有限元理论基础及Abaqus内部实现方式研究系列14： 壳的应力方向

Theoretical Foundation of Finite Element Method and Internal Implementation of Abaqus Series 14: Shell Stress Direction

SnowWave02 关注 Focus

更新于2019年12月11日 02:17 Updated on December 11, 2019, 02:17 浏览：4865 Views: 4865 评论：7 Comments: 7 收藏：7 Favorited: 7

（原创，转载请注明出处） (Original, please indicate the source for reproduction)

==概述== ==Overview==

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。有限元的理论发展了几十年已经相当成熟，商用有限元软件同样也是采用这些成熟的有限元理论，只是在实际应用过程中，商用CAE软件在传统的理论基础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，每个主流商用软件手册中都会注明各个单元的理论采用了哪种理论公式，但都只是提一下用什么方法修正，很多没有具体的实现公式。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。

This series of articles studies the mature finite element theoretical foundation and its implementation methods in commercial finite element software. The development of finite element theory has matured over decades, and commercial finite element software also adopts these mature finite element theories. However, in the actual application process, commercial CAE software will make corresponding corrections on the basis of traditional theories to solve different problems encountered in engineering, and the correction methods of each software are different. Each mainstream commercial software manual specifies which theoretical formula each element uses, but only mentions the correction method, and many do not provide specific implementation formulas. Commercial software is essentially a black box, and users can only guess its internal implementation methods from outside, except for developers.

一方面我们查阅各个主流商用软件的理论手册并通过进行大量的资料查阅猜测内部修正方法，另一方面我们自己编程实现结构有限元求解器，通过自研求解器和商软的结果比较来验证我们的猜测，如同管中窥豹一般来研究的修正方法，从而猜测商用有限元软件的内部计算方法。我们关注CAE中的结构有限元，所以主要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，同时对某些问题也会涉及其它的Nastran/Ansys等商软。为了理解方便有很多问题在数学上其实并不严谨，同时由于水平有限可能有许多的理论错误，欢迎交流讨论，也期待有更多的合作机会。

On one hand, we consult the theoretical manuals of various mainstream commercial software and guess the internal correction methods through extensive literature review. On the other hand, we program our own structural finite element solver and verify our guesses by comparing the results with those of commercial software. We study the correction methods like a glimpse through a tube, thus guessing the internal calculation methods of commercial finite element software. Since we focus on structural finite elements in CAE, we mainly choose Abaqus, which has relatively complete documentation among commercial structural finite element software, to study the internal implementation methods, and we will also involve other commercial software such as Nastran/Ansys for some issues. Many problems are not mathematically rigorous for the sake of understanding convenience, and due to our limited level, there may be many theoretical errors. We welcome discussions and look forward to more cooperation opportunities.

iSolver介绍视频： iSolver Introduction Video:

http://www.jishulink.com/college/video/c12884

==第14篇：壳的应力方向== == Article 14: Shell Stress Direction ==

有限元理论基础及Abaqus内部实现方式研究系列14： 壳的应力方向的图3

有限元理论基础及Abaqus内部实现方式研究系列14： 壳的应力方向的图4

有限元中，物理量用的最多的是标量、矢量和二阶张量。其中位移、坐标等都是矢量，而应变、应力等都是二阶张量。矢量很容易理解，体的应力等二阶张量直接就采用了全局坐标系的也不会有方向理解问题，但梁壳的应力结果很容易搞错，后处理结果中的S11、S12等的方向有时会觉得和预想的不一致但又不明所以。同时，这个方向也是单元材料的方向，在自编程序时，如果一开始坐标系的定义就弄错了，那么将直接导致和材料相关的刚度矩阵的错误，所以弄清应力的方向定义对自编程序和理解有限元结果都相当重要。

In finite element analysis, the most frequently used physical quantities are scalars, vectors, and second-order tensors. Displacements and coordinates are vectors, while strains and stresses are second-order tensors. Vectors are easy to understand, and second-order tensors like body stresses that are directly adopted in the global coordinate system do not have direction understanding issues. However, the stress results of beams and shells are often misunderstood, and the directions of S11, S12, etc., in the post-processing results sometimes seem inconsistent with expectations but are not clear why. At the same time, this direction is also the direction of the unit material. When writing self-written programs, if the definition of the coordinate system is incorrect from the beginning, it will directly lead to errors in the stiffness matrix related to the material. Therefore, understanding the definition of stress directions is quite important for writing self-written programs and understanding finite element results.

本章将简单介绍一下数学上张量和Abaqus中壳的应力方向，并说明Abaqus这么选取的意义，最后通过自编程序iSolver来验证壳的应力方向的正确性。

This chapter will briefly introduce tensors in mathematics and the stress direction of shells in Abaqus, explain the significance of Abaqus's selection, and finally verify the correctness of the shell stress direction through the self-written program iSolver.

