

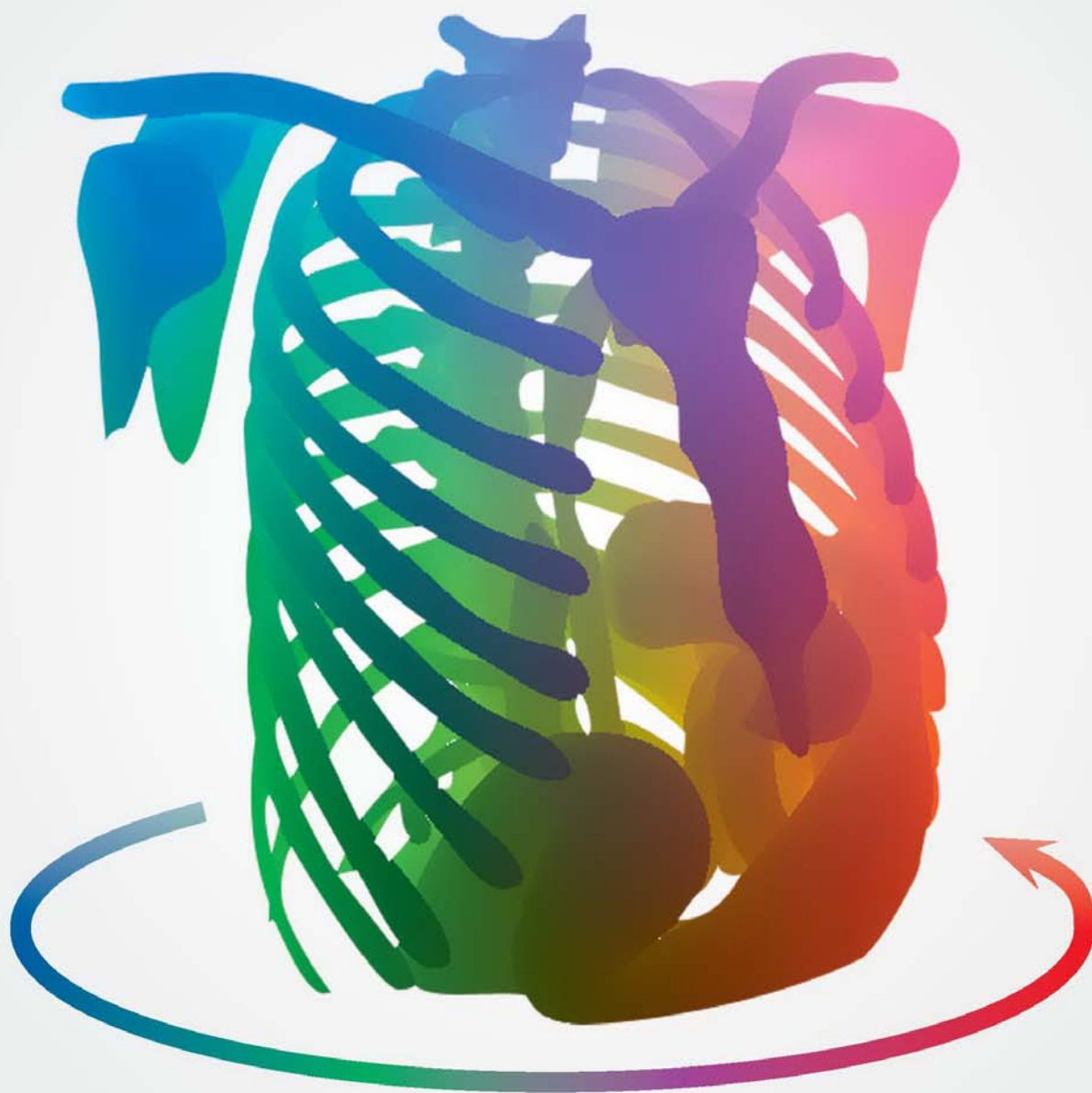
JOURNAL OF IMAGE AND GRAPHICS

主办: 中国科学院遥感与数字地球研究所
中国图象图形学学会
北京应用物理与计算数学研究所

中国图象图形学报

2015
02
VOL.20

ISSN1006-8961
CN11-3758/TB



等值面光线跟踪 P193



第20卷第2期（总第226期）

2015年2月16日

中国精品科技期刊
中国国际影响力优秀学术期刊
中国科技核心期刊

版权声明

凡向《中国图象图形学报》投稿，均视为同意在本刊网站及CNKI等全文数据库出版，所刊载论文已获得著作权人的授权。本刊所有图片均为非商业目的使用，所有内容，未经许可，不得转载或以其他方式使用。

Copyright

All rights reserved by Journal of Image and Graphics, Institute of Remote Sensing and Digital Earth, CAS. The content (including but not limited text, photo, etc) published in this journal is for non-commercial use.

主管单位 中国科学院

主办单位 中国科学院遥感与数字地球研究所

中国图象图形学学会

北京应用物理与计算数学研究所

主 编 李小文

编辑出版《中国图象图形学报》编辑出版委员会

邮政信箱 北京9718信箱

邮 编 100101

电子信箱 jig@irsa.ac.cn

电 话 010-64807995

网 址 www.cjig.cn

广告经营许可证 京朝工商广字第0361号

总 发 行 北京报刊发行局

订 购 全国各地邮局

海外发行 中国国际图书贸易集团有限公司

（邮政信箱：北京399信箱 邮编：100048）

印刷装订 北京科信印刷有限公司

Journal of Image and Graphics

Title inscription: Song Jian

Monthly, Started in 1996

Superintended by Chinese Academy of Sciences

Sponsored by Institute of Remote Sensing and Digital Earth, CAS

China Society of Image and Graphics

Institute of Applied Physics and Computational Mathematics

Editor-in-Chief LI Xiaowen

Editor, Publisher Editorial and Publishing Board of Journal of Image and Graphics

P.O.Box 9718, Beijing, P.R.China

Zip code 100101

E-mail jig@radi.ac.cn

Telephone 010-64807995

Website www.cjig.cn

Distributed by Beijing Bureau for Distribution of Newspapers and Journals

Domestic All Local Post Offices in China

Overseas China International Book Trading Corporation
(P.O.Box 399, Beijing 100048, P.R.China))

Printed by Beijing Kexin Printing Co., Ltd.

