

基于 VTK 的 CFD 可视化研究

李信凯,王 帅,封卫兵

(上海大学 计算机工程与科学学院,上海 200444)

摘 要:科学计算可视化自提出至今已在许多研究领域得到了广泛应用,计算流体力学(CFD)可视化正是其中之一。文中首先简要介绍了可视化工具库(VTK)及其流水线结构,之后介绍了计算流体力学软件的基本架构。在此基础上,以均匀流场网格为例,讨论了使用 VTK 进行流体可视化开发的通用流程。最后在团队开发的基于 LBM 方法的 CFD 软件基础上,实现可视化模块,包括基本标量属性(压力、温度等)的颜色映射,流体网格的内部切面以及流线方式的速度矢量可视化。同时展示了以上可视化方法的模拟结果。可视化模块与前处理和求解模块共同组成整个 CFD 软件,并通过图形结果直观地展示流场内部粒子的属性和运动规律。

关键词:计算流体力学;流体可视化;CFD 软件;可视化工具库

中图分类号:TP39

文献标识码:A

文章编号:1673-629X(2015)10-0029-05

doi:10.3969/j.issn.1673-629X.2015.10.006

Study of CFD Visualization Based on VTK

LI Xin-kai, WANG Shuai, FENG Wei-bing

(School of Computer Engineering and Science, Shanghai University, Shanghai 200444, China)

Abstract: Visualization in scientific computing has been widely used in many fields of scientific research since it was put forward, the Computational Fluid Dynamics (CFD) visualization is just one of them. In this paper, firstly present a brief introduction to the Visualization ToolKit (VTK) and the architecture of its pipeline, basic framework in CFD software. After what is discussed in the sections above, describe the common process of CFD visualization using VTK. Finally, add the visualization functions on the CFD software developed by the team based on LBM method. Visualization functions including color mapping of scalar properties such as pressure, temperature, etc, cut surface in the fluid grid and vector properties of velocity in the manner of streamline are realized. Visualization module constitutes the entire CFD software with pre-propose and resolver module, and shows the properties and law of motion of particles in the fluid field intuitively.

Key words: computational fluid dynamics; CFD visualization; CFD software; VTK

0 引 言

科学计算可视化(Visualization in Scientific Computing)的概念提出于 20 世纪 80 年代末,是运用计算机图形学、信息处理技术创建视觉图像,帮助人们理解科学技术概念或结果的那些错综复杂而又往往规模庞大的数字表现^[1]。其在经历近 30 年的发展中已应用到科学计算的各个领域,如天体物理学、地球和环境科学、流体力学、生物分子及医学等重要科学分支。可视化在科学计算中扮演的角色也越来越重要,成为其不可或缺的一部分并极大地推动了科学技术的发展。

伴随着可视化技术的发展,产生了众多的可视化软件和工具库,如第一个商业可视化软件 AVS^[2],IBM

公司开发的 Data Explorer^[3],美国 Tecplot 公司的 Tecplot 系列可视化软件^[4]以及开源可视化工具库(Visualization ToolKit, VTK)^[5]。这些工具库使科学可视化应用领域屏蔽底层的计算机图形相关技术,直接调用工具库提供的库函数生成可视化结果,大大降低了科学可视化应用开发的难度。其中, VTK 以开源、跨平台、支持并行处理等特性得到了广泛应用。VTK 以面向对象为基础,核心库为 C++ 语言实现并支持 Tcl/Tk、Python、Java 扩展,丰富了可视化应用的开发。

计算流体力学(Computational Fluid Dynamics, CFD)是一门数学、流体力学和计算机结合的交叉学科。通过求解流场中的基本方程,如 N-S 方程、Euler

收稿日期:2014-11-19

修回日期:2015-02-26

网络出版时间:2015-09-23

基金项目:国家自然科学基金重大研究计划项目(91330116)

作者简介:李信凯(1987-),男,硕士,研究方向为高性能计算与可视化;封卫兵,副教授,研究方向为高性能计算。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20150923.1506.054.html>

方程,对流体力学问题进行模拟和分析^[6-7]。文中主要就以 VTK 为开发工具,对团队开发的 LBM 软件系统生成的数据源进行可视化后处理。

可视化主要流程为:

(1)应用 VTK 提供的对象将待处理数据转换为 VTK 可处理的数据集,并建立起数据集的单元结构及属性(压力、温度等标量属性和速度矢量属性)。

(2)创建过滤器对象。

(3)把经过过滤器处理后得到的数据集和映射器对象连接创建 VTK 的可视化模型。

(4)创建图形模型,完成整个可视化过程。

1 VTK 简介

VTK 拥有一整套强大的流水线绘制流程,如图 1 所示。

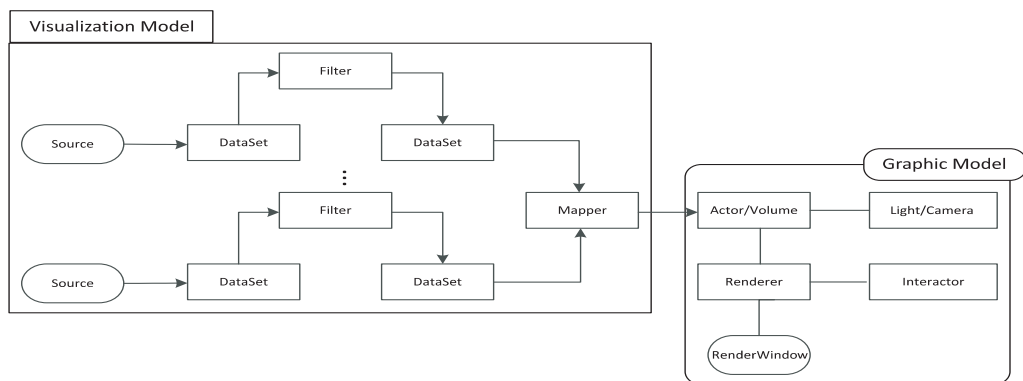


图 1 VTK 可视化流水线

流水线的优势在于各个模块(数据对象、过滤器对象、映射器等)独立开来,可视化流程功能清晰易用,并且可以更好地支持并行处理^[8]。VTK 流水线主要包括可视化模型和图形模型两种。下面简单介绍一下两种模型。

1.1 可视化模型

可视化模型基于许多商用软件所采用的数据流模式^[9]。该模式下,各模块连接成一个网络。在数据流经网络的过程中,各模块执行相应的算法操作。整个流程执行的结果就是从程序生成或文件读取的大量待可视化的元数据转换成由 VTK 数据集(vtkDataSet)对象表示的结构。

数据集、属性数据及过滤器是可视化模型的基本模块。数据集将离散点及属性组织成有空间拓扑结构的集合。常见的数据集类型包括均匀网(vtkImageData)、结构网格(vtkStructuredGrid)、无结构网格(vtkUnstructuredGrid)、多边形(vtkPolyData)等;属性数据包括标量(Scalar)、向量(Vector)、张量(Tensor)和纹理坐标(Texture coordinate)等。属性数据通过 VTK 中的点(vtkPoints)和单元(vtkCell)建立起数据集的基本结构。过滤器是对建立的基本数据集做二次处理的处理对象。过滤器对象接收一个或多个数据集作为输入,处理完成后输出一个结果数据集。常用的可视化方法如切割、轮廓提取、等值线和等值面都是过滤器算法实现的。

1.2 图形模型

VTK 的图形模型将 OpenGL 图形库进行封装,提

供更高层的 API,可以方便地获取 3D 图形的基本特征。图形模型采用目前最常用的 GUI 显示方法,同时支持 MFC、QT 等多种界面窗口。该模型包含 2D 几何、文字、图像绘制、2D 纹理映射、3D 光线投射映射等多种图形显示算法。图形模型由角色(vtkActor)、相机(vtkCamera)、灯光(Light)、属性(vtkProperty)、映射器(vtkMapper)、渲染器(vtkRenderer)、渲染窗口(vtkRenderWindow)等 9 种图形渲染类^[10]。

2 CFD 软件结构

作为传统流体力学与计算机结合的一门交叉学科,借助于计算机技术及软件工程的飞速发展,CFD 软件系统及方法也逐渐成为一大热门研究方向。

CFD 软件一般包含三大模块:前处理、求解器和后处理^[11-12]。

前处理模块主要是向 CFD 软件输入求解问题的计算域和属性、初始状态参数,划分网格等工作。前处理往往是最耗费计算资源的部分,所以要把握好计算精度与计算成本之间的平衡。