具体的验证详见下方视频（带配音）： The specific verification can be seen in the video below (with narration):

https://www.jishulink.com/college/video/c12884 20.14 理论系列文章14：壳的应力方向

https://www.jishulink.com/college/video/c12884 20.14 Series of Theoretical Articles 14: Shell Stress Direction

有限元理论基础及Abaqus内部实现方式研究系列14： 壳的应力方向的图5

1.1 数学上的张量方向 1.1 Tensor Direction in Mathematics

矢量的方向是一定的，但它的分量都是基于某个坐标系定义的，坐标系不同，那么分量结果也会不同。

The direction of a vector is fixed, but its components are defined based on a coordinate system. Different coordinate systems will result in different component values.

矢量可以表示为： Vectors can be expressed as:

$$\mathbf{a} = a^1\mathbf{e}_1 + a^2\mathbf{e}_2 + a^3\mathbf{e}_3.$$

显然，分量和坐标系的选取有关。譬如我们一般的直角全局坐标系如下，那么分量就是普通的x、y、z三个分量值。

It is obvious that the components are related to the choice of coordinate system. For example, the general rectangular global coordinate system we usually use is as follows, so the components are the ordinary x, y, z values.

$$\begin{aligned}\mathbf{e}_1 &= (1,0,0)\\ \mathbf{e}_2 &= (0,1,0)\\ \mathbf{e}_3 &= (0,0,1)\end{aligned}$$

和矢量类似，二阶张量可以表示如下，当然也可以用一个更简单的3X3的矩阵表示，显然，二阶张量的分量等也与坐标系的取值有关。

Similar to vectors, a second-order tensor can be expressed as follows; of course, it can also be represented by a simpler 3x3 matrix. It is obvious that the components of the second-order tensor are also related to the choice of coordinate system.

$$\begin{aligned}\mathbf{a} = &\mathbf{e}_1 a^{11}\mathbf{e}_1 + \mathbf{e}_1 a^{12}\mathbf{e}_2 + \mathbf{e}_1 a^{13}\mathbf{e}_3\\ &\mathbf{e}_2 a^{21}\mathbf{e}_1 + \mathbf{e}_2 a^{22}\mathbf{e}_2 + \mathbf{e}_2 a^{23}\mathbf{e}_3\\ &\mathbf{e}_3 a^{31}\mathbf{e}_1 + \mathbf{e}_3 a^{32}\mathbf{e}_2 + \mathbf{e}_3 a^{33}\mathbf{e}_3\end{aligned}$$

有限元理论基础及Abaqus内部实现方式研究系列14： 壳的应力方向的图10

1.2 Abaqus壳的应力方向 1.2 Stress direction of Abaqus shell

Abaqus后处理中壳的应力会输出S11，S22，S12等分量，分别对应上面二阶张量a的a11、a22、a12等分量，其它分量不输出，这三个量与壳的坐标系的选取密切相关。Abaqus中，在《anlysis user manual 12.1》的29.6.7 Three-dimensional conventional shell element library的Element output说明了壳采用的是局部坐标系，具体定义如下：

In Abaqus post-processing, the stress of the shell will output S11, S22, S12, etc., which correspond to the a11, a22, a12, etc., components of the above second-order tensor a, and other components are not output. These three quantities are closely related to the selection of the shell coordinate system. In Abaqus, in the "Analysis User Manual 12.1" section 29.6.7 Three-dimensional conventional shell element library, Element output specifies that the shell uses a local coordinate system, which is defined as follows:

S11：Local 11 direct stress.

S11: Local 11 direct stress.

S22：Local 22 direct stress.

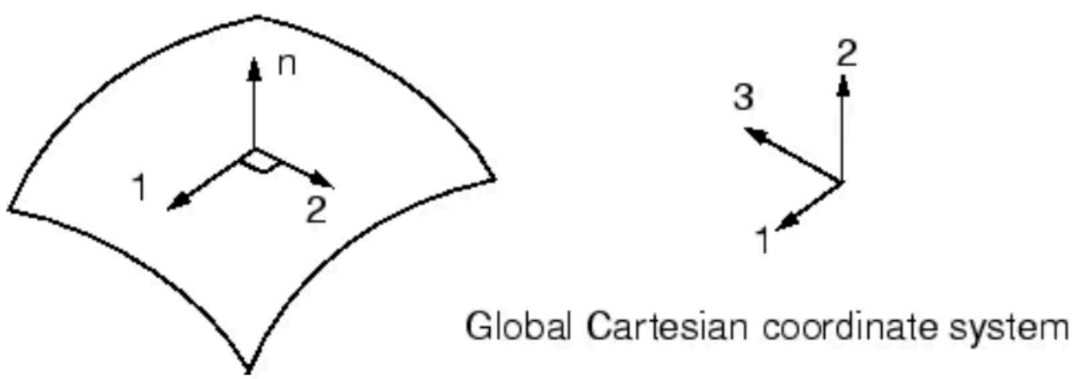
S22: Local 22 direct stress.

