有限元理论基础及Abaqus内部实现方式研究系列28： 几何非线性的T.L.和U.L.描述方法

Theoretical Foundation of Finite Element Method and Research on the Internal Implementation of Abaqus Series 28: Description Methods of T.L. and U.L. for Geometric Nonlinearity

SnowWave02 关注 Focus

更新于2020年10月11日 03:20 Updated on October 11, 2020, 03:20

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

有限元理论基础及Abaqus内部实现方式研究系列28： 几何非线性的T.L.和U.L.描述方法的图1

有限元理论基础及Abaqus内部实现方式研究系列28： 几何非线性的T.L.和U.L.描述方法的图2

1 概述 1 Overview

有限元理论基础及Abaqus内部实现方式研究系列28： 几何非线性的T.L.和U.L.描述方法的图3

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，通过

This series of articles studies the mature finite element theory foundation and its implementation in commercial finite element software, through

（1） 基础理论 (1) Basic Theory

（2） 商软操作 (2) Commercial Software Operation

（3） 自编程序 (3) Self-written program

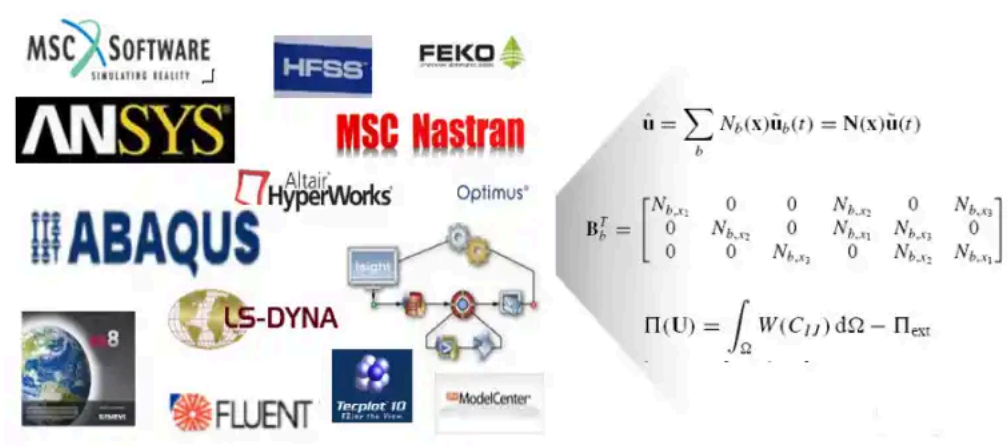
三者结合的方式将复杂繁琐的结构有限元理论通过简单直观的方式展现出来，同时深层次的学习有限元理论和商业软件的内部实现原理。

The combination of the three methods presents the complex and cumbersome structural finite element theory in a simple and intuitive way, while also deeply studying the internal implementation principles of finite element theory and commercial software.

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。

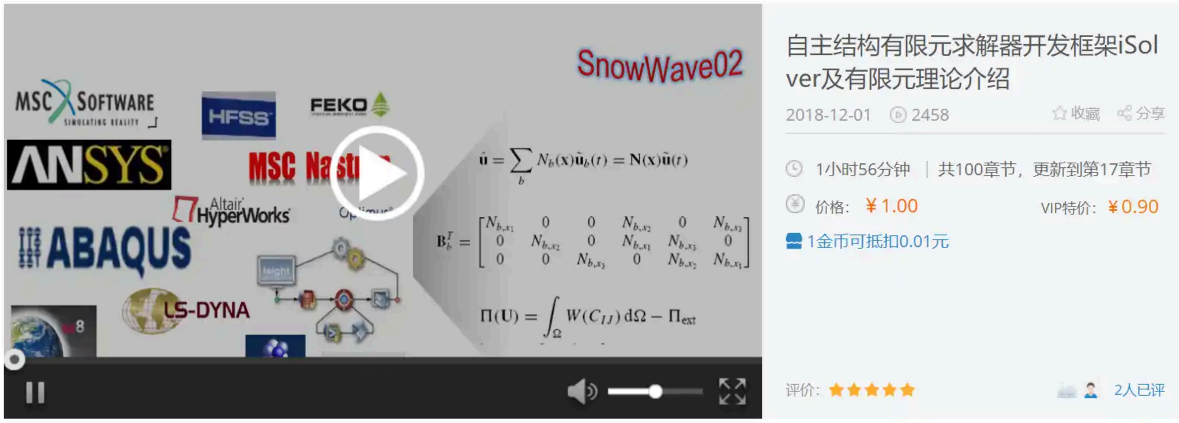
The theoretical development of finite elements has matured over decades, and commercial finite element software also adopts these mature finite element theories. However, in the actual application process, commercial CAE software will make corresponding corrections on the basis of traditional theories 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 a black box to the outside, and users can only guess the 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.

通用结构有限元求解器iSolver介绍视频： Introduction to the General Structural Finite Element Solver iSolver Video:



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

==第28篇：几何非线性的T.L.和U.L.描述方法==

==28th Article: Description Methods of T.L. and U.L. for Geometric Nonlinearity==

物理空间的很多量和方程都是相对某个参照系或者时刻点的，几何非线性的方程也不例外。K.J.Bathe教授1979年首次提出了应用于工程问题的几何非线性理论，他将几何非线性理论公式分为完全拉格朗格式(T.L. Total Lagrangrian格式)和更新拉格朗日格式(U.L. Updated Lagrangian格式)两种公式描述方式，物理方程中的所有的物理量统一在一种描述方式下表示，最终建立几何非线性的求解方程。此后，这两种描述方式称为商用结构有限元最广泛采用的几何非线性描述方法方法，此文将简单介绍一下物理量、网格的Lagrangina和Euler描述，虚功原理的T.L.和U.L.描述方法，然后通过和自主结构求解器iSolver的比对，验证Abaqus的几何非线性的描述方法。

Many quantities and equations in the physical space are relative to a reference frame or a specific moment, and equations of geometric nonlinearity are no exception. Professor K.J.Bathe first proposed the geometric nonlinearity theory applied to engineering problems in 1979. He divided the geometric nonlinearity theory formulas into two types: the Total Lagrangian (T.L.) format and the Updated Lagrangian (U.L.) format. All physical quantities in the physical equations are represented under a unified description method, and finally, the solution equations for geometric nonlinearity are established. Since then, these two description methods have become the most widely used geometric nonlinearity description methods in commercial structural finite element analysis. This article will briefly introduce the Lagrangian and Eulerian descriptions of physical quantities and grids, the T.L. and U.L. descriptions of the principle of virtual work, and then verify Abaqus' geometric nonlinearity description method through comparison with the self-developed structural solver iSolver.