CN 11-3758/TB

ISSN 1006-8961

CODEN ZTTXFZ

国外发行代号 M1406

国内邮发代号 82-831

国内定价 50.00元



多尺度活动网格在云场景仿真中的应用(第0202页)



自然环境视频中基于显著鲁棒轨迹的行为识别(第0245页)



虚拟手抓持力觉生成算法真实性的评价(第0280页)

图像处理和编码

引入神经网络中间神经元的快速小波图像压缩

张海涛, 张永霖 0159

拟正态分布扩散的图像平滑

周先春, 汪美玲, 周林锋 0169

可选特征的快速分形图像编码

袁宗文, 鲁业频, 杨汉生 0177

图像分析和识别

融合多尺度码本的全局编码图像分类

董振宇, 赵杰煜, 祝军 0183

计算机图形学

结合K-D树和Shell的快速动态等值面光线跟踪法

罗月童, 石放放, 张伟, 朱会国 0193

多尺度活动网格在云场景仿真中的应用

范晓磊, 张立民, 张建廷, 徐涛 0202

第十届和谐人机环境联合会议专栏

低资源语言的无监督语音关键词检测技术综述

杨鹏, 谢磊, 张艳宁 0211

快速离散Curvelet变换域的图像融合

杨勇, 董松, 黄淑英 0219

高效视频编码中变换跳过模式的快速选择

王宁, 张永飞, 樊锐 0229

主动学习的多标签图像在线分类

徐美香, 孙福明, 李豪杰 0237

自然环境视频中基于显著鲁棒轨迹的行为识别

易云, 王瀚漓 0245

特征变换和数据集分块的行人比对

沈洋, 林巍峭, 殷惠清 0254

移动手持环境下触摸式目标选择技术对比

辛义忠, 李岩, 袁伟强, 郑谦 0264

虚拟环境的用户意图捕获

程成, 赵东坡, 卢保安 0271

虚拟手抓持力觉生成算法真实性的评价

杨文珍, 许艳, 颜传武, 吴新丽, 张昊, 孟闯 0280

交互式乐器演奏的六自由度力觉渲染方法

童号, 史有姣, 王党校, 张玉茹 0288

CONTENTS

JOURNAL OF IMAGE AND GRAPHICS



Multi-scale moving grids and its application in clouds scene simulation(P0202)



Action recognition from unconstrained videos via salient and robust trajectory (P0245)



Validation the Authenticity of Virtual fingers Grasping Haptic Algorithm(P0280)

Image Processing and Coding

- Fast wavelet image compression introducing neural network of relay neuron
Zhang Haitao, Zhang Yonglin 0159
- Image smoothing algorithm based on matching normal distribution diffusion
Zhou Xianchun, Wang Meiling, Zhou Linfeng 0169
- Optional features fast fractal image coding algorithm
Yuan Zongwen, Lu Yepin, Yang Hansheng 0177

Image Analysis and Recognition

- Image classification based on global coding combined with multi-scale codebook
Dong Zhenyu, Zhao Jieyu, Zhu Jun 0183

Computer Graphics

- Fast isosurface ray tracing method by combining K-D tree and Shell
Luo Yuetong, Shi Fangfang, Zhang Wei, Zhu Huiguo 0193
- Multi-scale moving grids and its application in clouds scene simulation
Fan Xiaolei, Zhang Limin, Zhang Jianting, Xu Tao 0202

Column of HHME'2014

- Survey on unsupervised spoken term detection for low-resource languages
Yang Peng, Xie Lei, Zhang Yanning 0211
- Image fusion based on fast discrete Curvelet transform
Yang Yong, Tong Song, Huang Shuying 0219
- Fast transform skip mode decision for high efficiency video coding
Wang Ning, Zhang Yongfei, Fan Rui 0229
- Online multi-label image classification with active learning
Xu Meixiang, Sun Fuming, Li haojie 0237
- Action recognition from unconstrained videos via salient and robust trajectory
Yi Yun, Wang Hanli 0245
- Pedestrian re-identification based on feature transformation and dataset classification
Shen Yang, Lin Weiyao, Yin Huiqing 0254
- Comparison of target selection techniques in mobile handheld environment
Xin Yizhong, Li Yan, Yuan Weiqiang, Zheng Qian 0264
- Intent understanding for virtual environments
Cheng Cheng, Zhao Dongpo, Lu Baoan 0271
- Validation the authenticity of virtual fingers grasping haptic algorithm
Yang Wenzhen, Xu Yan, Yan Chuanwu, Wu Xinli, Zhang Hao, Meng Chuang 0280
- Interactive simulation of music instrument playing by a six degree-of-freedom haptic rendering approach
Tong Hao, Shi Youjiao, Wang Dangxiao, Zhang Yuru 0288

中图法分类号: TP391.9 文献标识码: A 文章编号: 1006-8961(2015)02-0202-09

论文引用格式: Fan X L, Zhang L M, Zhang J T, Xu T. Multi-scale moving grids and its application in clouds scene simulation[J]. Journal of Image and Graphics, 2015, 20(2): 0202-0210. [范晓磊, 张立民, 张建廷, 徐涛. 多尺度活动网格在云场景仿真中的应用[J]. 中国图象图形学报, 2015, 20(2): 0202-0210.] [DOI:10.11834/jig.20150206]

多尺度活动网格在云场景仿真中的应用

范晓磊^{1,3}, 张立民², 张建廷¹, 徐涛¹

1. 海军航空工程学院电子信息工程系, 烟台 264001; 2. 海军航空工程学院科研部, 烟台 264001;
3. 第二炮兵工程大学士官职业技术教育学院, 青州 262500

摘要: 已有的基于网格的流体仿真方法存在效率低的问题, 不利于物体参与交互, 提出一种多尺度活动网格方法用于云场景的仿真以及云与物体的交互仿真。**方法** 自动选择全局与局部网格的分辨率, 对参与交互的物体建立 OBB(oriented bounding box)层次包围盒以提高仿真效率。针对云的粘性系数非常小的特点, 简化了它们的运动方程与热动力学方程。基于提出的方法对偏微分方程进行了离散化, 在方程离散化处理过程中, 采用迎风差分方法以保证仿真过程的稳定性。实现过程中, 运用 GPU 的计算性能提高仿真速度。**结果** 通过云场景的仿真进行了实验, 结果表明, 可以较为真实地绘制不同时间段的云场景, 能满足大规模云场景的仿真需求, 实现了云场景以及云与刚性物体交互的实时绘制。**结论** 相比已有基于网格的方法, 本文方法可以容易地在 GPU 上实现, 且提高了仿真效率与绘制图像的真实感。

关键词: 实时绘制; 多尺度活动网格; 云场景仿真; 方程简化; 迎风差分

Multi-scale moving grids and its application in clouds scene simulation

Fan Xiaolei^{1,3}, Zhang Limin², Zhang Jianting¹, Xu Tao¹

1. Department of Electronics and Information Engineering, Naval Aeronautical and Astronautical University, Yantai 264001, China;
2. Department of Scientific Research, Naval Aeronautical and Astronautical University, Yantai 264001, China;
3. Noncommissioned Officers Vocational and Technical Education College, The Second Artillery Engineering University, Qingzhou 262500, China