S12：Local 12 shear stress.

S12: Local 12 shear stress.

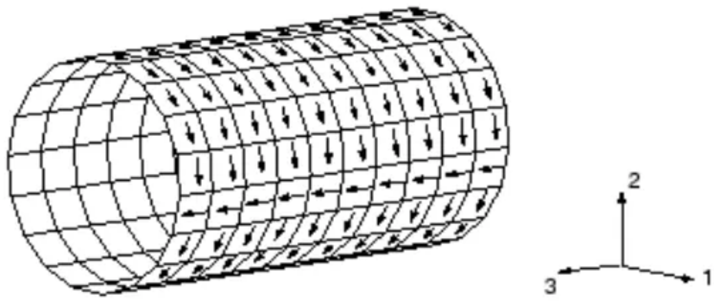
（1） local 1（以下称为T1方向）默认情况下为x方向在表面上的投影。

(1) Local 1 (hereinafter referred to as the T1 direction) is the projection of the x-direction on the surface by default.



如果x方向正好垂直于表面(Abaqus取cos夹角<0.1度），那么取z方向投影，所以下方的圆柱面上当x轴和表面垂直时，S11的方向明显和上下不一致了。

If the x-direction is exactly perpendicular to the surface (Abaqus takes the cosine angle to be less than 0.1 degrees), then the projection in the z-direction is taken. Therefore, when the x-axis is perpendicular to the surface on the lower cylindrical surface, the direction of S11 is obviously different from the top and bottom.



（2）n方向（即T3方向）垂直于壳的表面，至于是往上还是往下，由节点顺序决定。

(2) The n direction (i.e., the T3 direction) is perpendicular to the shell surface. Whether it goes up or down is determined by the node sequence.

（3）第三个方向local 2（以下称为T2方向）和T1，T3满足右手定则。显然在壳的表面绕T3转动90度。

(3) The third direction, local 2 (hereinafter referred to as T2 direction), and T1, T3 satisfy the right-hand rule. It is obvious that the surface of the shell rotates 90 degrees around T3.

由上面的定义，可以看出应力的方向的S11，S22都是沿着表面的，而不是全局坐标系下的。

According to the above definition, it can be seen that the stress directions S11 and S22 are along the surface, rather than in the global coordinate system.

有限元理论基础及Abaqus内部实现方式研究系列14：壳的应力方向的图15
1.3 应力方向选取原因 1.3 Reason for Stress Direction Selection

壳的应力方向为何要取成上面的T1，T2和T3？有两个原因：

Why should the stress directions of the shell be taken as T1, T2, and T3? There are two reasons:

有限元理论基础及Abaqus内部实现方式研究系列14：壳的应力方向的图16
1.3.1 插值函数 1.3.1 Interpolation Function

T1，T2在面内才能使得坐标和位移可以使用二维平面内的插值公式。

T1, T2 must be within the plane to enable the use of interpolation formulas within the two-dimensional plane.

位移的插值是有限元的基础，插值方法如下： The interpolation of displacement is the foundation of finite element analysis, and the interpolation methods are as follows:

$$x = \sum_{i=1}^{4} N_i x_i; \quad y = \sum_{i=1}^{4} N_i y_i$$

壳的插值方式有两种，一种是平面壳的理论，先按平面壳来计算K，然后再将K做坐标变换，此时单元内部任意点的坐标值只有x，y，没有z，插值为四个节点的坐标。这种方法就要求在壳的坐标系下，四个节点的z方向的坐标zi=0，只剩xi和yi，这种方法计算精度不是很准，因为一个四边形的四个点不能总是保证在一个平面上的，只要是

曲面划分成四边形很有肯定就会是这种情况。另一种更精确的算法是当做曲面壳，abaqus就是这么做的。此时坐标的插值函数不变，但换为了三个坐标x、y、z的分别插值，四个节点的z方向的坐标zi不需要强制要求是0了。

There are two types of interpolation methods for shells: one is the theoretical method for plane shells, which first calculates the stiffness matrix K according to the plane shell, and then transforms K with coordinates. In this case, the coordinates of any point within the element only have x and y, without z, and the interpolation is based on the coordinates of the four nodes. This method requires that the z-directional coordinates of the four nodes in the shell coordinate system be 0, leaving only xi and yi. This method is not very accurate in terms of calculation precision because it cannot be guaranteed that the four points of a quadrilateral are always on a plane. Whenever a curved surface is divided into quadrilaterals, it is very likely to be the case. Another more accurate algorithm is to treat it as a curved shell, which is how Abaqus does it. In this case, the interpolation function for coordinates remains unchanged, but it is replaced with interpolation for the three coordinates x, y, and z separately, and there is no longer a requirement for the z-directional coordinates of the four nodes to be 0.

