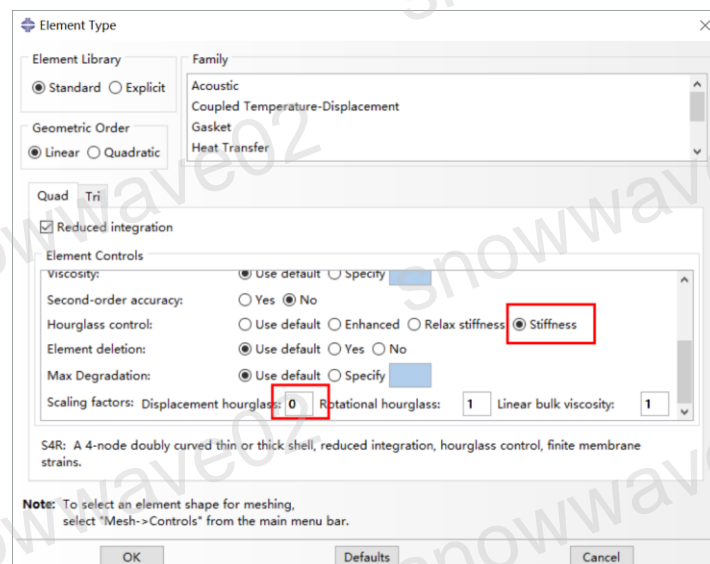
采用 S4R 单元，划分为 4X12 的网格。



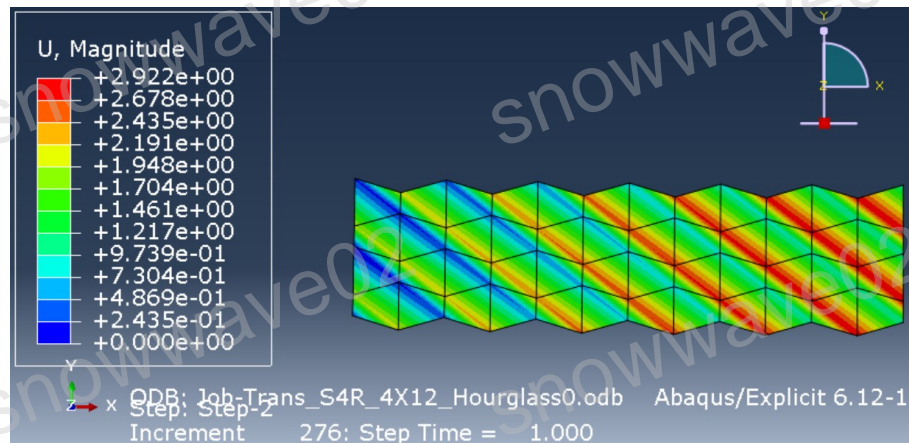
我们将分析没加沙漏控制和加了沙漏控制的情况。

（1）没加沙漏控制

由于受到的是平面内的弯曲载荷，所以只有薄膜部分的计算沙漏才起作用。在 S4R 的设置中，将 Hourglass Control 设为 Stiffness，将薄膜部分的沙漏控制设为 0，那么薄膜部分将不控制沙漏。