Abstract: **Objective** Previous grid-based fluid simulation methods are inefficient and are thus non-conducive to the interaction between objects and fluids. The multiscale adaptive moving grid method is proposed in this study. The proposed method is used in cloud scene simulation and in interaction simulation between objects and clouds. **Method** The resolutions of global and local grids can be adaptively selected, and the oriented bounding box hierarchical trees of objects are established to enhance simulation efficiency. Given the small kinematic viscosity of clouds, the controlling and thermal dynamics equations are simplified. Partial differential equations are discretized according to the proposed method. Moreover, upwind difference is applied to ensure the stability of the simulation process. The method is designed to take advantage of GPU computing performance to improve simulation speed. **Result** Cloud scene simulation is performed in the tests. Results show that the proposed method can render cloud scenes in different periods realistically and thus meets the requirements of large-scale cloud scene simulation. Simulation of cloud scenes and of the interaction between rigid objects and clouds can be real-

收稿日期: 2014-07-09; 修回日期: 2014-10-17

基金项目: 国家自然科学基金项目(61102167)

第一作者简介: 范晓磊(1976—), 男, 讲师, 海军航空工程学院信息与通信工程专业博士研究生, 主要研究方向为计算机图形学、视景仿真、虚拟现实。E-mail: gm_fxli@163.com

Supported by: National Natural Science Foundation of China (61102167)

ized in real time. **Conclusion** Compared with the previous grid-based methods, the proposed method can be conveniently realized in a GPU, thus enhancing simulation efficiency and the realism of rendered images.

Key words: real-time rendering; multi-scale moving grids; clouds scene simulation; equations simplification; upwind difference

0 引言

流体仿真是多年来计算机图形学领域研究的热点,有广泛的应用。文献[1-6]中实现了流体的仿真,并对流体的运动方程求解提出了不同的改进方法,但所实现的方法中,受到稳定性的困扰。云作为一种常见的流体,它的仿真方法可以分为两类:启发式的方法和基于物理的方法^[7]。近年来,基于物理的方法越来越成为主流。基于物理的方法关键在于偏微分方程的求解,跟其他流体的仿真类似。文献[1,8]基于流体动力学生成了云,该方法仅适用于单一类型云的仿真。Harris 基于物理的方法对动态云场景进行了仿真^[9]。文献[10]基于流体动力学方法模拟大规模云场景,但未实现动态的交互。

基于网格的方法是流体仿真中解偏微分方程时常用的技术,基于网格的方法大多在仿真空间中生成一个网格空间来计算流体的运动,将控制流体运动的 N-S (Navier-Stokes) 方程使用网格来离散化,采用有限差分的方法求解。

Stam 在文献[11]中解决了数值仿真的稳定性问题之后,出现了许多流体仿真方法^[12-14]。这些方法均使用了单一网格,在表示跟流体交互的物体形状时,网格点的数量需要大幅度增加,且难以实现 GPU 加速。文献[3]中仿真了烟与刚性物体的交互,为烟和待交互的刚体建立了不同的网格分辨率,但未能实现自适应选取分辨率,影响了方法的实用性。且物体所占用的网格单元同样要进行流体仿真计算,需要频繁的数据传输。文献[6]中采用了自适应的多网格化方法实现了烟与刚体的交互,但仿真过程中需要计算包括物体在内的所有网格单元。

文献[1,9]对云的仿真中运用了网格化的方法,文献[9]中还实现了刚体跟云的交互过程。这些方法存在着网格单一,不利于其他物体参与交互的问题,且云的数据均需要在预处理阶段生成。为解决上述问题,提出了多尺度活动网格的方法用于云的仿真,并通过仿真实验进行了验证。

1 云的建模及方程的简化处理

1.1 流体动力学方程及简化处理

云的运动可以用非压缩流体的 N-S 方程描述如下^[9,15]

$$\frac{\partial \mathbf{u}}{\partial t} = -(\mathbf{u} \cdot \nabla) \mathbf{u} - \frac{1}{\rho} \nabla p + \frac{1}{Re} \nabla^2 \mathbf{u} + \mathbf{F} \quad (1)$$

$$\nabla \cdot \mathbf{u} = 0 \quad (2)$$

式中, $\mathbf{u}$ 为流体速度,为矢量, t 为时间变量, ρ 为流体密度,为常量, p 为气压, Re 为雷诺数, $\mathbf{F}$ 为施加到流体上的外部作用力, ∇ 为梯度算子, $\nabla \cdot$ 为散度算子, ∇^2 为拉普拉斯算子。式(1)称为运动方程,式(2)为连续性方程。式(1)中右边第1项称为平流项,第2项为压力项,第3项为粘性项,第4项为外部作用力。在对云的仿真中,对 N-S 方程进行了简化。

由式(1)(2)可知平流项是一惯性效果,它与粘性项所产生的效果之间的关系可由雷诺数来衡量。

在云的仿真中,雷诺数 Re 可以表示为 $Re = \frac{u \cdot L}{\nu}$, 其中

u 是特征速度, L 是特征长度, ν 是粘性系数。在 $0^\circ\text{C} \sim 60^\circ\text{C}$, ν 的变化范围在 $1.3 \times 10^{-5} \text{ m}^2/\text{s} \sim 1.9 \times 10^{-5} \text{ m}^2/\text{s}$ ^[15], 对流层的高度一般在从地球表面到 12 km 这个范围,取特征长度 L 为 100 m 或者更大值时, Re 将大于 10^7 量级,也就是说惯性力远大于粘性力,故可忽略粘性项,从而简化方程求解,描述云形成过程的 N-S 方程也就简化为非压缩流体的欧拉方程

$$\frac{\partial \mathbf{u}}{\partial t} = -(\mathbf{u} \cdot \nabla) \mathbf{u} - \frac{1}{\rho} \nabla p + \mathbf{F} \quad (3)$$

$$\nabla \cdot \mathbf{u} = 0 \quad (4)$$

1.2 能量方程及简化处理

为加强云仿真的真实感,充分考虑热动力学对仿真过程的影响。为计算浮力,需要对气团内温度的变化进行建模,可用下面的微分方程描述^[15]

$$\frac{\partial T}{\partial t} = -\mathbf{u} \cdot \nabla T + \frac{1}{Re} \times \frac{1}{Pr} \nabla^2 T + q''' \quad (5)$$