不管是哪种方法，插值函数都是二维的，如下： Regardless of the method, the interpolation function is two-dimensional, as follows:

$$N_1(\xi,\eta) = l_1(\xi)l_1(\eta) = \frac{1}{4}(1-\xi)(1-\eta)$$

$$N_2(\xi,\eta) = l_2(\xi)l_1(\eta) = \frac{1}{4}(1+\xi)(1-\eta)$$

$$N_3(\xi,\eta) = l_2(\xi)l_2(\eta) = \frac{1}{4}(1+\xi)(1+\eta)$$

$$N_4(\xi,\eta) = l_1(\xi)l_2(\eta) = \frac{1}{4}(1-\xi)(1+\eta)$$

只有T1、T2在壳平面内实际坐标才能映射到等参上。

Only the actual coordinates of T1 and T2 within the shell plane can be mapped to the isoparametric coordinates.

有限元理论基础及Abaqus内部实现方式研究系列14： 壳的应力方向的图21

1.3.2 本构关系 1.3.2 Constitutive Relationship

壳的应变方向为何要取成T3和面垂直的另一个原因是因为只有这样用壳来简化分析对象时材料的本构关系才是最简单的。因为壳是对体的简化，当体的厚度远小于面内尺寸时，那么可以用壳的理论来近似，此时可以把壳的刚度分为薄膜效应刚度、面外弯曲刚度、面外横向剪切刚度等部分，每一部分本构关系由相应简单的材料相关的D矩阵决定，譬如薄膜效应和面外弯曲的D矩阵都是下方矩阵：

Another reason why the stress direction of the shell needs to be taken as T3 and perpendicular to the surface is that only in this way, when using the shell to simplify the analysis object, the material's constitutive relationship is the simplest. Because the shell is a simplification of the body, when the thickness of the body is much smaller than the in-plane dimensions, the shell theory can be approximated, at this time, the stiffness of the shell can be divided into parts such as membrane effect stiffness, out-of-plane bending stiffness, and out-of-plane shear stiffness, and the constitutive relationship of each part is determined by the corresponding simple material-related D matrix, for example, the D matrices for membrane effect and out-of-plane bending are as follows:

$$
\begin{bmatrix}
\frac{E}{1-\nu^2} & \frac{\nu E}{1-\nu^2} & 0 \\
\frac{\nu E}{1-\nu^2} & \frac{E}{1-\nu^2} & 0 \\
0 & 0 & G=\frac{E}{2(1+\nu)}
\end{bmatrix}
$$

上面的本构关系是各向同性材料的，面内的T1和T2不影响D矩阵，但各向异性就不一样，此时D矩阵将与面内的T1，T2相关，这也是应力的坐标系也叫做材料坐标系的原因。

The constitutive relationship mentioned above is for isotropic materials, where the in-plane T1 and T2 do not affect the D matrix. However, this is not the case for anisotropic materials, where the D matrix will be related to the in-plane T1 and T2. This is why the stress coordinate system is also called the material coordinate system.

有限元理论基础及Abaqus内部实现方式研究系列14： 壳的应力方向的图24
1.4 程序内部实现方法 1.4 Method of Internal Implementation

其实，原理非常简单，但程序实现并不像看起来的那么简单。无论是Abaqus还是自编程序内部想要实现壳的应力的方向，和前面讨论的应力方向选取原因是一致的，主要是两点：

In fact, the principle is very simple, but the program implementation is not as simple as it looks. Whether it is Abaqus or a self-written program, to implement the stress direction of shells internally, it is consistent with the reasons for selecting the stress direction discussed earlier, mainly involving two points:

（1）坐标和位移用局部坐标系表示，也就是下面等式：

(1) Coordinates and displacements are represented in a local coordinate system, as shown in the following equation:

$$\mathbf{x}(S_i) = \overline{\mathbf{x}}(S_\alpha) + \overline{f}_{33}(S_\alpha)\mathbf{t}_3(S_\alpha)S_3.$$

S1，S2，S3分别对应T1，T2，T3三个方向的坐标分量。

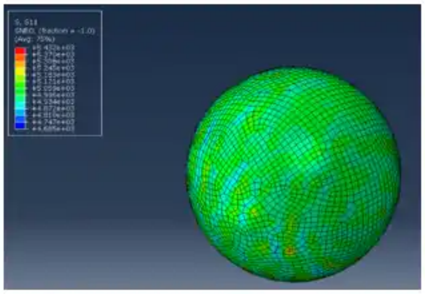
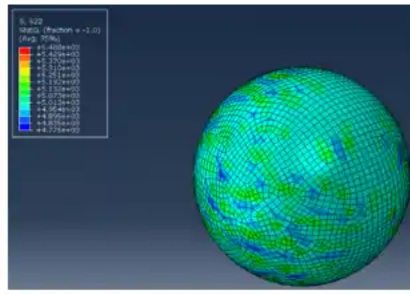
S1, S2, S3 correspond to the coordinate components in the T1, T2, T3 directions, respectively.