1.1 初始构型和当前构型 1.1 Initial Configuration and Current Configuration

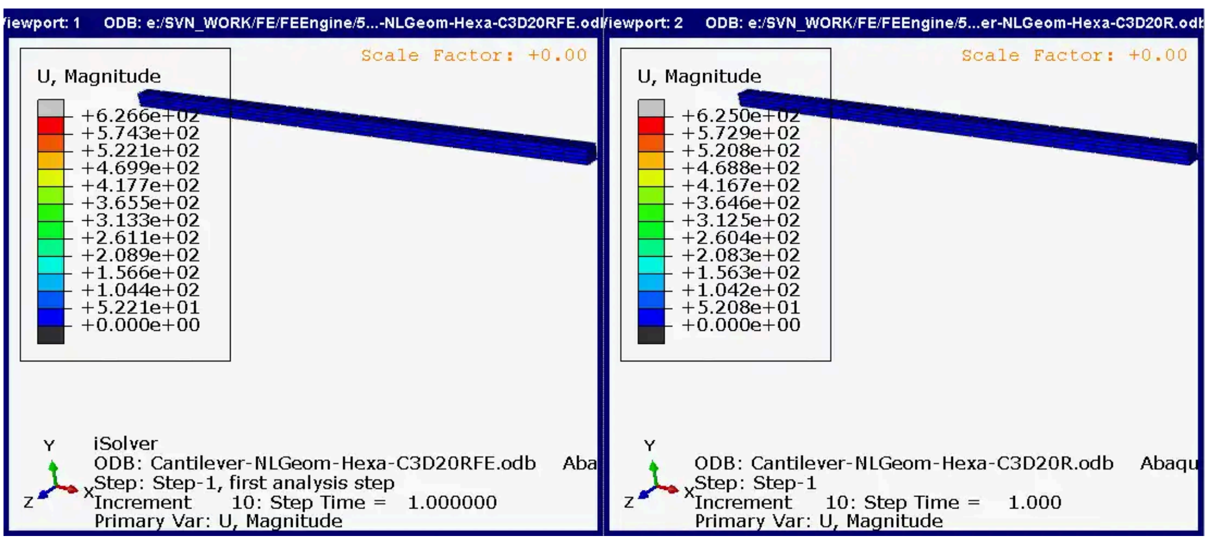
我们先从构型的含义说起，几何非线性理论由三维连续体的虚功方程，提出了适用大位移、大转动几何非线性的解法。若把物体视为由无数质点所构成，并把这些质点称为材料点，一个物体从初始状态由于受到外部载荷运动，运动到另一个状态的过程中，每一个时刻，物体中所有材料点的位置的集合，定义为该物体的一个构型

（configuration），且材料点不会凭空消失。

We start with the meaning of configuration. The geometric nonlinear theory, derived from the virtual work equation of three-dimensional continua, proposes a solution method applicable to large displacement and large rotation geometric nonlinearities. If an object is regarded as consisting of countless particles, and these particles are called material points, the collection of positions of all material points in an object at any given moment, during the process of moving from an initial state to another state due to external loads, is defined as a configuration (configuration) of the object, and material points will not disappear spontaneously.

物体的材料点在0时刻占有的区域称为初始构型C0（initial configuration），材料点的位置在在当前t时刻占有的区域称为当前构型Ct（current configuration）。譬如下面的Abaqus和iSolver计算的几何大变形下的悬臂梁，初始时刻，悬臂梁在一个水平平面上，由未变形的体单元组成初始构型。而下压的任意时刻t，悬臂梁将弯曲，由变形的体单元组成当前构型。

The region occupied by the material points of an object at time 0 is called the initial configuration C0 (initial configuration), and the region occupied by the positions of the material points at the current time t is called the current configuration Ct (current configuration). For example, the cantilever beam under large deformation calculated by Abaqus and iSolver, at the initial moment, the cantilever beam is on a horizontal plane, consisting of undeformed solid elements to form the initial configuration. And at any moment t when it is pressed down, the cantilever beam will bend, consisting of deformed solid elements to form the current configuration.



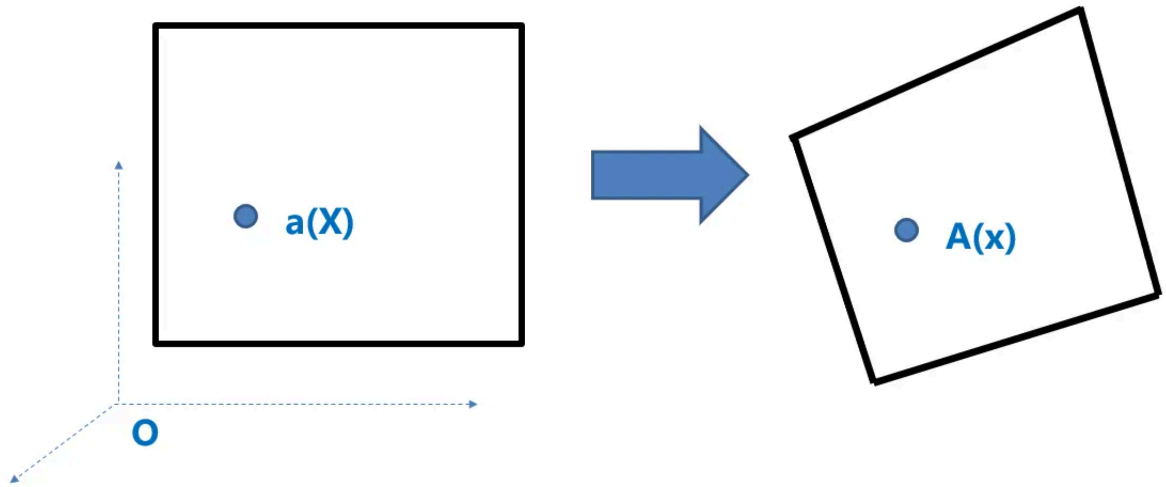
1.2 物理量的Lagrangian和Euler描述方式

1.2 Lagrangian and Eulerian descriptions of physical quantities

为描述各个物理量，可以为初始构型和当前构型选取不同的参考坐标系。当然，一般情况下，无论是初始构型还是当前构型，都以空间中固定不变的全局直角坐标系O来参考，此时，C0下任意一个材料点a在该全局坐标系O下记为X，因为材料点不会凭空消失，在t时刻C1构型下，必然会移动到某一位置A，该点在同一个全局坐标系下的坐标

为x。

To describe various physical quantities, different reference coordinate systems can be chosen for the initial and current configurations. Of course, in general, whether it is the initial or current configuration, the fixed and unchanging global rectangular coordinate system O in space is used as the reference. At this time, any material point a under C0 is denoted as X in this global coordinate system O, because the material point will not disappear arbitrarily, and at time t, under the configuration C1, it will necessarily move to a certain position A, and the coordinate of this point in the same global coordinate system is x.



