



# 洪水淹没模拟三维可视化技术研究

张彪<sup>1,2</sup>, 程光<sup>1,2</sup>

(1.东南大学计算机科学与工程学院, 南京, 210000;

2.教育部计算机网络与信息集成重点实验室(东南大学), 南京, 211189)

**摘要:**洪水淹没模拟三维可视化直观展示洪水淹没演进情况,是现今防洪调度与决策的重要技术支撑。本文探讨了洪水淹没模拟三维可视化中的地理环境三维可视化,洪水淹没演进动态可视化和流场动态可视化三个问题,并提出了基于OSG和osgEarth三维渲染引擎的解决方案。通过应用本文提出的解决方案,实现了一个洪水淹没模拟三维可视化系统,达到了较好的洪水淹没模拟三维可视化效果。

**关键词:**洪水淹没模拟; 三维可视化; OSG(OpenSceneGraph); osgEarth; 流场; 粒子

## Study of Three-Dimensional Visualization Technology of Flood Inundation Simulation

Zhang Biao<sup>1,2</sup>, Cheng Guang<sup>1,2</sup>

(1. School of Computer Science and Engineering, Southeast University, Nanjing, 210000;

2. Key Laboratory of Computer Network and Information Integration(Southeast University), Ministry of Education, 211189)

**Abstract:** Three-dimensional visualization of flood inundation simulation will display the flood evolution process vividly. It can provide important technical support for flood control and decision-making. In this article, we have investigated three related problems about three-dimensional visualization of flood inundation process: 3d geographical environment modeling, dynamic visualization of flood evolution process and dynamic visualization of flow field. And we also have proposed solutions for these problems based on OSG and osgEarth 3d rendering engine. By applying our solutions, we have implemented a 3d visualization system of flood inundation simulation and have achieved better visualization effects.

**Key words:** Flood Inundation Simulation; 3D Visualization; OSG(OpenSceneGraph); osgEarth; Flow field; Particle

我国是洪水灾害频发的地区,几乎每年都要遭受各种或大或小的洪水灾害。在全国约有35%的耕地,40%的人口和70%的工农业生产经常受到洪水灾害的威胁,并且因洪水及其衍生灾害造成的损失居各种灾害之首<sup>[1]</sup>。

洪水灾害的频发性与高威胁性使得防洪体系建设具有重要意义。除了传统的防洪工程措施外,现今数字防洪体系的建设也越来越发挥出重要作用。所谓数字防洪,就是运用计算机构建研究区域洪水淹没的数值模型,并基于数值模型的计算结果对研究区域洪水淹没过程进行重现或是预演,从而为防洪调度决策,应急预案的定制等

提供技术支持和服务<sup>[2]</sup>。

为了达到较好的洪水淹没模拟效果,研究者们一直在进行不断的探索。随着三维可视化技术的出现,对洪水淹没演进过程进行三维可视化成为研究者们重点关注的方向<sup>[3]</sup>。

## 1 洪水淹没模拟三维可视化的主要内容

洪水淹没模拟三维可视化主要包含三个方面的内容:第一是洪水淹没模拟研究目标区域三维地理环境的构建,即通过三维可视化手段尽可能真实地表现研究区域地理环境;第二是洪水淹没演进过程的动态可视化,即通过动态可视化方式表现洪水在研究区域淹没演进的动态过程;第三是流场的动态可视化,这里所谓的流场是指研究

**作者简介:**张彪,男,硕士研究生, E-mail: zhangbiaocn@foxmail.com; 程光,男,教授, E-mail: gcheng@njnet.edu.cn

区域水流运动的空间分布,即各个位置处水流运动的动量(方向和流速)。综合应用实现上述三个方面的内容,使得真实再现或预演研究区域的洪水淹没过程成为可能。如图1所示展现了这三个方面的相互关系以及它们在洪水淹没模拟三维可视化中的位置。