求解器是指整个系统的求解方案,根据前处理过程所生成的模型网格、所选的数值算法、边界条件等进行迭代求解,并输出计算结果。

后处理即可视化,是指经过前处理和求解器阶段生成的网格点属性数据(温度、压力、速度等物理量)转化为图形图像、动画等直观的表现形式。

目前市场上已经存在多种商用 CFD 软件,其中占据市场主流的有 Fluent、CFX、Tecplot 等。这些商用软

件都提供一整套的 CFD 解决方案,从前处理到最后的标量、矢量、等值线、流场的可视化实现。目前国内 CFD 软件的研究也进入高速发展阶段。而 CFD 软件的研发面临着诸多困难,软件通用性和效率是其首要待解决的问题,这就需要计算流体力学和计算机技术人才的联合培养,两大科学分支技术的深入交叉研究,开发出高质量、高可用性的 CFD 软件系统。

3 可视化系统及结果展示

3.1 可视化模块构建步骤

在不考虑用户界面的情况下,可视化实现的首要步骤就是根据具体应用构建可视化模型。不同的可视化模块基于相似的流水线结构,抽取公共方法步骤,提高复用率。首先根据求解器产生的属性数据格式选择最适宜数据集对象。建立顶点集、单元等拓扑结构,并将网格点属性与单元属性建立起映射关系。然后根据想要的二次处理算法,创建相应的过滤器,如:获取特定切面、等值面分级着色、提取单元子集等算法。其次调用映射器对象将过滤器处理后的映射到 Actor 对象,这样就建立了完整的可视化模型。最后建立图形模型,完成整个可视化流水线。

3.2 具体实现方法

(1) 选择合适的数据集,建立拓扑结构。如生成的网格数据的拓扑结构为均匀的立方体,则采用 `vtkImageData` 对象较为合适;另外该网格还广泛应用于医学、工程图像的读入数据集。如网格不是标准立方体,但结构遵循一定的规则并给出了网格的所有顶点坐标,应使用 `vtkStructuredGrid` 对象。网格点空间分布不遵循明显的空间规律,需要采用 `vtkUnstructuredGrid` 对象。`vtkPolyData` 适用于包括三角型在内的多边形网格数据,是一种空间面网格结构,而前三种属于体网格。

文中基于团队开发的 CFD 软件进行可视化开发,以流场产生的均匀网格为例进行演示说明。

首先创建 `vtkImageData` 对象,并分别设置网格的起始点,三维方向上的网格尺寸和网格点数。再创建一个属性数组与每个单元的数据属性建立映射。文中处理的是网格中心点坐标和相关属性,这种方法可以大量缩减前处理和求解过程的计算量和生成的数据量。但 `vtkImageData` 等对象没有直接通过网格中心坐标建立数据集的方法,因此需要特殊的算法进行处理。处理方法如下代码所示:

```
//获取起始网格中心坐标
```

```
tempPoint = Multiple_Lattice. GetMultipleLatticeCoordinate (0, 0);
```

```
SemiLSize = LSize/2;
```

```
//起始点坐标减去半个网格尺寸得到起始顶点坐标
imageData->SetOrigin( tempPoint. X-semiLSize, tempPoint. Y-semiLSize, tempPoint. Z-semiLSize);
//对网格循环处理
for( int it=0; it<meshnumber; it++)
{
...
tempPoint = Multiple_Lattice. GetMultipleLatticeCoordinate (0, it);
xCoords->InsertNextValue( tempPoint. X-semiLSize);
xCoords->InsertNextValue( tempPoint. X+semiLSize);
yCoords->InsertNextValue( tempPoint. Y-semiLSize);
yCoords->InsertNextValue( tempPoint. Y+semiLSize);
zCoords->InsertNextValue( tempPoint. Z-semiLSize);
zCoords->InsertNextValue( tempPoint. Z+semiLSize);
//设置属性数据与颜色映射方法,param 为待映射的属性参数
```

```
if ( param == Pressure)
{
scalars->InsertTuple1 ( it, Multiple_Lattice. GetMultipleLatticePressure(0, it);
}
else if( param == Temperature) { ... }
...
}
...
//获取三维方向上的顶点坐标范围
xCoords->GetValueRange( xRange);
yCoords->GetValueRange( yRange);
zCoords->GetValueRange( zRange);
//计算三维方向上的顶点数,+0.5 为四舍五入用于解决精度问题
```

```
int xDim = ( xRange[1]-xRange[0])/LSize+1+0.5;
int yDim = ( yRange[1]-yRange[0])/LSize+1+0.5;
int zDim = ( zRange[1]-zRange[0])/LSize+1+0.5;
//建立数据集的拓扑
imageData->SetDimensions( xDim, yDim, zDim);
imageData->SetSpacing( LSize, LSize, LSize);
```