在数学上，X和x就相当于两种自变量，对于参考坐标系O中任意的变量y，就有两种表示方法。

In mathematics, X and x are equivalent to two independent variables. For any variable y in the reference coordinate system O, there are two representation methods.

1.2.1 Lagrangian描述 1.2.1 Lagrangian Description

第一种：表示为X的函数：y=f(X)。此时我们称为Lagrangian（Lagrangian英文后面统一简称为L）坐标描述方法。典型物理量：Green应变。

The first method: expressed as a function of X: y=f(X). At this time, it is called the Lagrangian (the English abbreviation for Lagrangian is L) coordinate description method. Typical physical quantity: Green strain.

- 注：Green应变也称为Green-Lagrange 应变，猜测可能就是因为是Lagrange描述的原因。

  Note: Green strain is also known as Green-Lagrange strain, and it is probably due to the Lagrange description.

1.2.2 Eulerian描述 1.2.2 Eulerian Description

第二种：表示为x的函数：y=f(x)。此时我们称为Eulerian（Eulerian英文名后面统一简称为E）坐标描述方法。典型物理量：真实应变。

The second method: expressed as a function of x: y=f(x). At this point, we call it the Eulerian (the English name of Eulerian is abbreviated as E) coordinate description method. Typical physical quantity: true strain.

1.2.3 两种描述分析 1.2.3 Two Descriptions of Analysis

这两者表述的是同一个模型，举个简单的例子，譬如材料点随时间t的运动为x=2*X*t。那么在坐标系O下，在t时刻的速度的Lagrangian和Eulerian描述方法分别为：

These two descriptions refer to the same model. For example, let's take the motion of a material point over time t as x = 2 * X * t. Then, at time t in the coordinate system O, the Lagrangian and Eulerian descriptions of the velocity are:

v(X)=2*X和v(x)=2*x/2t=x/t

v(X) = 2*X and v(x) = 2*x/(2t) = x/t

这两种描述方法都是相对同一坐标系O的物理量的表示方法，那么得到的数值结果完全一致，只不过写成函数的时候自变量不一致一样。

Both of these description methods are representations of physical quantities relative to the same coordinate system O, and the numerical results obtained are completely consistent; the only difference is that the independent variables are written differently in the function.

1.3 网格的Lagrangian和Eulerian描述

1.3 Lagrangian and Eulerian Descriptions of Meshes

在讲T.L.和U.L.之前，我们先解释一下L.代表的Lagrangian网格的含义。前面所述物理量的坐标描述也有L和E之分，网格也有L和E之分。这两者是不同的概念。在L和E网格定义中，表示的是两种网格随时间变化的位置定位方式，而物理量的坐标描述仅仅是变量的表示方式，这两者没有直接关系。为了说明L和E网格的区别，举一个简单的例子，譬如二维平面的运动如下：x=X+tY, y=Y，那么一个长方形运动后的变形为一个平行四边形，具体的材料点a(在全局坐标系O中坐标为X，Y)变形后为A（在O中坐标为x，y）。

Before discussing T.L. and U.L., let's first explain the meaning of L. in Lagrangian mesh. The coordinate descriptions of the physical quantities mentioned earlier also have L and E, and the mesh also has L and E. These are different concepts. In the definitions of L and E meshes, it represents two different ways of positioning the meshes as they change over time, while the coordinate descriptions of physical quantities are merely the representation of variables, and there is no direct relationship between the two. To illustrate the difference between L and E meshes, let's take a simple example, such as the motion of a two-dimensional plane: x = X + tY, y = Y. The deformation of a rectangle after motion becomes a parallelogram, and the specific material point a (with coordinates X, Y in the global coordinate system O) deforms to A (with coordinates x, y in O).

网格主要由节点组成，而节点在全局空间O中的坐标决定了网格的形状。节点的坐标有两种取法：

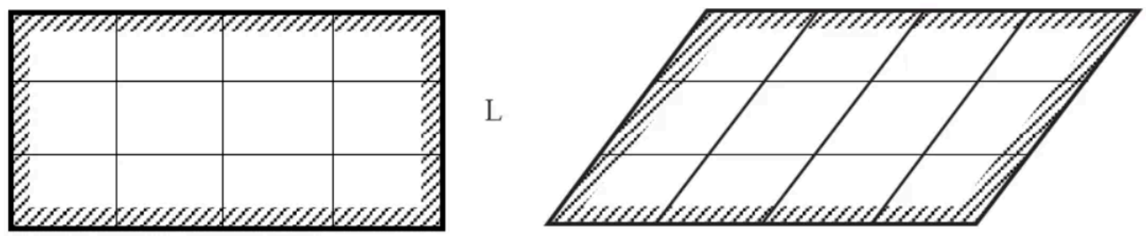
The mesh is mainly composed of nodes, and the coordinates of the nodes in the global space O determine the shape of the mesh. There are two ways to take the coordinates of the nodes:

1.3.1 网格的Lagrangian描述 1.3.1 Lagrangian Description of Mesh

无论在哪个时刻，取同一个材料点变形后的xi，yi坐标，此时如果一开始网格点在a点，那么变形后必然在A点，而且，如果用L描述，那么可以得到，Xi，Yi=常数。那么得到的网格将类似下方，随构型变形而自动更新变化。这

就是Lagrangian网格。

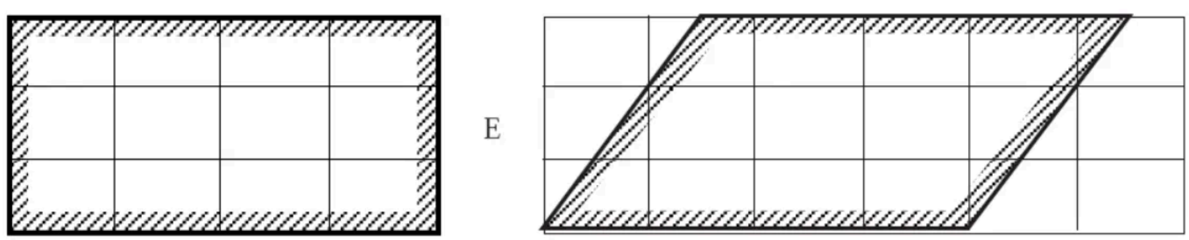
At any given moment, if the same material point's xi, yi coordinates after deformation are taken, then if the grid point was initially at point a, it will necessarily be at point A after deformation. Moreover, if described using the Lagrangian method, it can be obtained that Xi, Yi = constant. The resulting mesh will be similar to the one below, automatically updating and changing with the deformation of the structure. This is the Lagrangian mesh.