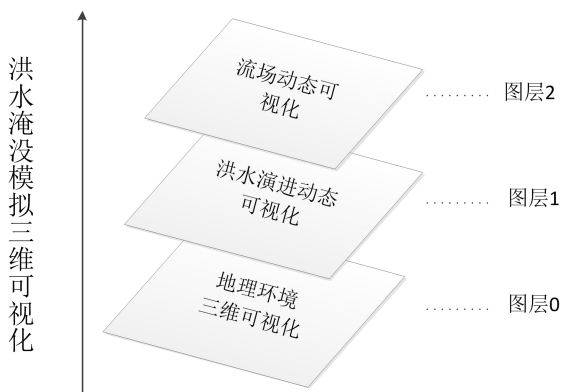


图1 洪水模拟三维可视化的主要内容

## 2 相关研究

对于地理环境三维可视化,研究者们提出了许多相关的技术和算法,并开发出了许多三维渲染引擎,使得对地理环境的三维可视化更加快速便捷。在[4][5]中,作者综合探讨了三维地理环境可视化中诸如地形建模,纹理生成,光照流水特效等问题,并介绍了地形建模的LOD(Level Of Detail, 细节层次)算法,程序合成纹理,天空盒渲染等问题的解决思路。在[6]中作者详尽探讨了LOD算法的设计与实现过程。在[7]中作者针对基于四叉树结构的LOD大规模地形建模算法进行了讨论与实现,并进行了实验验证,表明了该算法在渲染大规模地形时的高效性能。在[8]中作者以火星形貌三维建模与绘制为例,讨论并实现了基于OSG三维渲染引擎的火星形貌三维可视化,而OSG正是基于四叉树LOD算法实现大规模地形建模的。在[9]中作者基于多分辨率LOD模型技术,设计并实现了一种面向全球的卫星影像金字塔模型,在OSG中已有类似的构建影像金字塔模型的相关功能模块和工具。在[10]中作者介绍基于三维建模软件的数字城市建模中地形地物要素建模技术和方法。

对于洪水淹没演进动态可视化问题,有很多不同的解决思路。在[11]中作者介绍了基于ArcGIS地理信息系统软件平台的实现方式,这种

方式的优点是可以充分利用既有平台的空间查询与分析功能。在[12]中作者讨论了基于DEM数据的洪水淹没模拟及其可视化问题,这种方式以DEM数据作为输入,然后以程序方式对数据进行处理,计算出各时刻洪水的淹没范围,将淹没范围以图形形式时序地展现,以表现洪水淹没演进的动态过程。在[13]中作者讨论了一种基于水动力学模型的洪水淹没数值模拟方式,即基于水动力学数值模型计算研究区域内各节点在每一时刻的水位和水流动量数据,然后再根据计算结果对洪水淹没过程动态可视化,可视化可以基于原生的图形API或者功能完备的图形渲染引擎。这种方式由于有较为精确的数值计算模型的支持,因此表现洪水淹没演进过程也更准确。

对于流场动态可视化,由于其对流场特性的丰富表现力,自90年代以来就一直是研究者们关注的问题。较为常见的流场表示法有箭头标记法,流线法,路径轨迹法等,随着可视化技术的发展,研究者们也在不断提出新的流场可视化方法。在[17]中作者综合介绍了流场可视化的各种方法,并对它们加以实现对比。在[18]提出了一种线积分卷积的纹理图表示法,它能够有效地表征二维矢量流场,既能清楚直观地反映出流场总体分布,又能表达流场的细节。但这种方法的不足之处在于效率过低,以及不能表现流场瞬间时刻的流动方向,在[19][20]中作者分别提出了针对这一方法中存在的问题的改进方法。在[21]中作者综述了过去基于积分方法的流场可视化方法,包括线积分卷积法,粒子系统法等。[22]中对粒子系统法中的运动轨迹计算这一核心问题作了详尽阐述。