（2）本构关系用局部坐标系做旋转变化。 The constitutive relationship is rotated using a local coordinate system.

有限元理论基础及Abaqus内部实现方式研究系列14： 壳的应力方向的图27
1.5 iSolver验证 1.5 iSolver verification

为了证明壳的S11、S22是沿表面方向而不是全局坐标系方向，我们用一个球壳加均匀压力，可发现S11和S22在abaqus中是球对称的，只有沿着表面才会是应力相等，如果沿着全局的xyz，那么显然不应该球对称。

In order to prove that the S11, S22 of the shell are along the surface direction rather than the global coordinate system direction, we apply a uniform pressure to a spherical shell and find that S11 and S22 are spherically symmetric in Abaqus, only being equal along the surface. If along the global xyz, it is obviously not spherically symmetric.

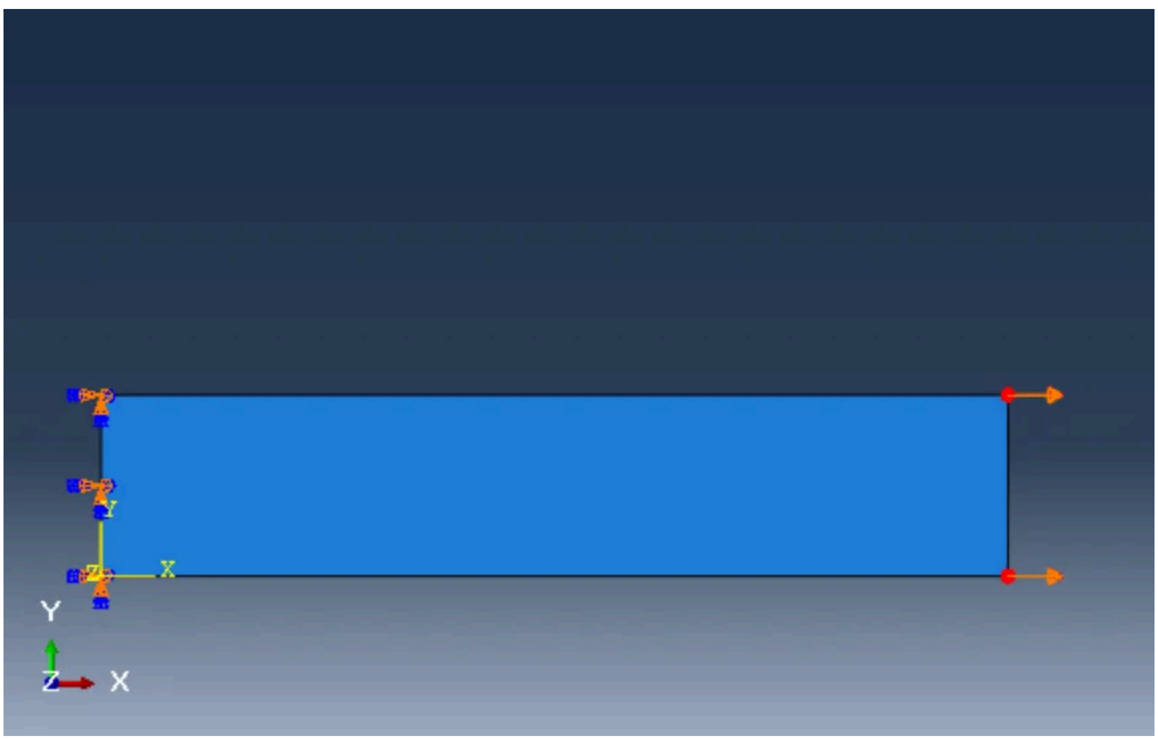
为了进一步验证，我们在iSolver中按照Abaqus的T1、T2、T3定义壳单元方向，并通过一个简单的例子来验证Abaqus壳的应力方向。

To further verify, we define the shell element direction in iSolver according to Abaqus' T1, T2, T3, and verify the stress direction of Abaqus shell through a simple example.

有限元理论基础及Abaqus内部实现方式研究系列14：壳的应力方向的图30

1.5.1 算例 Example 1.5.1

只取一个简单的长方形单元。 Only take a simple rectangular element.



参数如下： Parameters as follows:

尺寸：5X1，厚度0.1。 Dimensions: 5X1, thickness 0.1.