1.3.2 网格的Eulerian描述 1.3.2 Eulerian Description of Mesh

无论在哪个时刻，取同一个材料点变形前的Xi，Yi坐标，此时如果一开始网格点在a点，那么变形后还是在a点。那么得到的网格将类似下方，固定在空间不变。这就是Eulerian网格。

At any given moment, if the Xi, Yi coordinates before deformation of the same material point are taken, then if the grid point was initially at point a, it will still be at point a after deformation. The resulting mesh will be similar to the one below, fixed in space and unchanging. This is the Eulerian mesh.



结构有限元中一般都是Lagrangian网格，这也是为何T.L.和U.L.方法称为Lagrangian方法的原因。

In structural finite element analysis, Lagrangian meshes are generally used, which is why the T.L. and U.L. methods are called Lagrangian methods.

1.4 虚功原理的T.L.和U.L.描述方式 1.4 Description of the Virtual Work Principle using T.L. and U.L.

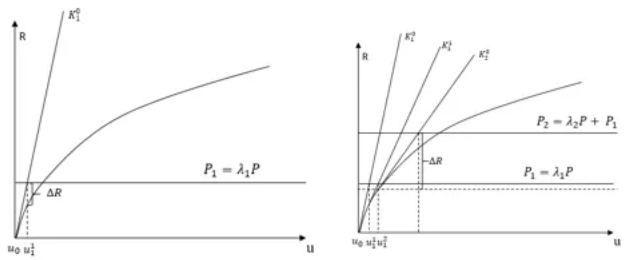
有限元中，T.L.和U.L.用于非线性问题中的虚功原理的描述方式，是整个有限元求解方程的描述方式，而前面物理量的描述方式是虚功原理中各个量的描述方式。

In finite element analysis, T.L. and U.L. are used to describe the virtual work principle in nonlinear problems, which is the description of the entire finite element solution equation. The description of physical quantities before this is the description of each quantity in the virtual work principle.

非线性问题在有限元中都是以增量法来求解，增量法的基本思想就是将施加外载荷的过程分割为若干个时间增量步，假定这个时间增量步是固定的，dt,2dt,3dt,….，每一步只施加一个比较小的载荷增量，总载荷引起的位移就是

所有增量步位移增量的累加。

Nonlinear problems in finite element analysis are solved using the incremental method, which essentially involves dividing the process of applying external loads into several time increments, assuming that each time increment is fixed, dt, 2dt, 3dt, etc., with only a small load increment applied at each step. The total displacement caused by the total load is the sum of the displacement increments in all the increments.



具体的理论和Abaqus实现过程可参考我们以前**系列文章**4：**非线性问题的求解。**

The specific theories and Abaqus implementation process can be referred to in our previous series article 4: Solution of Nonlinear Problems.

由虚功原理在(n+1)dt时刻应变能为： By the principle of virtual work, the strain energy at the moment (n+1)dt is:

$$W = \int_V \sigma * \delta\varepsilon \, dV$$

σ和ε分别表示 Cauchy 应力（真实应力）和真实应变。

上面每个量都是都是参考(n+1)dt时刻的坐标的，此时将相对参考时刻的坐标写在左下方，将要求的时刻点写在左上方，那么就是：

Each of these quantities refers to the coordinates at the (n+1)dt moment, at this time, the coordinates relative to the reference moment are written in the lower left corner, and the required moment is written in the upper left corner, so that:

$$ {}^{(n+1)dt}_{(n+1)dt}W = \int_{{}^{(n+1)dt}_{(n+1)dt}V} {}^{(n+1)dt}_{(n+1)dt}\sigma * {}^{(n+1)dt}_{(n+1)dt}\delta\varepsilon \, d{}^{(n+1)dt}_{(n+1)dt}V $$

${}^{(n+1)dt}_{(n+1)dt}\sigma$表示(n+1)dt 时刻的应力用(n+1)dt 时刻的坐标表示，其它类似，但现在(n+1)dt 时刻坐标本身就是待求项，所以，此时所有量都无法描述，更不要说求解方程了。只能换一种量的描述方式，发现前面时刻 dt,…n*dt 时刻的坐标已经求出，时刻(n+1)dt 的未知量描述用他们来描述。为描述物体在(n+1)dt 时刻变形和运动，理论来说可以选择任何一个特定时刻的构型作为参考构型，以确定每一时刻每一质点的物理量，但实际中为方便起见，这个参考构型一般选择以下两种：

1.4.1 T.L.描述 1.4.1 T.L. Description

以初始构型为参考构型，也就是用 Lagrangian 坐标 X 描述物理量，那么就是 T.L. 方法（Total Lagranian 方法）

$$ {}^{(n+1)dt}_{0}W = \int_{{}^{0}_{0}V} {}^{(n+1)dt}_{0}\sigma * {}^{(n+1)dt}_{0}\delta\varepsilon * {}^{(n+1)dt}_{0}J * d{}^{0}_{0}V $$

其中

a.体积${}^{0}_{0}V$其实就是初始构型的体积积分，方便起见，可以直接写为 V0 注意不再是(n+1)dt 时刻的体积了。可以类比普通的函数积分一样：譬如[0,1]区间（类比体积）的积分

$$ y = \int_0^1 x \, dx $$

当取 X=x/2 时，积分变为：

$$ y = \int_0^{0.5} X/4 \, dX $$

区间变为了[0,0.5]，也就是初始构型的体积。

b.J 为体积微分之分，和积分点位置有关。

$$ {}^{(n+1)dt}_{0}J = dV/dV0 $$

c.Cauchy 应力和变形率相对初始时刻的值实际求解相对困难，为了克服这个问题，分别变换到第二 P.K. 应力和 Green 应变，具体的变换关系详见其它理论文档。变为：

$$ {}^{(n+1)dt}_{0}W = \int_{{}^{0}_{0}V} {}^{(n+1)dt}_{0}S * {}^{(n+1)dt}_{0}\delta\varepsilon^G \, d{}^{0}_{0}V $$

- 注意，有些读者认为T.L.只能用于小应变，但这种说法个人认为并不正确，这里的推导没有涉及是否是大应变还是小应变，同时，后面文章我们将证明U.L.和T.L描述是可以等价转换的，既然U.L.可用于大应变，那么T.L.也适用于大应变、大转动、大位移情况。

  Note that some readers believe that T.L. can only be used for small strains, but this statement is not correct in my opinion. The derivation does not involve whether it is large or small strain, and in the following article, we will prove that U.L. and T.L. descriptions can be equivalently converted. Since U.L. can be used for large strains, T.L. is also applicable to large strains, large rotations, and large displacements.