## 3 地理环境三维可视化

地理环境主要是指一个区域的地理位置,地形结构和地貌情况等自然因素以及社会经济因素的综合构成。现今对地理环境的三维可视化主要有三种途径:完全基于三维建模软件的方式,完全基于三维渲染引擎的程序开发方式,以及前两种结合运用的方式。本文中采用完全基于三维渲染引擎的程序开发方式对研究区域的地理环境进行三维可视化,采用的渲染引擎是OSG(OpenSceneGraph)和osgEarth<sup>[14-16]</sup>。

一般洪水淹没演进模拟的研究目标区域面积

从几十平方公里到几百平方公里不等,有的甚至达到上千平方公里,如此广阔的研究区域包含了大规模的地理信息数据,个别研究区域面积相对较小的,也会因为地形结构的复杂而包含大量的地理信息数据。因此,对这些研究区域进行地理环境三维可视化工作的主要挑战在于对大规模地理信息数据的管理和调度<sup>[5]</sup>,使之既能逼真地反映原始三维地理环境,还能保证流畅地进行交互式的漫游。在这些地理信息数据中又以反映研究区域地形结构的地形 DEM (Digital Elevation Model, 数字高程模型) 数据,反映研究区域地貌情况的地貌纹理数据和反映研究区域社会经济情况的城镇建筑模型数据最为重要,也最难管理和调度。

### 3.1 地形 DEM 数据的管理与调度

本文中地形数据的管理与调度基于一个简单事实,即我们观察到的现实场景,往往近处的物体详实清晰,远处的物体简略模糊。因此本文中通过 OSG 和 osgEarth 对四叉树结构的内置支持,通过对地形数据分层分块,建立地形分页数据库来管理地形数据,并通过 OSG 对分页数据库动态调度的支持,实现三维地形的高效渲染<sup>[6-8]</sup>。

1) 四叉树是一种层次数据结构,对地形数据进行层次化管理<sup>[6]</sup>。其根节点代表了整个地形区域,每一个内节点都有四颗子树,代表着该节点表示的地形分块后的四块地形。越靠近根节点的表示区域的地形越简单,绘制所需地形信息少,绘制效率高;越靠近叶子节点的表示相应区域地形拥有更多细节,但所需地形信息量大,绘制效率低。在进行交互式漫游时,根据区域地形的起伏以及与视点的距离,动态地选择相应的四叉树节点层次来绘制各区域地形。如图 2 所示是地形的分层分块,及其对应的四叉树结构,图中节点 T1, T21, T22, T23, T24, T31, T32, T33, T34, T4 组合起来表示某个状态下我们所看到的地形。

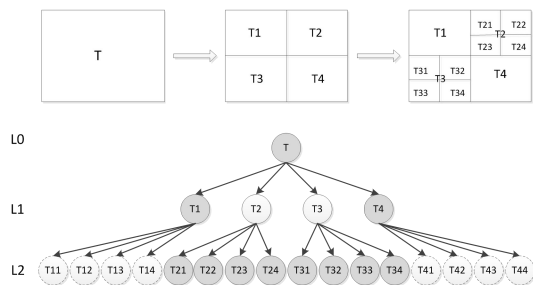


图 2 地形四叉树结构

2) 在进行大规模场景漫游时还会发生的一种情况就是,有些地形数据块可能暂时还未出现在场景中或者是已经在场景中出现了,由于系统内存中有限的容量限制,对这些数据要动态地加载到内存中或是从内存卸载,OSG 的分页数据库动态调度技术实现了这一功能,动态调度过程如图 3 所示。图中弃用对象表示需要从内存中卸载的地形数据块,新对象表示需要加载到内存中的地形数据块,编译操作指的是 OSG 中为节省重复绘制同一模型的时间而进行的优化操作。