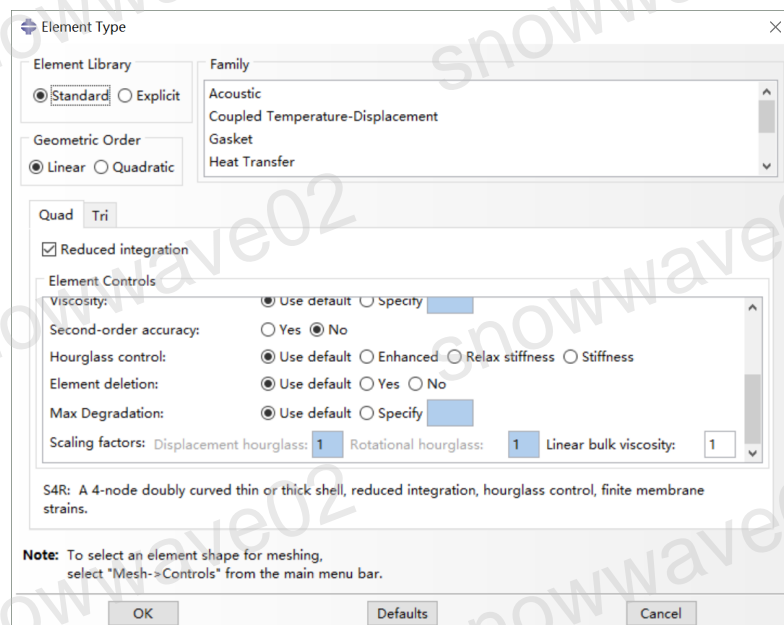


分析后，可以看到 Abaqus 在 1s 后网格交错成沙漏形状，明显存在沙漏现象。

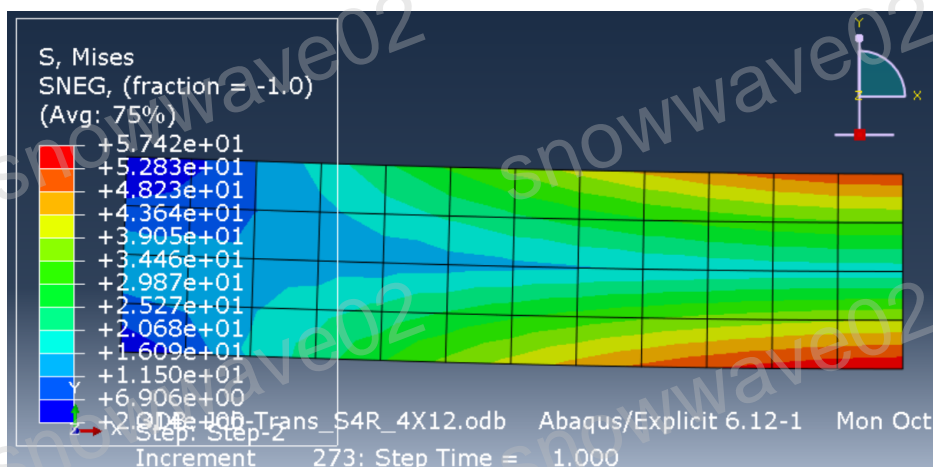


(2) 添加沙漏控制

在 S4R 的设置中，Hourglass 选择 User Default。此时采用 Abaqus 默认的沙漏控制。

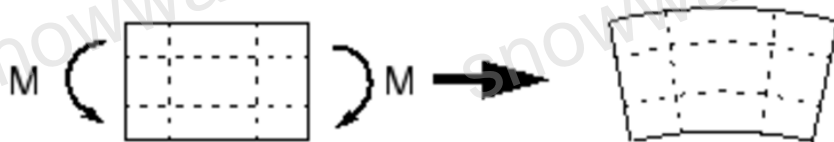


分析后，在 Abaqus 可看到 1s 后的网格形状基本正常（其实沙漏依然存在，后处理中可发现 ALLAE/ALLIE 依然>10%），说明 Abaqus 的 S4R 单元默认对沙漏进行了控制。