1.4.2 U.L.描述 1.4.2 U.L. Description

以当前时刻ndt构型为参考构型，用Eulerian坐标x描述物理量，那么就是U.L.方法（Updated Lagrangian方法）

Taking the current ndt configuration as the reference configuration, using Eulerian coordinates x to describe physical quantities, it is the U.L. method (Updated Lagrangian method).

$$ {}^{(n+1)dt}_{ndt}W = \int_{{}^{ndt}_{ndt}V} {}^{(n+1)dt}_{ndt}\sigma * {}^{(n+1)dt}_{ndt}\delta\varepsilon * {}^{(n+1)dt}_{ndt}J d^{ndt}_{ndt}V $$

- 上式的转换数学上并不严格，还差一个应变对时间的导数，严格推导和系列文章22：几何非线性的刚度矩阵求解类似，需要将上述表达式先做线性化。

  The transformation of the above formula is not mathematically rigorous, it is missing a derivative of strain with respect to time. A strict derivation is similar to the series article 22: solving the stiffness matrix of geometric nonlinearity, and the above expression needs to be linearized first.

1.5 Abaqus的内部描述方式验证方法 1.5 Verification methods for Abaqus' internal description method

在Abaqus所有的帮助手册中，找不到任何T.L.还是U.L.的信息，由上节所述，我们只能反过来，通过查看Abaqus的变量的度量方式，来猜测Abaqus用了T.L.还是U.L.方式，Abaqus文档中多处提到了各种变量的度量，但并不一定真实，而且前后明显矛盾，最好的方式还是直接查Abaqus程序的结果。

In all the help manuals of Abaqus, there is no information about T.L. or U.L. By the above section, we can only guess whether Abaqus uses T.L. or U.L. by looking at the measurement methods of Abaqus' variables. The Abaqus documentation mentions the measurement of various variables, but it is not necessarily true, and there are obvious contradictions. The best way is still to directly check the results of the Abaqus program.

注意，此处不一定能采用后处理模块中显示的应变量。为了和试验对应，Abaqus几何非线性后处理显示是自然应变和Cauchy应力，它将程序中的相关度量在显式时做了转换，与程序中的真正用于计算的度量并不一致。详细说明可以查看**本系列文章27： Abaqus内部计算和显示的应变。**

Note that it may not be possible to use the strain values displayed in the post-processing module. To correspond with the experiment, the Abaqus geometric nonlinear post-processing displays natural strain and Cauchy stress, which converts the relevant measures in the program at the time of display, and they are not consistent with the actual measures used for calculation. Detailed explanations can be found in Series Article 27: Abaqus Internal Calculation and Display of Strain.

我们通过UMAT等子程序来查看Abaqus子程序接口中的应变应力度量，同时和iSolver采用同样度量的结果比对，大体猜测Abaqus几何非线性的描述方式。

We can view the strain and stress measures in the Abaqus subroutine interface through UMAT and other subroutines, and compare them with the results of iSolver using the same measures to roughly guess the description method of Abaqus geometric nonlinearity.

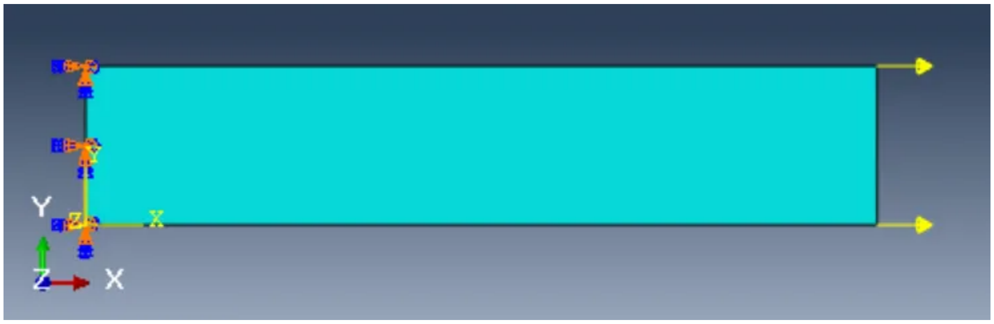
1.5.1 Abaqus中T.L.的描述 1.5.1 Description of T.L. in Abaqus

Abaqus中S4R5、STRI3、STR65、S4RS、S8R5、S3RS、B33，B23等单元都采用T.L.的描述方式。采用一个简单的算例，证明如下：

The Abaqus elements S4R5, STRI3, STR65, S4RS, S8R5, S3RS, B33, B23, etc., all use the T.L. description method. A simple example is used to demonstrate as follows:

有限元理论基础及Abaqus内部实现方式研究系列28： 几何非线性的T.L.和U.L.描述方法的图31

1.5.1.1 算例介绍 1.5.1.1 Example Introduction



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.

左侧两个节点固支。 The two nodes on the left are fixed.

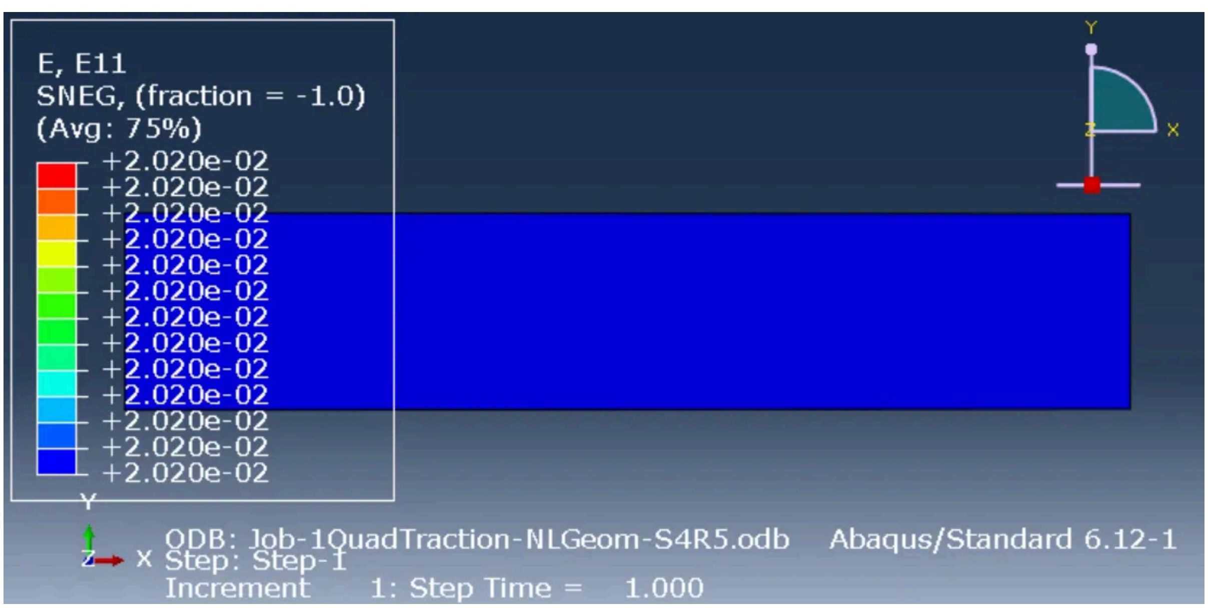
右侧两个节点每个加集中力1e5， x方向。 Each of the two nodes on the right is subjected to a concentrated force of 1e5 in the x-direction.