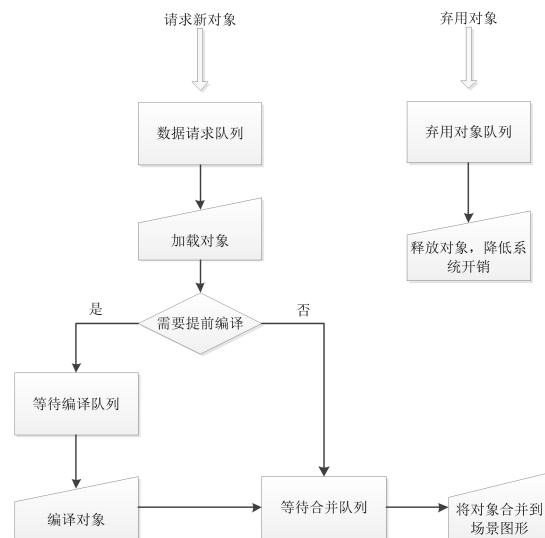


图 3 分页数据动态调度过程

### 3.2 地貌纹理数据的管理与调度

地貌纹理数据能够表达很多通过其他手段无法表现的细节,把它作为做地形 DEM 数据的一个重要补充,能够更加真实地反映一个区域的总体地理环境。现今地貌纹理数据主要有两种获取手段:程序合成和卫星影像。程序合成地貌纹理数据是指通过编写程序根据一定得规则生成研究区域的地貌纹理数据;卫星影像地貌纹理数据则是通过太空卫星所携带的高分辨率相机对研究区域拍摄照片获得的。显然卫星影像数据是完全真实的,比程序合成的数据更能反映研究区域的地理环境,因此本文中采用卫星影像数据来表达地貌纹理。但是采用卫星影像数据存在的问题在于一张覆盖广阔研究区域的高分辨率卫星影像图的大小有可能达到 GB 级别,由于系统内存的限制,不可能一次性把原始的数据全部装入内存,而且也没有必要,同样基于近详远略的事实,我们首先利用 OSG 对卫星图像进行分层,即构建卫星影像的金字塔模型,然后与地形结合进行分块,从

而形成完整的地形地貌分页数据库,再通过 OSG 的分页数据库动态调度机制进行调度。金字塔模型是一种多分辨率的层次模型<sup>[9]</sup>,对于一张分辨率较高,规模较大的影像图,以此图为基础采用倍率递降采样的方法进行构建,从而形成多个分辨率层次。假设原始影像的分辨率为  $R_0$ ,倍率为  $m$ ,则第  $k$  层的分辨率  $R_k$  为:

$$R_k = R_0 * m^{-k}$$

图 4 所示是中国湖南省洞庭湖区域卫星影像的金字塔分层结构,  $L_0$  为最低层,代表原始影像。

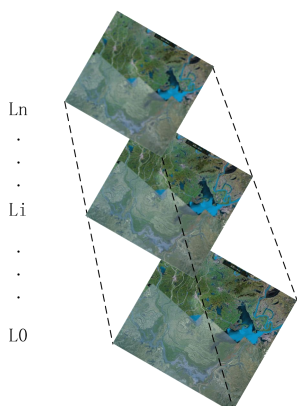


图 4 卫星影像金字塔分层结构

### 3.3 城镇建筑模型数据的管理与调度

城镇建筑模型数据一般包括城镇三维可视化建模所需要的建筑物三维模型,建筑物地理位置和建筑物表面纹理三类数据。传统上这三类数据是城镇建模必须的,但是对于一个城镇而言,这三类数据往往数据量较大,并且来源不同,导致要管理的数据种类也较多。比如,传统上建筑物三维模型通常是通过三维建模软件(如 CAD)预先进行设计,然后保存到特定格式的文件中,再导入到三维可视化程序的场景进行渲染;地理位置数据通常是通过地理信息系统方面的软件来获取的,这种类型的数据一般是保存在另一种特定格式的文件中(如 Shapefile)。在本文中我们基于 osgEarth 对地理信息数据的支持,采用了另一种完全不同的方案,即不预先对建筑物三维模型进行建模,然后再根据地理位置数据进行定位,最后添加建筑物表面纹理进行渲染。而是将与建筑物模型的相关的数据,如高度,地理位置,建筑物类型等,全部作为属性数据存储于 Shapefile 文件中。渲染时根据这些属性数据在相应位置渲染相应高度的规则或非规则立方体来