材料：Young’s Modulus 1e8， Poisson Ratio 0.3。

Material: Young's Modulus 1e8, Poisson Ratio 0.3.

右侧两个节点固支。 The two nodes on the right are fixed.

左侧两个节点每个加集中力1e5，在壳的平面内往外拉伸。

Each of the two nodes on the left is subjected to a concentrated force of 1e5, pulling outward in the plane of the shell.

一开始在XY平面内，长度5方向在X轴，宽度方向在Y轴上。此时将得到一个应力的二阶张量S。

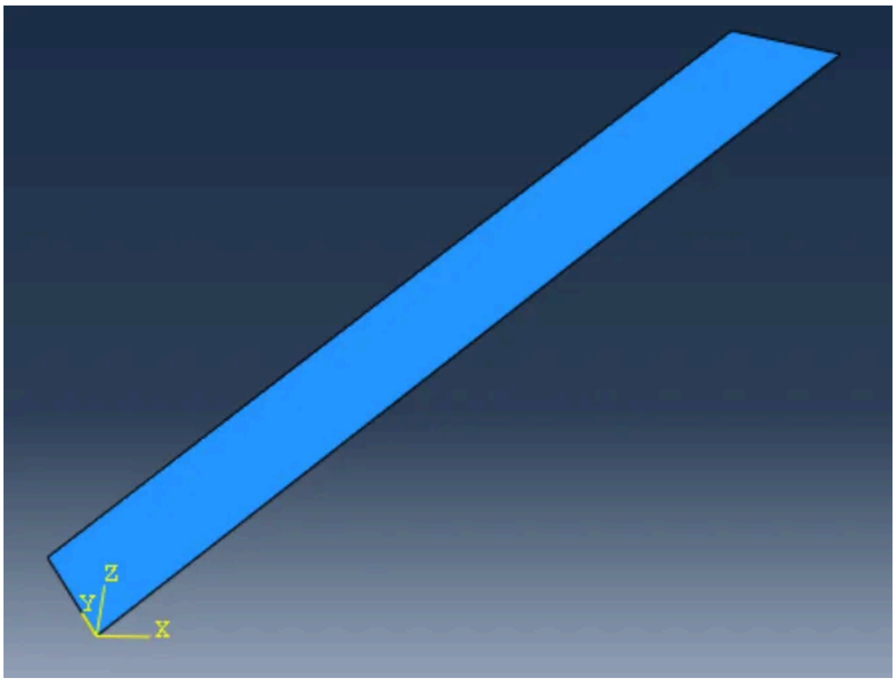
Initially in the XY plane, the length direction along the X-axis, and the width direction along the Y-axis. At this point, a second-order stress tensor S will be obtained.

有限元理论基础及Abaqus内部实现方式研究系列14： 壳的应力方向的图33

1.5.2 Y轴转动45度 Rotate 45 degrees around the Y-axis.

沿Y轴转动45度，按照Abaqus的定义方式，显然，二阶张量S完全不变。

Rotating 45 degrees along the Y-axis, according to the definition method of Abaqus, it is obvious that the second-order tensor S remains completely unchanged.