式中, T 为温度, t 为时间变量, q''' 为热量,该方程为能量方程。

对于能量方程右边的第2项,由前面 N-S 方程中去除粘性项可知, Re 的值要大于 10^7 的量级,而 Pr (Prandtl number, 普朗特数) 的值变化范围对液体是 $3 \sim 300$, 气体则是 $0.7 \sim 1.0^{[15]}$, 可见, 对于气体 $\frac{1}{Re} \times \frac{1}{Pr}$ 的变化范围为 $1 \times 10^{-7} \sim 1.429 \times 10^{-6}$, 导致项 $\frac{1}{Re} \times \frac{1}{Pr} \nabla^2 T$ 的值非常小, 故为简化将该项省略, 从而能量方程简化为:

$$\frac{\partial T}{\partial t} = -\mathbf{u} \cdot \nabla T + q''' = -\left(u \frac{\partial T}{\partial x} + v \frac{\partial T}{\partial y} + w \frac{\partial T}{\partial z}\right) + q''' \quad (6)$$

式中, x, y, z 分别为网格空间坐标的3个分量。

2 多尺度自适应活动网格方法

在流体仿真过程中, 流体与周围的物体进行交互是常见的现象, 已有流体与周围物体的交互仿真方法中, 大多是将整个仿真空间(含交互的物体空间)作为一个完整的空间。虽然文献[3, 6]中使用了仿真子空间, 但也存在着前文所说的不足。在云的交互仿真方法中, 同样将仿真空间作为一个不变的网格空间, 不论待交互的物体远近均采用相同尺度的网格, 当视点或交互的物体位置发生变化时, 将严重影响绘制的效率和图像的真实感, 提出的多尺度活动网格方法可以解决这些不足。

2.1 多尺度活动网格

2.1.1 多尺度活动网格的表示

多尺度活动网格方法首先将整个仿真空间(包含待交互的物体)作为全局仿真空间, 该网格空间称为全局网格, 网格空间均匀划分, 图1为2维均匀网格空间及其参与交互物体。图中阴影区域表示待交互物体, 右下角的物体是运动之后的效果, 包围着物体的绿色网格称为局部网格, 具有不同的分辨率。仿真过程中全局网格和局部网格的分辨率可以发生变化。在上方包围着物体的蓝色方框是该物体的 OBB(oriented bounding box)包围盒。仿真空间全局网格和局部网格具体选用哪个级别的分辨率根据视点与仿真空间以及物体之间距离决定。参与交互的物体根据局部网格分辨率的判定原则在其边缘区域建立局部网格, 该局部网格只在物体边缘占用的全

局网格单元上建立, 如图1中绿色网格所示。物体在仿真空间中可以自由运动, 不受速度的限制, 其 OBB 包围盒作相应的运动。由于速度不受限制, 拓展了方法的适用范围, 参与交互的物体可以是高速运动的物体。图2是3维情形, 图中红色阴影代表交互的物体, 右侧是一个独立的网格单元。

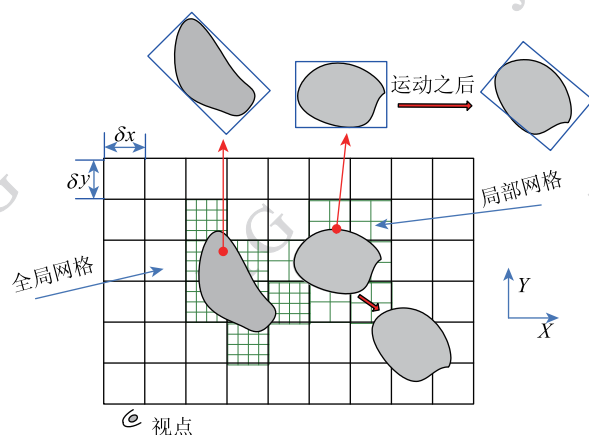


图1 2维均匀多尺度交叉网格
Fig. 1 2D multi-scale staggered grid

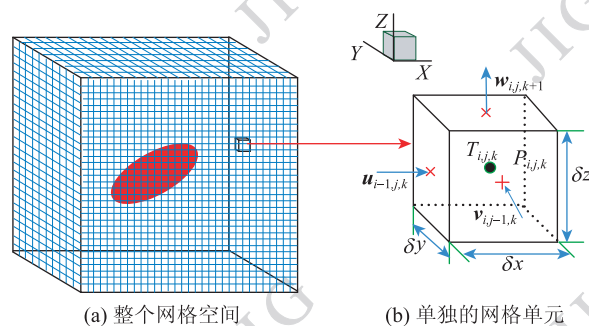


图2 3维均匀网格
Fig. 2 3D staggered grid

文中全局网格采用可变的分辨率, 这样在需要时会更加清晰地显示出流体更多的细节。对于流体所用到的全局网格的分辨率, 也会由于视点距离仿真空间中心的远近更新全局网格的分辨率, 但在不同的时间步长上更新全局网格分辨率较为耗时, 将根据全局网格分辨率的判定规则, 不会频繁更新。为交互的物体建立了 OBB 层次包围盒, 以加快物体与流体仿真单元的相交测试速度。

2.1.2 网格的拆分与合并

由于仿真空间可以采用不同层级分辨率的网格, 不论是全局网格还是局部网格, 在变换网格分辨率时, 均会出现网格的拆分与合并问题。

对2维网格, 不同分辨率的网格之间符合 $1, 4, 16, \dots, 4^n$ 这样的关系, 也就是说网格单元或者是

一拆四或是四合一, 在 x, y 轴方向上的网格单元数分别是 2^n 个 (n 可以不相同)。对 3 维网格, 在 x, y, z 轴方向上的网格单元数分别是 2^n 个 (n 可以不相同), 网格单元是一拆八或是八合一。网格单元的合并或拆分起始位置从原点出发沿着每个轴向推进, 对于局部网格, 则是沿着物体所占用的边缘网格进行, 图 3、图 4 分别为 2 维、3 维网格的拆分与合并过程。