实现对建筑物建模。尽管这样的建模方式会使得很多建筑物损失一些细节,但这已经足以表现一个城镇的总体建筑分布情况。并且,本文中还是基于 osgEarth 对 XML 索引的支持在建筑物表面纹理数据层之上创建纹理数据的 XML 索引文件。在进行渲染时,系统从索引中选取相应纹理,由于城镇中很多建筑物(比如同一个居民社区的建筑物)表面的纹理是相同的,因此这可以使得在不同建筑物之间共享表面纹理,从而减少了纹理数据加载量,提高系统性能。图 5 所示展示了本文城镇建筑的建模过程。

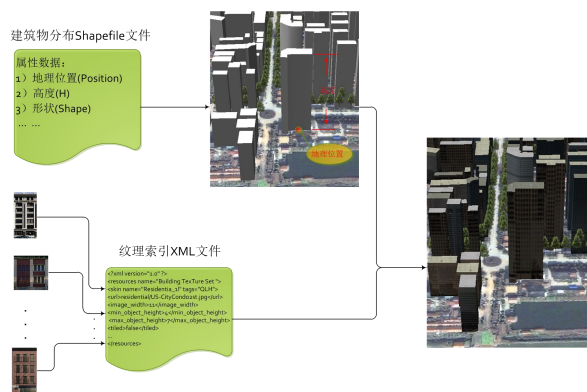


图 5 城镇建筑实时建模

## 4 洪水淹没演进动态可视化

洪水淹没演进是一个动态的过程。本文中洪水淹没演进的可视化主要是根据水动力学模型的计算结果进行。将水动力学模型的计算结果数据按照时序加以展现即可完成洪水淹没演进的动态可视化,如图 6 所示展现了洪水淹没演进动态可视化算法流程。

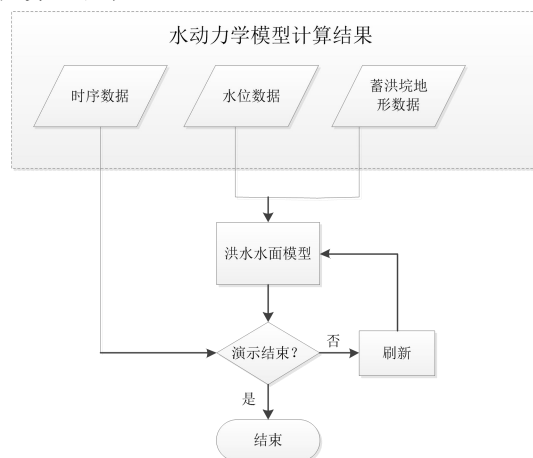


图 6 洪水淹没演进动态可视化算法流程  
从图 6 中可知本文中只构建洪水水面模型,

这是由于我们一般看到的都是洪水的水面部分，因此为了在达到比较真实的演进效果的同时减少系统的性能开销，我们只对洪水的水面进行构建，进行整个洪水水体（包括水面和水体内部）的构建是复杂而没有必要的。

本文中构建洪水水面时，我们把它看做一张覆盖在地形上的非规则三角形网格图层，初始时，洪水网格图层与地形网格完全相同，并紧密贴合在地形网格图层上面，网格中各顶点的高程属性值等于该点处地形的高程，当洪水过来时，动态改变节点的高程属性，从而达到模拟洪水演进的效果，如图 7 所示。图中为了更清晰地表示对洪水演进进行动态可视化的方法，我们用网格表示洪水图层，用颜色表示水深。在图 7 中 T0 时刻洪水网格图层（红色所示）与地形图层（暗黄色所示）紧密贴合在一起。从 T1 时刻开始洪水开始演进，此时洪水图层网格根据水位数据发生变化，洪水通过闸口后向外围扩散，使相邻的网格顶点高程也开始依次变化，从而表现出洪水波浪式前进的形态。