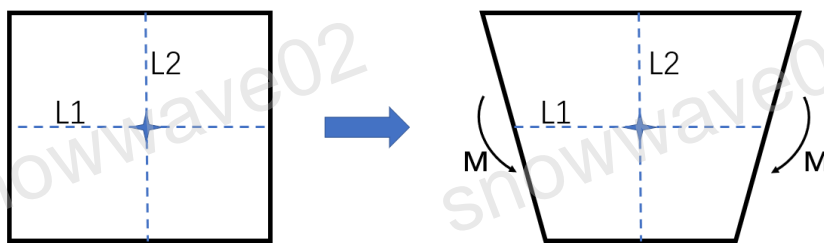
2.2.2 沙漏产生的原因

壳单元受到面内弯矩时，正常情况下形状应该出现如下图所示：



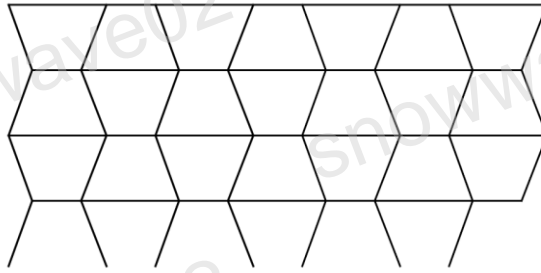
上方受拉，下方受压，导致上方是拉力，下方是压缩力，这样使得在左右边界力矩保持平衡。

对线性减缩单元，将出现如下情况：

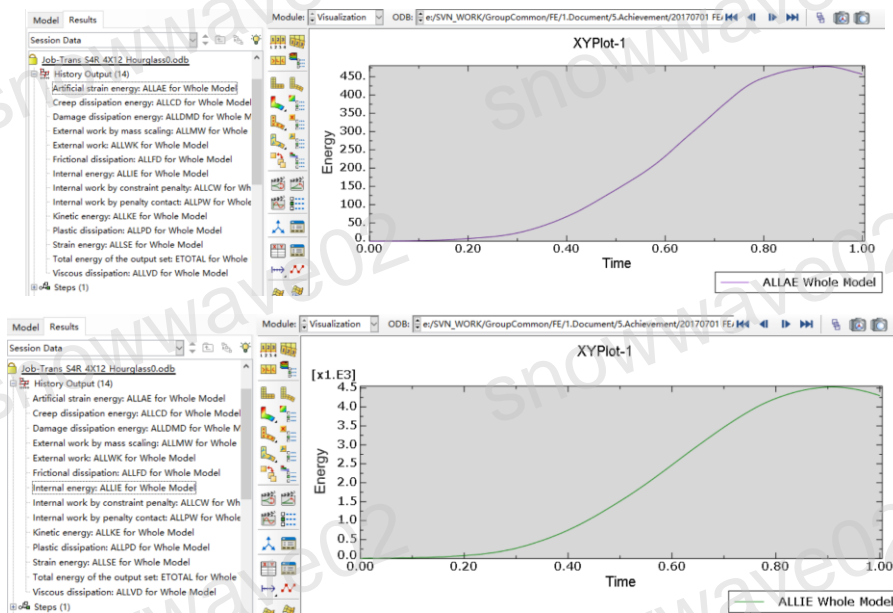


由于单元拉伸应力只计算中心点的值，而中心点 L1 的长度没有变化，使得单元拉伸应力为 0，也就是和外力矩 M 无法平衡，导致单元扭曲的非常厉害。

一个单元扭曲的形状如上图，当存在多个这样的单元时，会形成如下的形状。上下两个单元就类似一个沙漏。



沙漏判断方法最常用的是察看单元变形情况，此时单元变成交替出现的梯形形状。那么就说明产生了沙漏现象。另外可以计算沙漏应变能与内能的比值，在 **Abaqus** 中就是查看 **ALLAE/ALLIE** 的比值，如果 **>5%-10%** 说明出现沙漏现象了，更精准的说这并不是判断出现沙漏的方法，而是判断沙漏控制有没有影响结果的方法。**ALLAE** 是增加了沙漏刚度后，沙漏应变产生的人工应变能，如果 **Abaqus** 没有采取任何沙漏控制，也就是没有增加人工应变能，那么理论上讲人工应变能应该为 0，但因为横向剪切的沙漏应变能用户是无法控制为 0 的，使得 **ALLAE** 在 **S4R** 单元中不可能为 0，同时在出现沙漏时 **ALLIE** 一般比实际值小，使得出现沙漏时 **ALLAE/ALLIE** 变大，所以也可以用这个判据用来判断是否出现沙漏。