在对网格的循环处理过程中,中心点坐标通过在三维方向上分别加減 1/2 个网格宽度,转化为网格顶点坐标。`Scalar` 是存放属性颜色映射的数组,使用颜色映射表通过特定的映射方法与每个网格单元的属性数据关联,这也是标量属性可视化最常用的方法。颜色映射表映射公式为:

$$Scalar_i = \frac{Param_i - Param_{min}}{Param_{max} - Param_{min}} \quad (1)$$

其中, $Param_i$ 为第 i 个网格的属性; $Param_{max}$ 、 $Param_{min}$ 分别代表属性最大和最小值。

`vtkLookupTable` 内部就是通过这种方式将属性与

颜色进行映射,并将 `vtkLookupTable` 嵌入映射器内部。

(2) 过滤器数据处理。在基本的数据集结构和颜色映射构建后,大部分应用还需要使用特定的过滤器提供的数据再处理功能。如数据集的切割、裁剪,等值面提取,网格平滑等。这就需要过滤器接收数据集对象作为输入,处理后的输出结果也是一种数据集。可视化流水线还支持链式过滤,顾名思义,就是指处理后的数据集还可以再次经过过滤器进行处理。VTK 还支持将过滤器处理后生成的数据集进行组合,这种方式可以极大地增加处理的灵活性。

在上一步创建数据集和颜色映射后,只能看到表面网格的结果。如果想进一步观察内部结构的可视化结果,就需要调用切割过滤器对象,切割后得到的是内部网格切面。物体的切割表面使用的是插值的方式重建,具体方法可以通过隐函数实现。VTK 提供了创建切割平面的对象和方法,根据一个已知点和法向量建立隐函数切面。VTK 支持满足隐函数 $F(x,y,z)=C$ 的任意几何体的构建。隐函数可以描述球体、圆柱、圆锥、二次曲面等图形,增加了过滤器的灵活性,满足不同的可视化应用需求。

切割过滤器接收的输入是上一部分建立的数据集对象 `imageData`,将过滤器的输出结果作为映射器的输入完成可视化模型的建立过程。

建立图形模型,完成整个可视化流水线。整个过程和其中所涉及到的对象在图 1 中的图形模型已经示出,在这里不再赘述。

3.3 实验结果展示

文中基于软件系统生成的数据源进行可视化流程。软件读入 STL 原网格文件,对流体各种参数设置后进行网格剖分、设置初始化流速为 $(1,0,0)$,然后运行求解过程,求解算法完成后便生成可视化模块的输入数据源。

选择待可视化的流场物理量(压力、温度等),数据源经过可视化过程后(3.1 部分所述)在结果图形控件上显示。以压力为例,可视化结果如图 2 所示。

基本的属性标量可视化后,输入点坐标和法线的三维分量建立隐函数切面。调用切面过滤算法对整个流场进行切割并重建切割面,这样就可查看任意流场内部的属性分布情况。图 3、4 分别显示 $(0,1,2)$ 和 $(1,0,0)$ 两个方向上的内部压力分布效果。

流线是矢量可视化的主要方式。流线可以描述网格内部流体流动方向,并可以以颜色映射流速。流线可视化需要一个管线和积分方法,常用的积分方法有二阶 Runge-Kutta、四阶 Runge-Kutta 和 Euler 法^[13]。文中选用 VTK 库自带的四阶 Runge-Kutta 法,流体为空气,中心区域为 $-1 \sim 1$ 的物面。可视化后的流线图