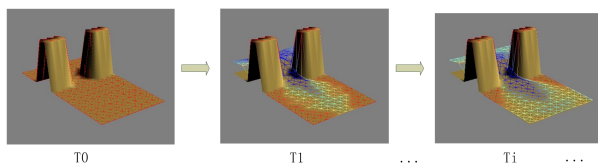


图 7 洪水淹没演进动态可视化

## 5 流场动态可视化

流场是一个广泛存在于计算流体动力学领域的概念，表示某一时刻流体运动的空间分布，通常以矢量场的形式出现。流场大致上可以分为两类：稳定流场（Steady Flow）和不稳定流场（Unsteady Flow）。稳定流场即空间上各点的流体运动动量不随时间变化而变化的矢量场；不稳定流场则相反，即各点处的流体运动动量是随时间变化的。洪水流场就是一个表征洪水流动空间分布特性的不稳定流场。类似于风场等许多其它类型的流场一样，洪水流场也是肉眼不可见的，对其进行可视化有助于定性定量地研究洪水的流动特性。本文将洪水流场定义为研究区域  $D$  的一个映射函数：

$$\vec{v} = F(P, t)$$

即对研究区域  $D$  中的每一个采样点  $P$  都赋予一

个表征该点处在时刻  $t$  时的水流动量  $\vec{v}$ 。本文在进行水动力学模型建模计算时空间上只考虑洪水的平行流动，因此矢量  $\vec{v}$  也就是一个二维矢量，流场也就相应为二维流场。

### 5.1 常见流场可视化方法

洪水流场动态可视化的方法一直是洪水淹没模拟领域的研究热点，现今研究者们已经提出了多种进行洪水流场动态可视化的方法，较有代表性的有以下四种<sup>[21]</sup>：

#### 1) 箭头标记法

箭头标记法是指通过箭头所指的方向来表示洪水流场相应点处的水流方向，箭头的长短表示相应点处的水流流速，如图 8 所示。

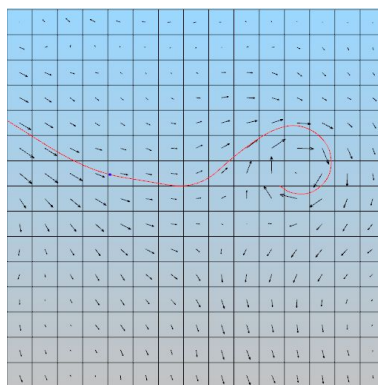


图 8 流场箭头标记法（FlowVis 生成）

#### 2) Streamline

一条 Streamline（流线）是与它所经过的流场各点处的矢量  $\vec{v}$  相切的曲线，Streamline 适用于稳定流场的表示，它指出了稳定流场中水流粒子运动所遵循的轨迹。多条 Streamline 组合构成流场的 Streamline 表示法，如图 9 所示。

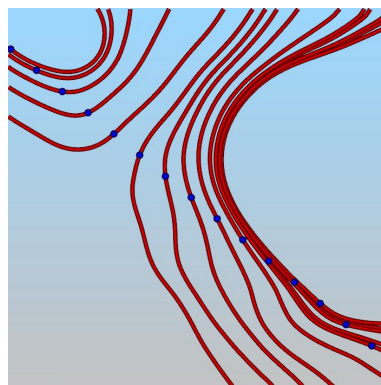


图 9 流场 Streamline（FlowVis 生成）

### 3) Pathline

Pathline (路线) 是与 Streamline 相似的概念, 不过它适用于不稳定流场。水流粒子在经过不稳定流场时其运动轨迹不是预先可以完全确定的, 而是随着时间不断修正运动路线, 如图 10 所示。