由上可知，沙漏主要是因为拉伸应变和理论不一致，也就是只与拉伸刚度有关，与剪切刚度无关。那么，和剪切自锁一样，沙漏在面内受弯和面外受弯时都会存在。

2.2.3 控制沙漏的方法

2.2.3.1 通过算法抑制沙漏的方法

沙漏最终会表现到后台的方程求解上，对线性减缩单元，沙漏会导致薄膜刚度秩的缺失，举个简单的例子，就是会导致类似下方的刚度矩阵。

$$\begin{bmatrix} 1 & 1 \\ 2 & 2 \end{bmatrix}$$

2X2 的矩阵满秩的话秩是 2，但上面这样的矩阵秩只有 1，在加上了载荷后是无解的，只不过在计算机中由于精度误差，计算中的刚度矩阵将会变为类似下方的矩阵：

$$\begin{bmatrix} 1.0000000000000001 & 1 \\ 2 & 2.0000000000000001 \end{bmatrix}$$

这时只要加一个非常小的载荷将导致解非常大。这也是当存在沙漏时，单元变形明显变大的原因。

沙漏控制现在最常用的方法是增加一个沙漏刚度。增加这个沙漏刚度后刚度矩阵将变为类似下方的矩阵：

$$\begin{bmatrix} 1.001 & 1 \\ 2 & 2.001 \end{bmatrix}$$

这个时候解依然存在，不一定是正确解，但肯定比由于计算机精度造成的误差小很多了。

那么增加多少这样的刚度合适呢？

首先，这个沙漏控制只对沙漏模式（设为 H）起作用，但对其它模式（O）不起作用，也就是在没有沙漏现象的正常情况这个刚度应该不起作用，这点通过增加一个归一化的 B 矩阵，使得 $B \cdot H = 1$ ，同时对其它模式 $B \cdot O = 0$ 来实现。

其次，这个沙漏刚度应该有个放大因子决定沙漏刚度的大小，这个沙漏因子的取值受下面两者影响：

- (1) 有效抑制沙漏模式，这点要求这个因子越大越好。
- (2) 沙漏刚度会导致产生人工的沙漏应变能，该能量在实际中并不存在，这点要求这个因子越小越好。

上面两点本身就是矛盾的两点，所以很多时候这个因子根本不存在。

2.2.3.2 改变建模的方法

通过增加人工刚度来抑制沙漏的方法很多情况下沙漏刚度的放大因子无法找到，Abaqus 中 S4R 默认了一个适合大多数情况的沙漏控制的放大因子，如果采用了默认的 S4R 依然出现沙漏，那么大有可能再怎么改沙漏控制也已经无济于事了，更有效的方法是改变建模方法。常用的有以下几种：

- (1) 换完全积分单元，譬如 S4R 换成 S4 单元，这种完全积分单元可以完全消除沙漏出现。
- (2) 改用高次单元，譬如 S8。
- (3) 避免施加单点载荷，改为压力载荷或者分布在多点，也可以减少沙漏。
- (4) 细化网格，对受弯曲力作用时，Abaqus 建议在弯曲方向至少划分 4 个网格。

✧ 详见：Abaqus 6.12 Get Started Manual：4.1 Element formulation and integration

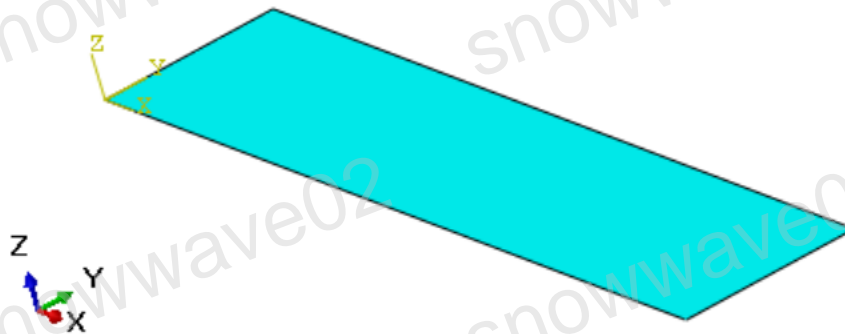
The results suggest that at least four elements should be used through the thickness when modeling any structures carrying bending loads with this type of element.