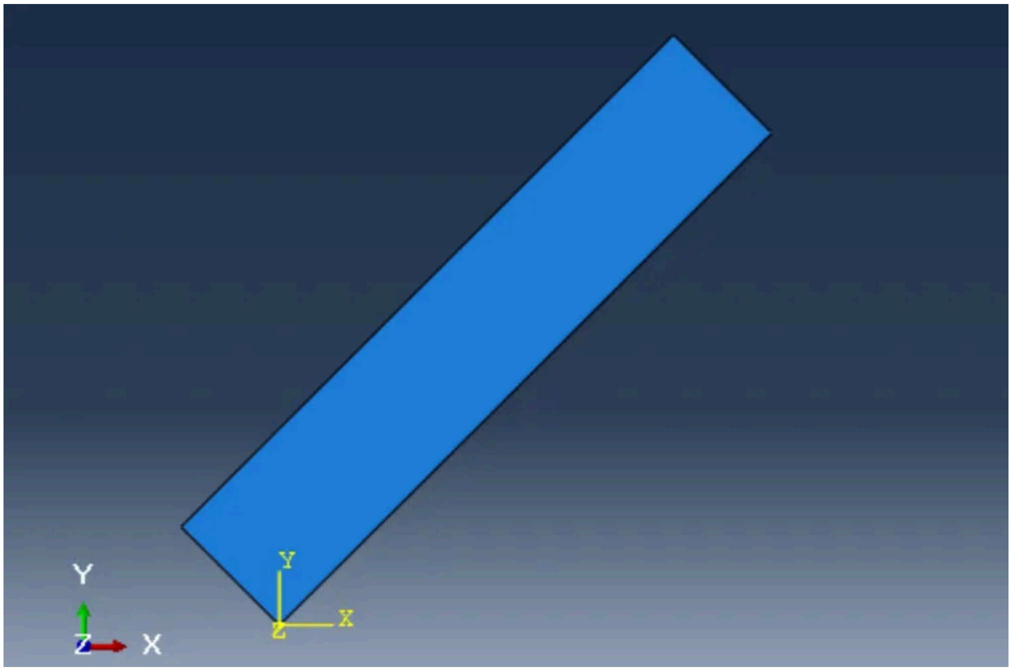


有限元理论基础及Abaqus内部实现方式研究系列14： 壳的应力方向的图36

1.5.3 Z轴转动45度 1.5.3 Rotation around the Z-axis by 45 degrees

沿Z轴转动45时，显然，壳的应力将不同，需要坐标变换。

When rotated 45 degrees around the Z-axis, it is obvious that the shell's stress will be different, and a coordinate transformation is required.



设R为从原始模型的到新模型的变换矩阵，那么从数学的角度如果是一个列矢量a，那么变换后的矢量为：

Let R be the transformation matrix from the original model to the new model. Then, from a mathematical perspective, if a is a column vector, the transformed vector is:

A=R*a A = R * a

而如果是二阶张量S，在原始坐标系下二阶张量可以表示为

If it is a second-order tensor S, in the original coordinate system, a second-order tensor can be represented as

S=a*b’ S = a * b'

b’为列矢量b的转置。 The transpose of column vector b.

显然变换后的张量为： Clearly, the transformed tensor is:

SNew=(R*a)*(R*b)’=R*a*b’*R’=R*S*R’

SNew=(R*a)*(R*b)'=R*a*b'*R'=R*S*R'

其中R’表示转置。 Where R' denotes the transpose.

有限元理论基础及Abaqus内部实现方式研究系列14： 壳的应力方向的图39

1.5.4 iSolver验证演示 1.5.4 iSolver Verification Demonstration

iSolver中详细的验证证明请参考演示视频（带配音）：

For detailed verification proof in iSolver, please refer to the demonstration video (with narration):

https://www.jishulink.com/college/video/c12884 20理论系列文章14：壳的应力方向

https://www.jishulink.com/college/video/c12884 20 Theory Series Article 14: Shell Stress Direction

有限元理论基础及Abaqus内部实现方式研究系列14： 壳的应力方向的图40

==总结== ==Summary==

本章简单介绍了一下数学上张量和Abaqus中壳的应力方向，并说明Abaqus这么选取的意义，最后通过自编程序iSolver来验证壳的应力方向的正确性。

This chapter briefly introduces tensors in mathematics and the stress direction of shells in Abaqus, explains the significance of Abaqus's selection, and finally verifies the correctness of the stress direction of shells through the self-written program iSolver.

如果有任何其它疑问或者项目合作意向，也欢迎联系我们：

If you have any other questions or intentions for project cooperation, feel free to contact us:

snowwave02 From www.jishulink.com

email: snowwave02@qq.com

以往的系列文章： Previous series articles:

第一篇：**S4壳单元刚度矩阵研究。**介绍Abaqus的S4刚度矩阵在普通厚壳理论上的修正。

First article: Research on the Stiffness Matrix of S4 Shell Element. Introduces the correction of Abaqus' S4 stiffness matrix in the theory of ordinary thick shell.

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

第二篇：**S4壳单元质量矩阵研究。**介绍Abaqus的S4和Nastran的Quad4单元的质量矩阵。

Second article: Research on the Mass Matrix of S4 Shell Element. Introduces the mass matrices of Abaqus' S4 and Nastran's Quad4 elements.

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

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。

Third article: Shear locking and hourglass control of S4 shell elements. Introduces how Abaqus S4 elements eliminate shear locking and how S4R suppresses hourglassing.

http://www.jishulink.com/content/post/350865

第四篇：**非线性问题的求解。**介绍Abaqus在非线性分析中采用的数值计算的求解方法。

Fourth article: Solution of nonlinear problems. This article introduces the numerical computation methods adopted by Abaqus in nonlinear analysis.

http://www.jishulink.com/content/post/360565

第五篇：**单元正确性验证。**介绍有限元单元正确性的验证方法，通过多个实例比较自研结构求解器程序iSolver与Abaqus的分析结果，从而说明整个正确性验证的过程和iSolver结果的正确性。

Fifth article: Element correctness verification. Introduces the verification methods for finite element element correctness, compares the analysis results of the self-developed structural solver program iSolver with Abaqus through multiple examples, thereby illustrating the entire correctness verification process and the correctness of the iSolver results.