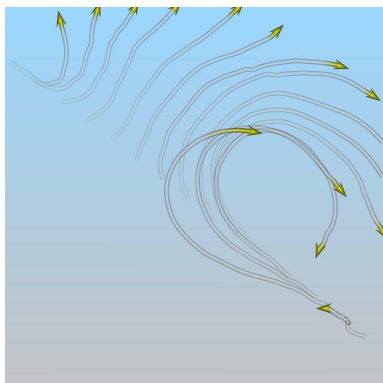


图 10 流场 Pathline (FlowVis 生成)

### 4) LIC: Line Integral Convolution

LIC (线积分卷积) 是一种基于纹理图的表示方法。它能够有效地表征二维矢量流场, 既能清楚直观地反映出流场的总体分布, 又能表达流场的细节。其基本原理基于运动模糊的思想, 可概述为利用一维低通卷积核双向对称地沿流方向卷积白噪声纹理, 最终合成矢量纹理<sup>[18-20]</sup>。如图 11 所示。

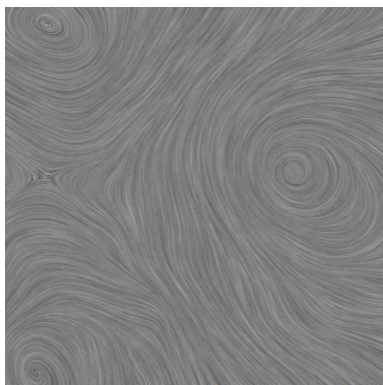


图 11 流场 LIC (FlowVis 生成)

上述四种方法有一个共同的特点是相对更适用于规则网格流场, 当应用于不规则网格流场时需加以改进。所谓规则网格流场是与不规则网格流场相对的概念, 不规则网格流场是指采样点位置不规则, 流场空间用不规则网格 (一般为不规则三角网格) 表示的流场, 如图 12 所示。图中箭头表示采样点处某个时刻的水流动量。

本文中的洪水流场是通过基于不规则三角网

格的水动力学模型计算得来的, 它是一个不规则网格流场。通过前述的讨论我们还知道洪水流场

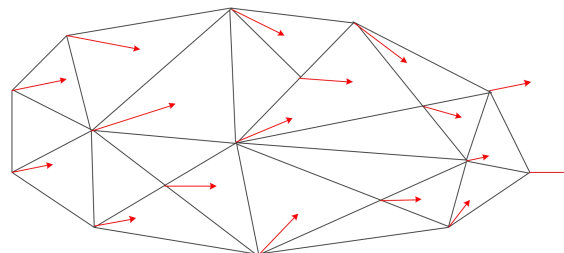


图 12 不规则网格流场示意图

是不稳定流场, 对于不规则网格的不稳定洪水流场, 前述几种动态可视化方法以及它们的后续改进方法, 如 FastLIC, OLIC 等, 都需要加以转换后才适用<sup>[21]</sup>, 并且上述几种方法由于对研究区域内的多数点都需要进行相应计算, 导致性能较低, 无法适应较大规模的洪水流场可视化的需求。另外, 这几种方法一般只能表现流场全局的时序动态, 无法表现特定时刻的流场动态。

## 5.2 基于 OSG 粒子系统的流场动态可视化

针对上述方法存在的问题, 本文采用了一种基于 OSG 粒子系统的洪水流场动态可视化方法, 以在简化一些不必要的细节的同时还能达到较好的可视化效果, 不但能表现流场全局的时序动态, 还能表现特定时刻的流场动态。基于粒子系统的洪水流场动态可视化算法流程如图 13 所示。