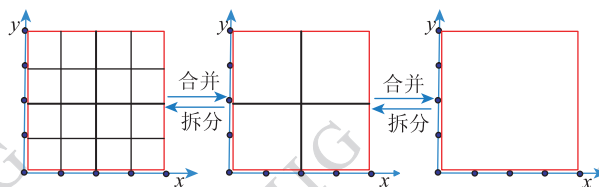


图 3 2 维网格的合并与拆分

Fig. 3 Combination and splitting of 2D grid cells

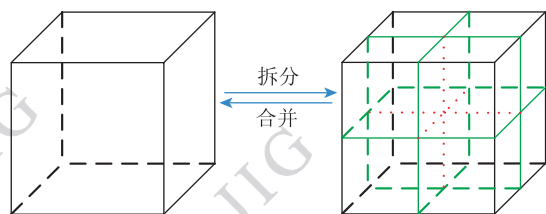


图 4 3 维网格的拆分与合并

Fig. 4 Splitting and combination of 3D grid cells

网格单元在合并之后将会有多余的数据, 以及拆分后会出现新的网格单元, 新的单元上使用哪些数据? 本文采用线性插值方法实现不同层级单元上的数据传输与变换。一般情形下, 单元拆分或合并后的数据符合图 5 的情况, 当由高分辨率网格向低分辨率网格变换时, 数据均为 $A = (a + b + c + d) / 4$, 反过来, 由低分辨率网格向高分辨率网格变换时, 则为 $a = \frac{1}{4}(A_1 + A_3) + \frac{3}{8}A + \frac{1}{8}A_2$, 其他的网格单元数据变换类似处理。

对网格单元进行拆分时, 会遇到单元是边缘网格单元的特殊情形, 此时仍采用上面的公式, 但所用到的其他边界网格单元 (实际不存在) 的数据按照所预设的边界值进行处理。对于合并的情形, 则不会出现此问题。

2.1.3 物体占用网格单元的处理

对参与交互的物体建立 OBB 层次包围盒, 其数据存储采用八叉树的层次性数据结构存储。数据结构的最顶层是最外层的包围盒, 最底层属于叶子节

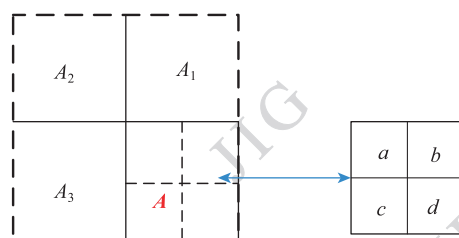


图 5 网格单元拆分或合并后的数据变换

Fig. 5 Data transfer after the splitting or combination of grid cells

点, 存储物体的基本单元信息。在对物体建立层次数据结构时, 只对物体的 3 维表面建立, 不需要物体内部数据, 这是由于物体是刚体, 在交互过程中不会发生变形, 这将提高层次包围盒的建立速度, 而且对物体是否占用网格单元的判定上也更为快速。

为记录物体是否占用了网格单元, 为每个网格单元 (包含全局网格和局部网格) 设置一个状态标志变量, 用于标记它的占用状态。物体与网格单元的占用判定采用物体的层次包围盒数据结构按照自顶向下的顺序依次排除, 顶层不相交则底下的层次肯定不相交, 只有在顶层相交时才会进行下一层的判断, 直至叶子节点。根据叶子节点与整个仿真实空间网格单元的相交情况, 如图 6(a)(b) 所示情形, 表示物体不占用网格单元 (不相交的情形下也是不占用的); 只有如图 6(c) 所示情形, 物体才占用了单元格, 则将单元格的状态变量标记为占用。

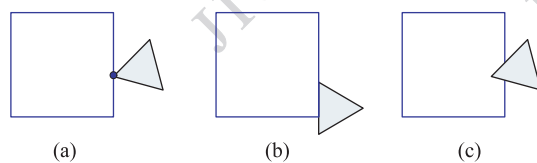


图 6 物体与网格单元的关系

Fig. 6 Relationship between object and grid cell

当物体运动之后, 也许会占用新的全局网格单元, 从而也会腾出新的未占用单元, 这些单元上的数据处理如下:

1) 由占用到未占用。沿着物体运动的方向, 从占用区域的边缘出发, 由外及内的顺序, 如由占用变为未占用状态, 则由其周围未占用的网格单元上的数据插值确定, 有几个相邻单元 (未占用单元) 则用几个。在 3 维情形下, 对于边缘区域只考虑与面相邻的情形, 不考虑与顶点、边相接的情形, 最后将单元的占用标志变量设置为未占用。

2) 由未占用到占用。其单元内存储的数据内容不需要考虑,只需将单元的占用标志变量设置为占用。

更为复杂也是常见的问题是物体运动之后,如果局部的网格分辨率发生了变化(跟全局网格不一致时),它占用的单元如何处理?如图7中的三角形物体,初始时局部网格采用的分辨率跟全局的不一致。它占用了全局网格的4个单元,但只占用了局部网格的4个小单元(阴影区域所示),发生运动之后移动到了左下方,并且局部网格的分辨率更高,它只占用了全局网格的一个单元,占用了新的更高分辨率下局部网格的4个小单元。

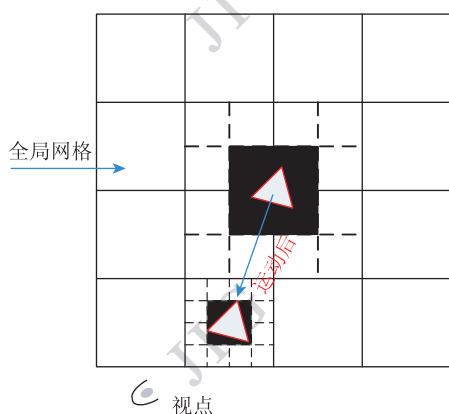


图7 物体运动后占用单元的变化

Fig. 7 Change of cell possessing after object moving

对此处理方法如下:

- (1) 由于运动,确定新的全局和局部网格分辨率;
- (2) 全局网格的拆分或合并(并不是必需的),如需要还要进行拆分或合并后的数据处理;
- (3) 根据层次包围盒与网格单元相交测试方法判定物体跟全局网格之间新的占用和未占用单元,并进行相应的数据处理,确保该时间步长内流体仿真正确进行;
- (4) 根据步骤(3),判断占用和未占用单元,确知物体占用了哪些新的全局网格单元,再由新的局部网格分辨率,拆分或合并已占用的全局网格单元,即进行局部网格的拆分或合并;
- (5) 对拆分或合并后形成的局部网格的小单元,再次确定其占用和未占用情形。当有多个物体参与交互时,它们可能会占用同一全局单元,如果它们的局部网格分辨率不相同,则对该占用单元采用更高的网格分辨率,在处理它们所占用的局部网格小单元时,确保不会发生重叠。

网格单元之间进行数据传递时,只考虑不同分辨率下全局网格单元间的数据传递。对于局部网格单元,只处理它们的占用标志,用于交互物体的显示。

2.2 网格分辨率的自适应选取

具体采用哪个层级的全局和局部网格分辨率,由视点距离仿真空间的中心位置以及视点距离参与交互物体的中心位置决定。仿真空间的中心位置容易得知,将每个物体最顶层 OBB 包围盒的中心作为物体的中心位置,可以重复使用前面计算得到的数据,减少计算量,如图8所示。