划分为一个单元。 Divided into a single element.

1.5.1.2 Abaqus应变结果 1.5.1.2 Abaqus Strain Results

Abaqus中采用S4R5单元，几何非线性单元NLGeom打开，得到一个迭代步后结果：

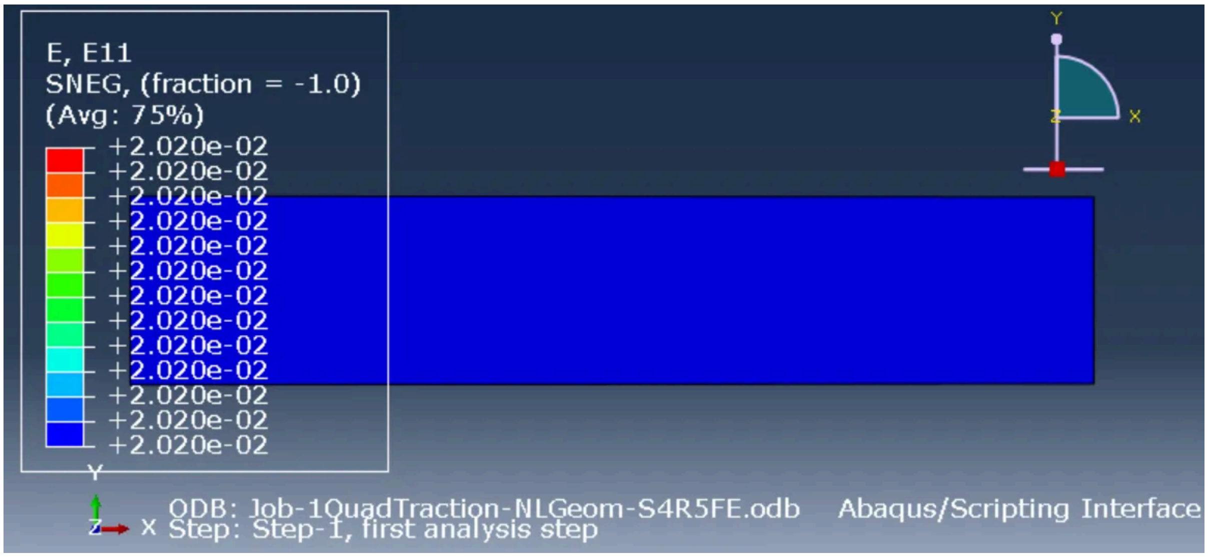
In Abaqus, using the S4R5 element, the geometric nonlinear element NLGeom is turned on, and after one iteration step, the results are obtained:



1.5.1.3 iSolver应变结果 1.5.1.3 iSolver strain results

iSolver中也采用S4R5的度量方式，同时采用T.L.方式，得到的应变度量如下，可发现，和Abaqus完全一致。

In iSolver, the same S4R5 measurement method is also used, and the T.L. method is adopted as well. The strain measurement obtained is as follows, which can be found to be completely consistent with Abaqus.



1.5.2 Abaqus中U.L.的描述 1.5.2 Description of U.L. in Abaqus

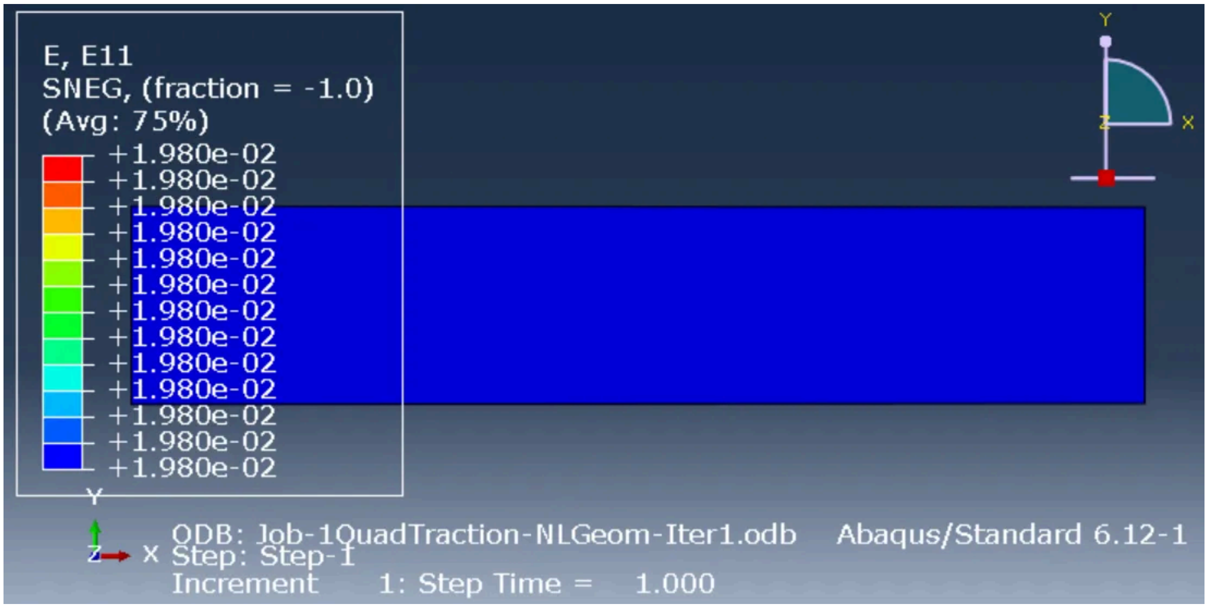
Abaqus中C3D8/C3D8R、S4/S4R、S3/S3R、B31，B21等非线性单元都采用T.L.的描述方式。采用一个上述同样的算例，证明如下：

Nonlinear elements such as C3D8/C3D8R, S4/S4R, S3/S3R, B31, and B21 in Abaqus all use the T.L. description method. The following example demonstrates this using the same case:

1.5.2.1 Abaqus应变结果 1.5.2.1 Abaqus strain results

几何非线性开关NLGeom=On，同时单元类型改为S4R。得到应变：

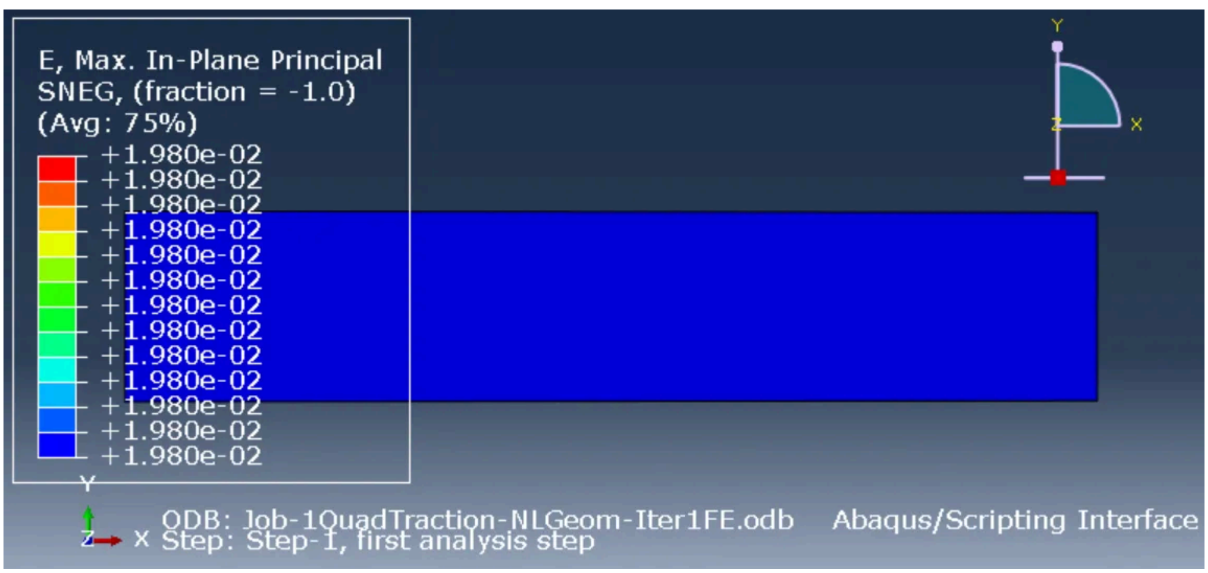
With NLGeom=On for geometric nonlinearity and the element type changed to S4R, the strain is obtained:



1.5.2.2 iSolver应变结果 1.5.2.2 iSolver strain results

iSolver中也采用S4R的度量方式，同时采用U.L.方式，得到的应变度量如下，可发现，和Abaqus完全一致。

iSolver also adopts the S4R measurement method, and uses the U.L. method as well. The strain measurement obtained is as follows, and it can be found to be completely consistent with Abaqus.



1.6 T.L.和U.L.描述方式的本质 The essence of the T.L. and U.L. description methods

既然Bathe教授在开辟几何非线性时就创立了T.L.和U.L.两种虚功原理的描述方式，而且有限元商软也是按这两种方式实现的，那Abaqus中为何从头到尾没提T.L.还是U.L.的描述方式呢？

Since Professor Bathe established the two virtual work principle description methods of T.L. and U.L. when pioneering geometric nonlinearity, and finite element commercial software also implements them in these two ways, why does Abaqus never mention the T.L. or U.L. description methods from beginning to end?

我们猜测这是Abaqus有意为之。这两种虚功原理的描述本来就是描述的同一个物理对象，所以理论上你采取任何一种都可以得到最终的正确解，但无论哪种描述都需要它用到物理量的描述方式结合起来满足一定的要求，譬如功的共轭、能量守恒、动量守恒等条件。同时，因为有限元中物理量的描述方式可以相互转换的，为了有限元的结果更能和实际试验的度量对应，Abaqus也会在T.L.中采用Euler描述的物理量的度量方式或者U.L.中采用Lagrangian描述的物理量的度量方式，譬如它本身提供的UMAT是真实应变和真实应力，但你完全可以不用管UMAT输入的DSTRAN等应变值，而用它提供的另一个参数变形梯度DFGRD1按Lagrangian描述方式实现Green应变（当然，这样得到的Jacobian矩阵DDSDDE和应力STRESS都没问题，但你改不了Abaqus内部利用STRESS来计算初始应力刚度和本身的材料刚度矩阵的算法，所以还需要Abaqus后面提供更多的接口才行），或者在UEL采用Green应变和2nd PK力来实现刚度阵AMATRX和内力RHS，也就是说已经突破了传统的T.L.和U.L.的界限，所以干脆Abaqus就不再提T.L.还是U.L.方式了。

We speculate that this is intentional on the part of Abaqus. The descriptions of these two virtual work principles are inherently describing the same physical object, so theoretically, you can use either one to obtain the final correct solution. However, regardless of which description is used, it needs to combine the description method of physical quantities to meet certain requirements, such as the conjugate of work, conservation of energy, and conservation of momentum. At the same time, because the description methods of physical quantities in finite elements can be interconverted, in order for the results of finite elements to better correspond to the measurements of actual experiments, Abaqus will also use the measurement method of physical quantities described by Euler in T.L. or the measurement method of physical quantities described by Lagrangian in U.L., for example, the UMAT it provides itself is the true strain and true stress, but you can completely ignore the strain values such as DSTRAN input by UMAT, and use another parameter deformation gradient DFGRD1 to implement Green strain in the Lagrangian description method (of course, the Jacobian matrix DDSDDE and stress STRESS obtained in this way are fine, but you cannot change the algorithm used by Abaqus internally to calculate the initial stiffness and the material stiffness matrix itself, so more interfaces need to be provided by Abaqus later), or in UEL, use Green strain and 2nd PK force to realize the stiffness matrix AMATRX and internal force RHS, which means that the traditional boundaries of T.L. and U.L. have been broken through. Therefore, Abaqus simply no longer mentions the T.L. or U.L. method.

```
SUBROUTINE UMAT(STRESS,STATEV,DDSDDE,SSE,SPD,SCD,
     1 RPL,DDSDDT,DRPLDE,DRPLDT,
     2 STRAN,DSTRAN,TIME,DTIME,TEMP,DTEMP,PREDEF,DPRED,CMNAME,
     3 NDI,NSHR,NTENS,NSTATV,PROPS,NPROPS,COORDS,DROT,PNEWDT,
     4 CELENT,DFGRD0,DFGRD1,NOEL,NPT,LAYER,KSPT,KSTEP,KINC)
C
      INCLUDE 'ABA_PARAM.INC'
C
      CHARACTER*80 CMNAME
      DIMENSION STRESS(NTENS),STATEV(NSTATV),
     1 DDSDDE(NTENS,NTENS),DDSDDT(NTENS),DRPLDE(NTENS),
     2 STRAN(NTENS),DSTRAN(NTENS),TIME(2),PREDEF(1),DPRED(1),
     3 PROPS(NPROPS),COORDS(3),DROT(3,3),DFGRD0(3,3),DFGRD1(3,3)


      user coding to define DDSDDE, STRESS, STATEV, SSE, SPD, SCD
      and, if necessary, RPL, DDSDDT, DRPLDE, DRPLDT, PNEWDT


      RETURN
      END
```