2.2.4 AbaqusS4R 壳单元沙漏控制的实现方式

Abaqus 的 S4R 壳单元通过算法来控制沙漏，壳单元刚度矩阵分为薄膜刚度、面外弯曲刚度、面外横向剪切刚度和 Drill 刚度。只有前面两项的减缩积分才有沙漏现象，但 Abaqus 对后两项也做了相应的修正，而且认为这些修正也是沙漏现象，这里我们只讨论 S4R 前两者的沙漏控制方式。

在 Abaqus 中通过一个例子的没加沙漏控制和加了沙漏控制后的刚度矩阵的变化，来猜测 Abaqus 加沙漏控制的方式。为了方便，我们只取一个简单的长方形单元作为例子。

Example3: 一个单元的例子



参数如下：

尺寸：5X1，厚度 $1e-4$ 。位于 XY 平面，其中 X 方向 5。

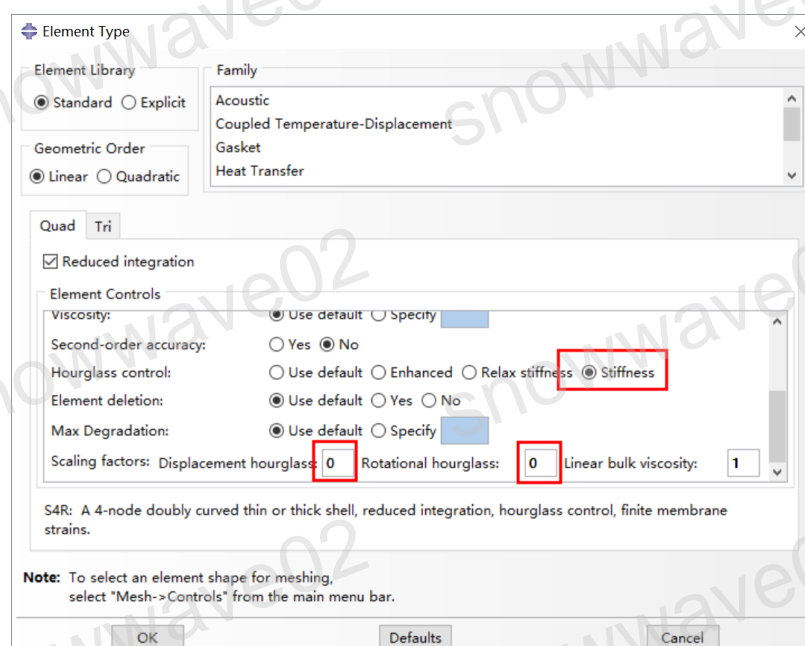
材料：Young's Modulus $1e8$ ，Poisson Ratio 0.3，density 0.0073。

边界固支约束。

(1) 首先按照 S4R 默认设置分析，得到刚度矩阵如下：

	1	2	3	4	5	6
1	$5.3712e+06$	$1.7849e+06$	0	$-5.0022e-12$	$1.2460e-11$	$-7.8557e+03$
2	$1.7849e+06$	$1.3935e+07$	0	$-2.7740e-11$	$5.0022e-12$	$1.5711e+03$
3	0	0	$4.8011e+06$	$2.1499e+06$	$-1.2530e+06$	0
4	$-5.0022e-12$	$-2.7740e-11$	$2.1499e+06$	$1.0441e+06$	$-2.1497e+05$	0
5	$1.2460e-11$	$5.0022e-12$	$-1.2530e+06$	$-2.1497e+05$	$2.0698e+06$	0
6	$-7.8557e+03$	$1.5711e+03$	0	0	0	$1.5711e+04$