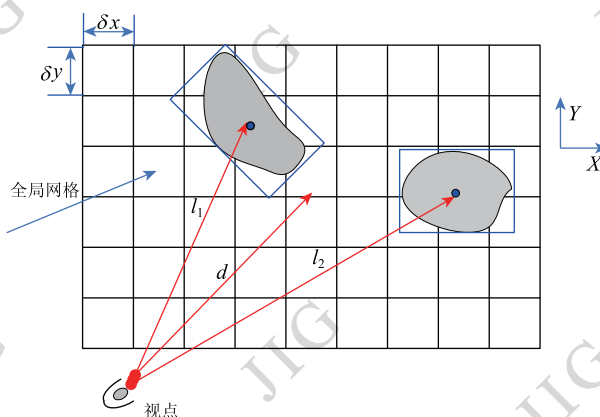


图8 视点与仿真空间、物体中心的距离

Fig. 8 Distance from the viewpoint to simulation space or to the center of object

当视点与仿真空间中心距离 d 发生变化时,为更多或更少地显示流体(云)的细节,需要调整全局网格的分辨率。当网格投影到屏幕上时,宽度为 h 的网格单元对应着 $(he)/d$ 个像素,其中 e 是常量,由相机的参数决定。当 $(he)/d = 1$ 时,即对应一个像素即可满足需要,如果比 1 更小即低于 1 个像素时,具有这样特征的信息在屏幕上显示不出来的。由此,当视点距离仿真空间中心距离 d 发生变化时,最终投影到屏幕上的网格单元基本保持占用一个像素大小。

初始时已知屏幕分辨率和显示屏幕的大小可以确定一个像素的大小宽度。此时,由 $(he)/d = 1$ 可知,设 $h_{\max}$ 为最高分辨率时的网格单元大小,仿真空间确定后,此值固定, $d_{\max}$ 为此时的视点到仿真空间中心的距离,则 $d_{\max} = h_{\max} e$, 令

$$M = \left\lfloor \log_2 \frac{d}{d_{\max}} \right\rfloor$$

式中, $\lfloor \cdot \rfloor$ 表示向下取整,新的全局网格分辨率改变为 $2^{(N-M)}$, N 为最高分辨率时的指数,可由初始设定的仿真空间大小和 $h_{\max}$ 求出。在实际的仿真过程

中,将全局网格的最高分辨率设为 $2^{(N-1)}$,也就是说比真正的最高分辨率低一个级别,并且改变的频率也不会如此高,采用 $M = \left\lfloor \log_4 \frac{d}{d_{\max}} \right\rfloor$ 。

局部网格分辨率的变化是随着视点的移动以及物体的不断运动而发生变化的,它们的变化采用类似全局网格的变化方式。设 $h_{\max}$ 为最高分辨率时的局部网格单元大小(跟全局网格是一致的), $l_{\max}$ 为此时的视点到仿真空间中心的距离,则 $l_{\max} = h_{\max} e$,设 l_i 为随时变化的视点到物体中心的距离,下标 i 表示第 i 个物体,令

$$L = \left\lfloor \log_2 \frac{l_i}{l_{\max}} \right\rfloor$$

新的局部网格分辨率改变为 $2^{(N-L)}$, N 值同上。

通过对网格分辨率的调整可以看出,由初始的屏幕显示分辨率和屏幕显示空间,可确定仿真空间的网格最高分辨率,在视点或物体变化的时候,可以自动调整全局网格和局部网格的分辨率,不需要手工调整或是实验的方式确定分辨率的级别,达到自适应调节的目的。

2.3 方程的离散化及自适应活动网格方法算法流程

2.3.1 方程的离散化处理及改进

采用有限差分的方法对微分方程进行离散化求解。引入迎风差分方法解决稳定性问题,并在不同

的阶段将迎风差分与中心差分相结合。迎风差分速度上略慢于向前或向后差分,但可以避免不稳定性。

在解欧拉方程(适用于N-S方程)和能量方程时,仿真区域采用文中提出的多尺度均匀网格来离散化,不同的未知变量处在不同的网格点。规定压强 p 、温度 T 在网格单元中央,速度分量 u, v, w 分别处于面的中央,如图2所示。对运动方程离散化时,为保证稳定性,采用迎风差分进行离散化改进。同样对式(6)中的一阶微分也采用迎风差分进行改进。

2.3.2 多尺度自适应活动网格方法的算法流程

考虑一个单独的时间步长,在该时间步长内,物体的运动优先考虑。物体先运动,可能会占据流体所处的单元,此时流体仿真单元的占用标志变量要修改,物体运动后腾出的网格单元可由流体占据,最后计算流体的运动。第 i 个时间步长的具体算法流程如图9所示。

在进行流体仿真处理时,沿着 x, y, z 轴方向,向前推进,直到最后单元格,如果遇到已占用单元,则直接跳过,不进行任何处理,提高了仿真效率,在仿真过程中未出现不稳定问题。

3 仿真实验结果及分析

本文方法在PC机上进行了仿真和验证,机器的

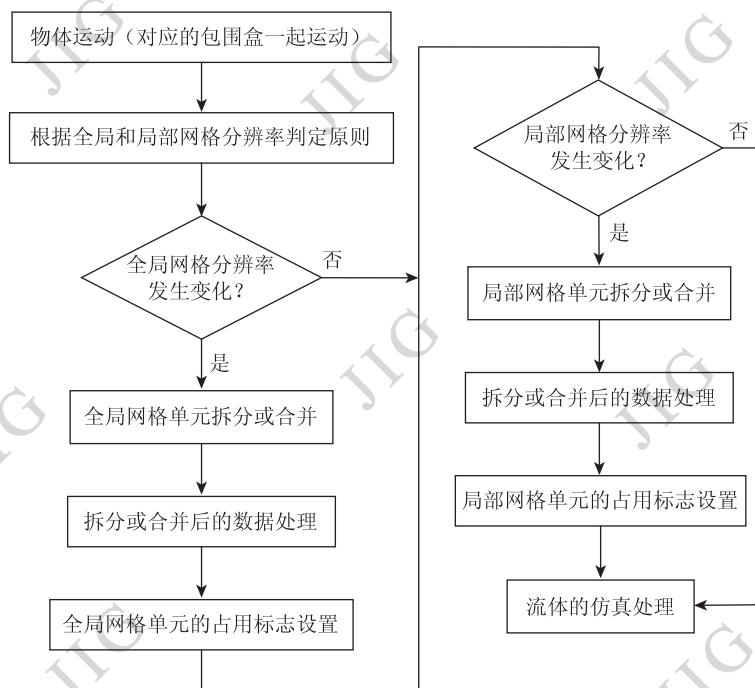


图9 多尺度自适应活动网格方法的算法流程

Fig. 9 Flow chart of the method of multi-scale adaptive moving grids

配置是: Intel i5-2380P CPU (3.10 GHz, 双核), 4.00 G 内存(RAM), 显卡为 NVIDIA GeForce GTX 550 Ti。