==总结== ==Summary==

本文共介绍了三种近似相关的描述方式，Abaqus和iSolver的对应实现方式分别如下：

This article introduces three approximation-related description methods, and the corresponding implementation methods in Abaqus and iSolver are as follows:

项次	描述方式	Abaqus 实现	iSolver 实现
1.物理量	Lagranian	Green 应变等采用该描述	和 Abaqus 一致
	Eulerian	真实应变等采用该描述	和 Abaqus 一致
2.网格	Lagranian	一般情况都是该描述	都是该描述
	Eulerian	ALE、CEL 等采用 L 和 E 结合的网格	暂无 ALE、CEL 功能
3.虚功原理	T.L.	S4R5、STRI3、STR65、S4RS、S8R5、S3RS、B33，B23 等	只实现 Abaqus 部分单元
	U.L.	C3D8/C3D8R、S4/S4R、S3/S3R、B31，B21 等	只实现 Abaqus 部分单元

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

有限元理论基础及Abaqus内部实现方式研究系列28： 几何非线性的T.L.和U.L.描述方法的图45

有限元理论基础及Abaqus内部实现方式研究系列28：几何非线性的T.L.和U.L.描述方法的图46

有限元理论基础及Abaqus内部实现方式研究系列28：几何非线性的T.L.和U.L.描述方法的图47

以往的系列文章： Previous series articles:

有限元理论基础及Abaqus内部实现方式研究系列28：几何非线性的T.L.和U.L.描述方法的图48

有限元理论基础及Abaqus内部实现方式研究系列28：几何非线性的T.L.和U.L.描述方法的图49

1.8.1 ========第一阶段========

1.8.1 ======== Phase One ========

第一篇：**S4壳单元刚度矩阵研究。** First article: Research on the stiffness matrix of S4 shell elements.

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

第二篇：**S4壳单元质量矩阵研究。** The Second Article: Research on the Mass Matrix of S4 Shell 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。** The Third Article: Shear Locking and Shear Band Control of S4 Shell Elements.

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

第四篇：**非线性问题的求解。** The Fourth Article: Solution of Nonlinear Problems.

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

第五篇：**单元正确性验证。** The Fifth Article: Element Accuracy Verification.

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

第六篇：**General梁单元的刚度矩阵。** Sixth Article: Stiffness Matrix of General Beam Elements.

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

第七篇：**C3D8六面体单元的刚度矩阵。** Seventh Article: Stiffness Matrix of C3D8 Hexahedral Elements.

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

第八篇：**UMAT用户子程序开发步骤。** Eighth Article: Steps for Developing UMAT User Subroutines.

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

第九篇：**编写线性UMAT Step By Step。**

Ninth Article: Writing Linear UMAT Step By Step.

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

第十篇：**耦合约束（Coupling constraints）的研究。**

The tenth article: Research on coupling constraints.

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

有限元理论基础及Abaqus内部实现方式研究系列28：几何非线性的T.L.和U.L.描述方法的图50

有限元理论基础及Abaqus内部实现方式研究系列28：几何非线性的T.L.和U.L.描述方法的图51

1.8.2 ========第二阶段========

1.8.2 ========Second Phase========

第十一篇：**自主CAE开发实战经验第一阶段总结。** The eleventh article: Summary of the first phase of independent CAE development experience.

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

第十二篇：**几何梁单元的刚度矩阵。** The twelfth article: Stiffness matrix of the geometric beam element.

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

第十三篇：**显式和隐式的区别。** The Thirteenth Article: The Difference Between Explicit and Implicit.

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

第十四篇：**壳的应力方向。** The Fourteenth Article: Stress Direction of Shells.

https://www.jishulink.com/content/post/1189260

第十五篇：**壳的剪切应力。** The Fifteenth Article: Shear Stress of Shells.

https://www.jishulink.com/content/post/1191641

第十六篇：**Part、Instance与Assembly。**

Chapter 16: Part, Instance, and Assembly.

https://www.jishulink.com/content/post/1195061

第十七篇：**几何非线性的物理含义。** The 17th article: The physical meaning of geometric nonlinearity.

https://www.jishulink.com/content/post/1198459

第十八篇：**几何非线性的应变。** The 18th article: Strain of geometric nonlinearity.

https://www.jishulink.com/content/post/1201375

第十九篇：**Abaqus几何非线性的设置和后台。** The 19th article: Settings and background of Abaqus geometric nonlinearity.

http://www.jishulink.com/content/post/1203064

第二十篇：**UEL用户子程序开发步骤。** The 20th article: Steps for developing UEL user subroutines.

https://www.jishulink.com/content/post/1204261

有限元理论基础及Abaqus内部实现方式研究系列28：几何非线性的T.L.和U.L.描述方法的图52
有限元理论基础及Abaqus内部实现方式研究系列28：几何非线性的T.L.和U.L.描述方法的图53

1.8.3 ========第三阶段========

1.8.3 ========Third Phase========

第二十一篇：**自主CAE开发实战经验第二阶段总结。** Chapter 21: Summary of the Second Stage of Autonomous CAE Development Practical Experience.

https://www.jishulink.com/content/post/1204970

第二十二篇：**几何非线性的刚度矩阵求解。** The 22nd article: Solution of the stiffness matrix for geometric nonlinearity.

http://www.jishulink.com/content/post/1254435

第二十三篇：

有限元理论基础及Abaqus内部实现方式研究系列28：几何非线性的T.L.和U.L.描述方法的图54

编写简单面内拉伸问题UEL Step By Step。

The 23rd article: Step By Step guide to writing a simple in-plane tensile problem UEL.

http://www.jishulink.com/content/post/1256835

第二十四篇：

有限元理论基础及Abaqus内部实现方式研究系列28：几何非线性的T.L.和U.L.描述方法的图55

显式求解Step By Step。 The 24th article: Step By Step explicit solution.

https://www.jishulink.com/content/post/1261165

第二十五篇：**显式分析的稳定时间增量。** The 25th article: Stability time increment for explicit analysis.

http://www.jishulink.com/content/post/1263601

第二十六篇：**编写线性VUMAT Step By Step。**

第二十六篇：Step by Step Guide to Writing Linear VUMAT.

https://www.jishulink.com/content/post/1266640

第二十七篇：**Abaqus内部计算和显示的应变。** 第二十七篇：Strain Calculation and Display in Abaqus Internal.

https://www.jishulink.com/content/post/1273788