如图 5 所示。

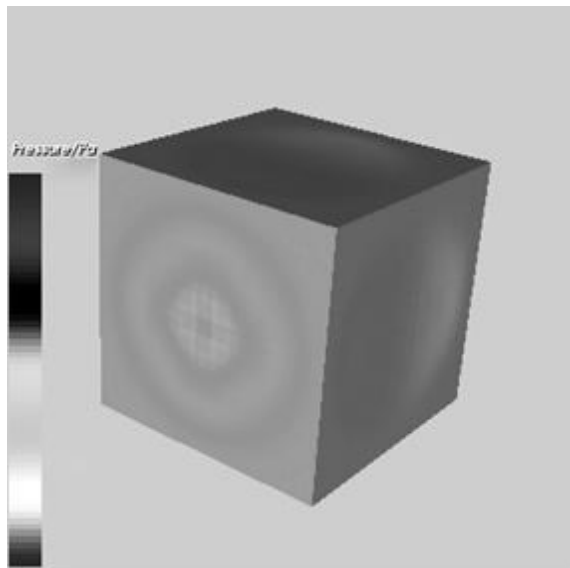


图 2 压力属性显示

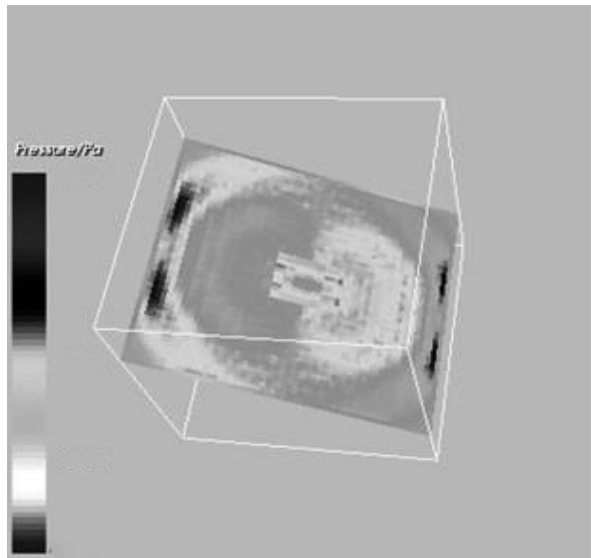


图 3 $(0,1,2)$ 方向上的内部压力分布切面

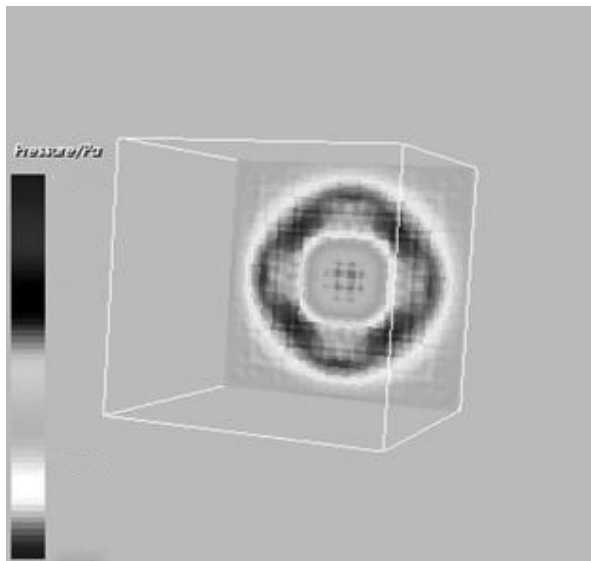


图 4 $(1,0,0)$ 方向上的内部压力分布切面

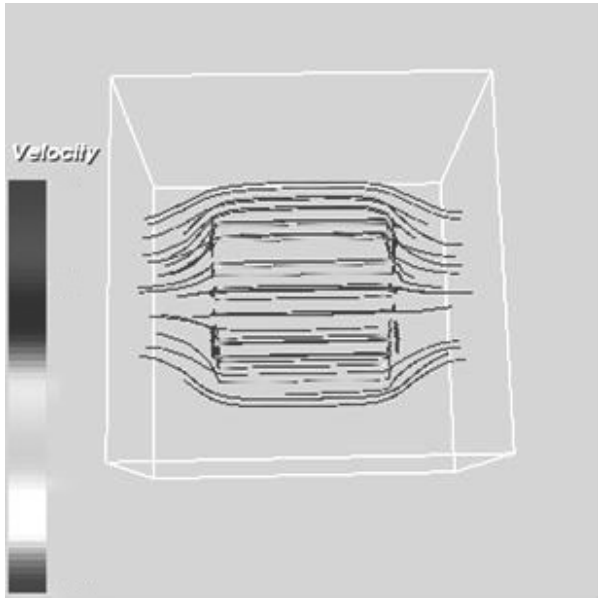


图 5 流线图

4 结束语

CFD 可视化作为科学计算可视化的一个重要领域,在发展过程中产生了众多可视化方法。文中以 VTK 为可视化工具库,在团队开发的 LBM 软件系统的基础上增加三维结构化 CFD 可视化功能模块。这些模块主要包括标量属性的颜色映射,内部网格切面及基本矢量可视化(流线、三维矢量线等)。流体可视化内容和方法有很多种,在基于特征的向量、等值面、粒子动画和张量相关的可视化技术和方法还有待更深入研究^[14]。由于科学计算的运算规模极为庞大,通常会生成一些超大规模的数据。数据量可能达到 TB 甚至 PB 级,因此在单一的 CPU 或计算节点进行可视化上往往会出现性能瓶颈。对这种大规模数据采用并行和分布式处理可极大提高可视化效率^[15]。因此并行可视化方法也成为可视化研究的一大热点。

参考文献:

- [1] McCormick B H, Defanti T A, Brown M D. Visualization in scientific computing[J]. Computer Graphics, 1987, 21(6): 1103-1109.
- [2] Upson C, Jr T F, Kamins D, et al. The application visualization system: a computational environment for scientific visualization [J]. IEEE Computer Graphics and Applications, 1989, 9(4):

30-42.

- [3] Abram G, Ternish L. An extended data-flow architecture for data analysis and visualization[J]. Computer Graphics, 1995, 29(2): 17-21.
- [4] 田 薇, 孙 秦, 贺海鸥. 基于 Tecplot 的飞行器有限元图形处理技术[J]. 机械设计与制造, 2006(2): 125-126.
- [5] Schroeder W, Martin K, Lorensen W. The visualization toolkit: an object-oriented approach to 3D graphics[M]. Upper Saddle River, NJ: Prentice Hall, 1996.
- [6] 刘晓波, 华祖林, 何国建. 计算流体力学的科学计算可视化研究进展[J]. 水动力学研究与进展, 2004, 19(1): 120-125.
- [7] Blazek J. Computational fluid dynamics: principles and applications[M]. Elsevier, Oxford: [s. n.], 2001.
- [8] Sewell C, Meredith J, Moreland K, et al. The SDAV software frameworks for visualization and analysis on next-generation multi-core and many-core architectures[C]//Proc of SCC. [s. l.]: [s. n.], 2012: 206-214.