3.1 基于 GPU 的加速实现

文中所提出的多尺度活动网格方法可以较容易地在 GPU 上实现,这是由于所用的网格单元是均匀网格单元。每个网格单元上的物理量如速度、压强、温度可以直接用 3 维纹理进行存储。由于所用到的网格单元有是否占用的标志,根据这个标志,在不同的仿真步骤中,决定是否存储该全局网格单元上的变量到纹理元素中,只有当全局网格的分辨率发生变化时才会更新整个纹理。仿真过程中网格单元的拆分与合并,虽然简单,但却是同一动作的反复进行,正适合 GPU 完成,故此也将其移植到 GPU 上实现。

具体的实现本文采用了文献[9,16]中所提出的方法,求解泊松方程时用的是逐次超松弛迭代的方法,它的实现比文献[9]中的 Jacobi 迭代复杂,但收敛速度更快。

3.2 稳定性问题分析

对文中采用的迎风差分方法跟中心差分及向前、向后差分方法进行了跟踪观察。通过采样发现,采用中心差分在个别的时间步长上会出现振荡问题,改用迎风差分后,则会避免这样的振荡出现,如图 10 所示。采用的迎风差分方法满足相容性,且是条件稳定和收敛的,其稳定条件为 CFL (Courant-Friedrichs-Lewy) 条件,即单位时间步长内运动的距离不超过该方向上网格单元的大小,即可满足仿真过程的稳定性。

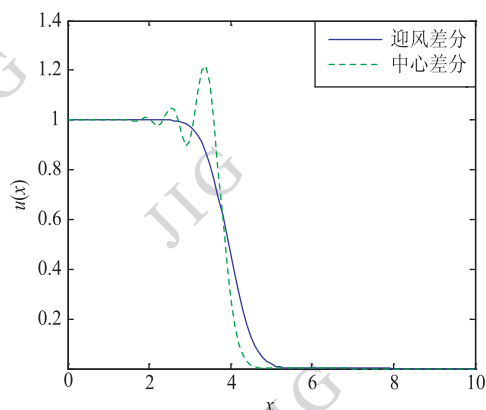


图 10 迎风差分与中心差分的稳定性比较

Fig. 10 Stability comparison of upwind difference and central difference

3.3 仿真时间比较与分析

通过云的仿真对多尺度活动网格方法进行了实验,同时将其他文献中提出的变换网格分辨率的方法进行了实验,并将本文方法与文献[3,6]中的采用全局和局部网格的方法,以及以前文献中只进行全局网格分辨率的更新方法进行了比较。

将每显示一次仿真效果为一个统计周期,统计流体的仿真计算以及跟物体交互所用时间。表 1 中的网格分辨率指最高网格分辨率,一般来说,局部网格分辨率要高于全局。表 1 中以前方法代表只进行全局网格分辨率的更新方法,DMYN 代表文献[3]中的方法,EQYF 代表文献[6]中的方法,OUR 表示本文提出的方法,不同网格分辨率下的时间统计如表 1 所列。

表 1 不同方法在不同网格分辨率下所用时间比较

Table 1 Time comparison of different methods in

different grid resolutions				/s
网格分辨率	以前方法	DMYN	EQYF	OUR
128 × 128 × 128	0.006 2	0.005 3	0.004 9	0.003 6
256 × 256 × 128	0.010 5	0.021 5	0.018 6	0.006 8
256 × 256 × 256	0.031 7	0.042 2	0.030 3	0.011
512 × 512 × 256	—	0.173 5	0.145 0	0.040 8
512 × 512 × 512	—	0.362	0.288 2	0.079
1 024 × 512 × 512	—	—	0.513 2	0.162

注:—表示计算时间较长,难以达到实时效果,未进行统计。

通过采样发现,以前方法在不同时间步长所用时间起伏较大,这是由于该步长是否需要网格更新引起。其他方法所用时间变化不大,这是由于其他方法网格分辨率的更新只是局部的。从所用时间来看,文献[6]中的方法跟文献[3]中的方法速度差别不大,它们均快于以前更新全局网格分辨率的方法,因为二者都是根据需要更新局部网格的分辨率,本文方法比文献[3,6]中的方法都要快,具体提高效率跟这些方法中选用的全局网格分辨率有关。

本文方法速度比较快主要有以下 4 个方面的原因:根据需要更新全局和局部网格分辨率,而文献[3,6]中虽然不更新全局网格分辨率,但它们的方法效率高受初始选用的全局网格分辨率的影响,本文方法虽变换全局网格分辨率,但变换频率比以前方法慢得多,对文献[3,6]中的方法和以前方法进行了折衷;网格空间的拆分与合并以及数据的传递简单;本文方法

进行流体仿真时不需要计算物体所占用的网格单元; 对参与交互的物体建立的 OBB 层次包围盒, 在判定物体占用网格单元时速度快, 提高了交互的效率。

3.4 仿真效果

运用本文方法对云进行了绘制, 绘制过程中云的光照计算运用了文献[9]中的多重前向散射光照模型。图 11 为绘制的云场景, 上述场景中没有参与交互的物体。



图 11 云场景绘制

Fig. 11 Clouds scenes simulation

从上面绘制的云场景可以看出, 采用本文方法所绘制的效果真实感较强。而且本文方法在绘制大规模云场景时更能体现它的优越性, 因为此时可以根据视点距仿真空间中心的距离调整所用的全局网格分辨率, 并保证仿真的实时性。

图 12 为与物体交互的云场景。

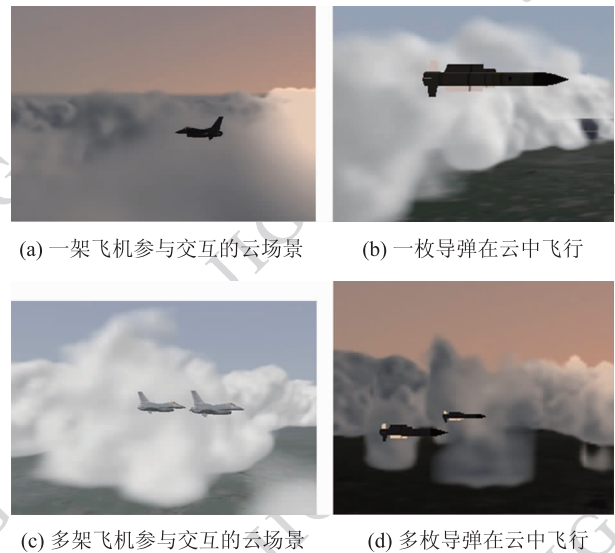


图 12 与物体交互的云场景

Fig. 12 Clouds scenes with objects

上述不同场景在绘制时的实时帧率统计如表 2 所示。从上面的场景图像以及表 2 的实时绘制帧率可以看出, 在特定的硬件条件下影响仿真速度的因素主要有两个: 仿真空间的分辨率和仿真空间中有无交互的对象。此外参与交互对象的多少以及仿真对象的复杂程度也对仿真的速度有着不同程度的影响。