https://www.jishulink.com/content/post/373743

第六篇：**General梁单元的刚度矩阵。**介绍梁单元的基础理论和Abaqus中General梁单元的刚度矩阵的修正方式，采用这些修正方式可以得到和Abaqus梁单元完全一致的刚度矩阵。

Sixth article: Stiffness matrix of General beam element. Introduces the basic theory of beam elements and the correction methods of the General beam element stiffness matrix in Abaqus. By using these correction methods, it is possible to obtain a stiffness matrix that is completely consistent with the Abaqus beam element.

https://www.jishulink.com/content/post/403932

第七篇：**C3D8六面体单元的刚度矩阵。**介绍六面体单元的基础理论和Abaqus中C3D8R六面体单元的刚度矩阵的修正方式，采用这些修正方式可以得到和Abaqus六面体单元完全一致的刚度矩阵。

Seventh article: Stiffness matrix of C3D8 hexahedral element. Introduces the basic theory of hexahedral elements and the correction methods of the C3D8R hexahedral element stiffness matrix in Abaqus. By using these correction methods, it is possible to obtain a stiffness matrix that is completely consistent with the Abaqus hexahedral element.

https://www.jishulink.com/content/post/430177

第八篇：**UMAT用户子程序开发步骤。**介绍基于Fortran和Matlab两种方式的Abaqus的UMAT的开发步骤，对比发现开发步骤基本相同，同时采用Matlab更加高效和灵活。

Eighth article: Steps for UMAT user subroutine development. Introduces the development steps of Abaqus UMAT based on both Fortran and Matlab, and finds that the development steps are basically the same. At the same time, Matlab is found to be more efficient and flexible.

https://www.jishulink.com/content/post/432848

第九篇：**编写线性UMAT Step By Step。**介绍基于Matlab线性零基础，从零开始Step by Step的UMAT的编写和调试方法，帮助初学者UMAT入门。

Chapter 9: Writing Linear UMAT Step by Step. Introduces the writing and debugging methods of UMAT based on Matlab linear zero foundation, starting from scratch step by step to help beginners get started with UMAT.

有限元理论基础及Abaqus内部实现方式研究系列14：壳的应力方向的图41

有限元理论基础及Abaqus内部实现方式研究系列14：壳的应力方向的图42

有限元理论基础及Abaqus内部实现方式研究系列14：壳的应力方向的图43

http://www.jishulink.com/content/post/440874

第十篇：**耦合约束（Coupling constraints）的研究。**介绍Abaqus中耦合约束的原理，并使用两个简单算例加以验证。

Chapter 10: Research on Coupling Constraints. Introduce the principle of coupling constraints in Abaqus and verify it with two simple examples.

http://www.jishulink.com/content/post/531029

第十一篇：**自主CAE开发实战经验第一阶段总结。**介绍了iSolver开发以来的阶段性总结，从整体角度上介绍一下自主CAE的一些实战经验，包括开发时间预估、框架设计、编程语言选择、测试、未来发展方向等。

The eleventh article: Summary of the first phase of independent CAE development experience. It introduces the phase-by-phase summary of the development of iSolver, and gives an overall introduction to some practical experiences of independent CAE, including development time estimation, framework design, programming language selection, testing, and future development directions.

http://www.jishulink.com/content/post/532475

第十二篇：**几何梁单元的刚度矩阵。**研究了Abaqus中几何梁的B31单元的刚度矩阵的求解方式，以L梁为例，介绍General梁用到的面积、惯性矩、扭转常数等参数在几何梁中是如何通过几何形状求得的，根据这些参数，可以得到和Abaqus完全一致的刚度矩阵，从而对只有几何梁组成的任意模型一般都能得到Abaqus完全一致的分析结果，并用一个简单的算例验证了该想法。

Twelfth article: Stiffness Matrix of Geometric Beam Element. This article studies the method of solving the stiffness matrix of the B31 element of geometric beam in Abaqus, taking the L beam as an example, and introduces how the parameters such as area, moment of inertia, and torsion constant used in General beam are obtained through geometric shape in geometric beam. Based on these parameters, a stiffness matrix consistent with Abaqus can be obtained, so that for any model composed only of geometric beams, Abaqus can generally obtain consistent analysis results. This idea is verified by a simple example.

http://www.jishulink.com/content/post/534362

第十三篇：**显式和隐式的区别。**介绍了显式和隐式的特点，并给出一个数学算例，分别利用前向欧拉和后向欧拉求解，以求直观表现显式和隐式在求解过程中的差异，以及增量步长对求解结果的影响。

Thirteenth article: The difference between explicit and implicit. It introduces the characteristics of explicit and implicit methods, and provides a mathematical example, using forward Euler and backward Euler methods respectively to solve, in order to intuitively demonstrate the differences between explicit and implicit methods in the solution process, as well as the influence of the increment step size on the solution results.

http://www.jishulink.com/content/post/537154