- [9] Haber R B, McNabb D A. Visualization idioms: a conceptual model for scientific visualization systems[C]//Proc of visualization in scientific computing. Los Alamitos, CA: IEEE Computer Society Press, 1990: 74-92.
- [10] Kitware Inc. The visualization toolkit user's guide[M]. [s. l.]: Kitware Inc Publishers, 2004.
- [11] Gayer M, Kortelainen J, Karhela T. CFD modeling as an integrated part of multi-level simulation of process plants: semantic modeling approach[C]//Proceedings of the 2010 summer computer simulation conference. [s. l.]: [s. n.], 2010: 219-227.
- [12] 虞幸皓, 陈志杨, 樊俊飞, 等. 一种针对冶金流动过程的 CFD 可视化系统开发[J]. 计算机应用研究, 2005, 22(10): 188-189.
- [13] 李海生, 杨 钦, 陈其明. 三维计算流体力学流场的流线构造[J]. 北京航空航天大学学报, 2003, 29(5): 434-437.
- [14] Peng Zhenmin, Grundy E, Laramée R S. Mesh-driven vector field clustering and visualization: an image-based approach [J]. IEEE Trans on Visualization and Computer Graphics, 2012, 18(2): 283-298.
- [15] Lo Li-ta, Sewell C, Ahrens J. PISTON: a portable cross-platform framework for data-parallel visualization operators[C]//Proc of Eurographics symposium on parallel graphics and visualization. [s. l.]: [s. n.], 2012.

基于VTK的CFD可视化研究

作者：[李信凯](#)，[王帅](#)，[封卫兵](#)，[LI Xin-kai](#)，[WANG Shuai](#)，[FENG Wei-bing](#)
作者单位：[上海大学 计算机工程与科学学院, 上海, 200444](#)
刊名：[计算机技术与发展](#)[ISTIC](#)
英文刊名：[Computer Technology and Development](#)
年，卷(期)：2015(10)

引用本文格式：[李信凯](#).[王帅](#).[封卫兵](#).[LI Xin-kai](#).[WANG Shuai](#).[FENG Wei-bing](#) [基于VTK的CFD可视化研究](#)[期刊论文]
]-[计算机技术与发展](#) 2015(10)