表 2 中的数据均为网格分辨率不超过 $512 \times 512 \times 512$ 规模时的统计结果, 在绘制过程中预设最高帧率的 120 帧/s, 当超过该预设值时均以 120 帧/s

表 2 云场景在不同网格分辨率下的绘制帧率

Table 2 Rendering frame rates of clouds scenes in different grids resolutions

不同云场景	云朵形成	黎明时	大规模云	有飞机参与交互(1 架)	有飞机参与交互(2 架)	有导弹参与交互(1 枚)	有导弹参与交互(2 枚)	
不同网格分辨率下的绘制帧率/(帧/s)	256 × 256 × 128	120	120	约 105	约 108	约 101	约 120	约 106
	256 × 256 × 256	120	约 110	约 77	约 82	约 79	约 89	约 83
	512 × 512 × 256	120	约 54	约 29	约 32	约 31	约 38	约 40
	512 × 512 × 512	约 91	约 32	约 15	约 17	约 22	约 21	约 21

进行显示。从表2中统计的数据可以看出,当网格分辨率不超过 $512 \times 512 \times 512$ 规模时运用本文方法均可实现实时绘制,当仿真空间的分辨率更高时,仿真将难以实现实时绘制,出现失真现象。当采用 512^3 级别以下的仿真空间时,增加参与交互对象或是建模对象变复杂时,均能实现实时绘制。

4 结 论

在实现云的实时绘制中提出了多尺度自适应活动网格的方法,提出的方法具有简单、易于GPU实现的特点,并对不同分辨率全局和局部网格的选择方法,以及拆分或合并后的数据处理进行了详细阐述。对参与交互的物体建立OBB层次包围盒,物体所占用的网格单元不参与仿真计算,以此来提高仿真速度。通过云场景的仿真对所提出方法进行了有效性验证,本文方法要快于以往的纯粹改变全局网格分辨率的方法,也优于单纯改变局部网格分辨率的方法,而且本文方法不需要进行云数据的预先生成,真正实现了云场景的实时仿真。

本文方法不但可以用于云场景的仿真,对其他流体的仿真也同样具有适用性,这是下一步的研究工作。对云等流体的实时仿真中,有不固定数目的物体(含变形体)参与交互是研究的热点也是难点,在接下来的研究工作中,将把本文提出的方法推广到这样的流体场景仿真中。

参考文献 (References)

- [1] Miyazaki R, Dobashi Y, Nishita T. Simulation of cumuliform clouds based on computational fluid dynamics[C]// Proceedings of Eurographics 2002 Short Presentation. Aire-la-Ville, Switzerland: The Eurographics Association, 2002: 405-410.
- [2] Selle A, Rasmussen N, Fedkiw R. A vortex particle method for smoke, water and explosions[J]. ACM Transactions on Graph., 2005, 24(3): 910-914.
- [3] Dobashi Y, Matsuda Y, Yamamoto T, et al. A fast simulation method using overlapping grids for interactions between smoke and rigid objects[J]. Computer Graphics Forum, 2008, 27(2): 477-486.
- [4] Lentine M, Zheng W, Fedkiw R. A novel algorithm for incompressible flow using only a coarse grid projection[J]. ACM Transactions on Graph., 2010, 29(4): 114(1-9).
- [5] Lentine M, Cong M, Patkar S, et al. Simulating free surface flow with very large time steps[C]// ACM SIGGRAPH/Eurographics Symposium on Computer Animation. Aire-la-Ville, Switzerland: The Eurographics Association, 2012: 107-116.
- [6] English R E, Qiu L, Yu Y, et al. An adaptive discretization of incompressible flow using a multitude of moving cartesian grids[J]. Journal of Computational Physics, 2013, 254(12): 107-154.
- [7] Qiu H, Yang K, Chen Y, et al. Survey on realistic simulation of cloud[J]. Computer Science, 2011, 38(6): 14-19. [邱航, 杨珂, 陈瑜, 等. 云的真实感模拟技术综述[J]. 计算机科学, 2011, 38(6): 14-19.]
- [8] Miyazaki R, Yoshida S, Dobashi Y, et al. A method for modeling clouds based on atmospheric fluid dynamics[C]// Proceedings of Pacific Graphics. Washington DC: IEEE Computer Society, 2001: 363-372.
- [9] Harris M J. Real-time cloud simulation and rendering[D]. Chapel Hill: University of North Carolina, 2003.
- [10] Ren W, Liang X H, Ma S, et al. A real-time simulation method for large-scale 3D clouds[J]. Journal of Computer-Aided Design & Computer Graphics, 2010, 22(4): 662-669. [任威, 梁晓辉, 马上, 等. 大规模三维云实时模拟方法[J]. 计算机辅助设计与图形学学报, 2010, 22(4): 662-669.]
- [11] Stam J. Stable fluids[C]// Proceedings of SIGGRAPH'99. New York: Association for Computing Machinery, 1999: 121-128.
- [12] Fedkiw R, Stam J, Jesen H W. Visual simulation of smoke[C]// Proceedings of SIGGRAPH'01. New York: Association for Computing Machinery, 2001: 15-22.
- [13] Carlson M, Mucha P J, Turk G. Rigid fluid: animating the interplay between rigid bodies and fluid[J]. ACM Transactions on Graphics, 2004, 23(3): 377-384.
- [14] Guendelman E, Selle A, Losasso F, et al. Coupling water and smoke to thin deformable and rigid shells[J]. ACM Transactions on Graphics, 2005, 24(3): 973-981.
- [15] Griebel M, Dornseifer T, Neunhoffer T. Numerical simulation in fluid dynamics: a practical introduction[M]. Philadelphia: SIAM Monographs on Mathematical Modeling and Computation. Society for Industrial and Applied Mathematics, 1997.
- [16] Crane K, Llamas I, Tariq S. Real-time simulation and rendering of 3D fluids[M]. GPU Gems 3. New York: Addison Wesley Professional, 2007: 633-675.