(2) 然后将 S4R 的薄膜刚度的面外弯曲刚度的沙漏控制设置为 0。



分析得到刚度矩阵如下：

	1	2	3	4	5	6
1	5.3650e+06	1.7849e+06	0	-5.0022e-12	1.2460e-11	-7.8557e+03
2	1.7849e+06	1.3929e+07	0	-2.7740e-11	5.0022e-12	1.5711e+03
3	0	0	4.8011e+06	2.1499e+06	-1.2530e+06	0
4	-5.0022e-12	-2.7740e-11	2.1499e+06	1.0439e+06	-2.1497e+05	0
5	1.2460e-11	5.0022e-12	-1.2530e+06	-2.1497e+05	2.0697e+06	0
6	-7.8557e+03	1.5711e+03	0	0	0	1.5711e+04

(3) 两者相减得到刚度矩阵的差异就是人工沙漏刚度：

	1	2	3	4	5	6
1	6250	0	0	0	0	0
2	0	6250	0	0	0	0
3	0	0	0	0	0	0
4	0	0	0	187.5000	0	0
5	0	0	0	0	187.5000	0
6	0	0	0	0	0	0

K(1,1)和 K(2,2)为薄膜的人工沙漏刚度，计算方法和 Belytchko 是完全一致的，只不过前面说的沙漏因子 **Abaqus** 取为如下值：

$$(r_F G) = 0.005 G.$$

K(4,4)和 K(5,5)为面外弯曲的人工沙漏刚度，**Abaqus** 沙漏因子在文档中为下面值：

$$(r_\theta G) = 0.00375 \frac{12 \int_{-t/2}^{t/2} G t^2 dt}{t^3}.$$

我验算发现面外弯曲按上面值算出来的人工沙漏刚度和 **Abaqus** 实际输出的差 4 倍，这个 4 倍怎么出来的 **Abaqus** 的文档并没有说明，我也没有查出来。

✧ 详见 **Abaqus 6.12 Analysis User's Manual 27.1.4 Section controls**

Default hourglass stiffness values 章节

2.3 总结

完全积分单元才有剪切自锁，虽然 **Abaqus** 的 **S4** 单元是完全积分，但内部已经做了修正完全消除了剪切自锁，所以不需要用户做任何设置。

减缩积分单元才有沙漏现象，**Abaqus** 的 **S4R** 默认增加一个人工的沙漏刚度来控制沙漏现象，如果发现结果还是不理想，那么需要采用其它建模方法才能控制沙漏了。

Abaqus 针对剪切自锁和沙漏控制做的修正如下：

	单元类型	元素	修正情况
剪切自锁	S4	薄膜刚度	假定应变方法修正
		面外弯曲刚度	无
		面外横向剪切刚度	假定应变方法修正
	S4R	所有项	无
沙漏控制	S4	薄膜刚度	无
		面外弯曲刚度	无
		面外横向剪切刚度	沙漏控制，和 Belytchko 公式不一致，暂时没研究
	S4R	薄膜刚度	和 Belytchko 公式一致，因子取 0.005G
		面外弯曲刚度	和 Belytchko 公式一致，因子取 0.00375G*4
		面外横向剪切刚度	沙漏控制，和 Belytchko 公式不一致，暂时没研究

如有任何疑问，欢迎联系我们：

snowwave02 From www.jishulink.com

email: snowwave02@qq.com

2.4 参考

[1] MacNeal,R.H.,“A Simple Quadrilateral Shell Element,”Computers and Structures, vol. 8, pp. 175–183, 1978.

[2] Bathe, K. J., and E. N. Dvorkin, “A Continuum Mechanics-Based Four-Node Shell Element for General Non-Linear Analysis,” International Journal of Computer Aided Engineering Software, vol. 1, 1984.

[3] Ted Belytschko, book: Nonlinear Finite Elements For Continua And Structures, 2001.