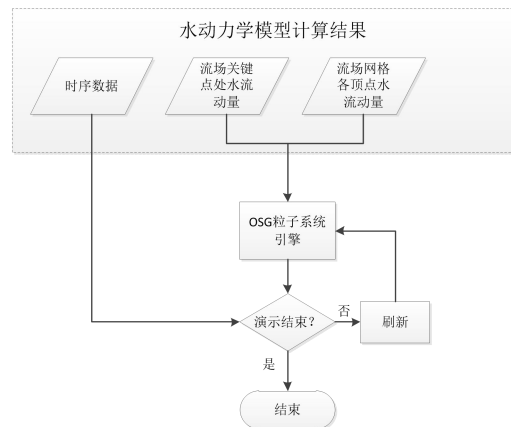


图 13 基于粒子系统流场动态可视化算法流程

基于 OSG 粒子系统的流场动态可视化有两个要解决的核心问题: 种子点选取问题和粒子运动轨迹计算问题, 下面分别加以讨论<sup>[21]</sup>。

### 5.2.1 种子点选取

粒子系统种子点选取对系统的可视化效果和性能有重要影响。种子点数不宜过多, 否则会降低系统性能; 种子点间的间距不能过大或过小,

过大则不能完整表现流场的整体特性，过小则可能会造成粒子运动轨迹相互干扰。对于规则网格流场，通常采用等像素距离分布的方式确定种子点，即在二维方向上隔固定像素距离选取种子点。但这种方法不适用于不规则网格流场，因为等像素间距处不一定都有采样点存在。本文中引入如下两个概念：

- 1) 顶点连通度：共享该顶点的三角单元个数。
- 2) 关键点：不规则三角网格中连通度大于或等于设定阈值的顶点。

如图 14 所示，连通度阈值设为 4，圆圈标记的顶点即为关键点。

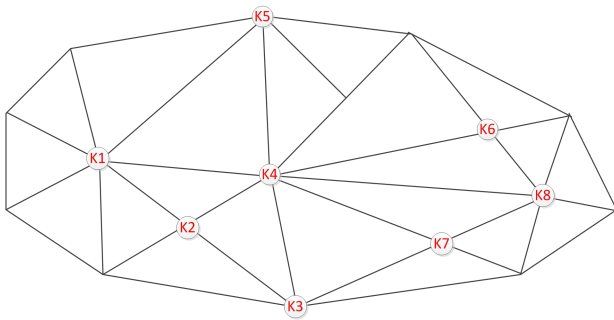


图 14 不规则三角网格的关键点

基于以下几点理由：

- 1) 非规则三角网格通过单元格的疏密来表现区域地形的复杂情况；
- 2) 单元格较密区域往往地形结构较为复杂，需要更为细致的体现；
- 3) 地形结构复杂区域的流场模式较为复杂，这些区域的流场可视化需要更多的采样点来表现；
- 4) 单元格较密区域的顶点的连通度一般较其它区域大。

本文选取关键点作为粒子种子点。

### 5.2.2 粒子运动轨迹计算

运动轨迹是指从种子点发出的粒子在流场中运动的路线，粒子运动轨迹主要通过数值计算方法确定<sup>[21-22]</sup>。本文中研究的洪水流场是二维的，二维流场的粒子运动轨迹数值计算方法主要有适用于规则网格的 Euler 法，Runge-Kutta 法，以及适用于不规则网格的拉格朗日法等。粒子运动轨迹的数值计算是一个非常耗费时间的过程，本文中为了在保证准确的前提下提高计算效率，提出了一种简化的数值计算方法。计算过程中粒子运动轨迹的确定基于下面的控制方程：

$$\frac{dp}{dt} = \vec{v}(p, t), \quad p(t_0) = p_0, \quad t \in [t_0, t_n]$$

其中， $p$  表示粒子在流场中的位置，在时刻  $t_0$  时，

粒子位于种子点  $p_0$  处； $\vec{v}(p, t)$  表示流场位置  $p$  处在  $t$  时刻的水流动量。

通过对控制方程的数值积分计算，我们可以构建从种子点  $p_0$  出发的粒子的完整运动轨迹。假定

$p_k$  表示粒子在  $t_k$  时刻所处的位置，那么粒子在

$t_{k+1}$  时刻的位置  $p_{k+1}$  可通过下式计算：

$$p_{k+1} = p_k + \vec{v}(p_k, t_k) \cdot \Delta t$$

其中， $\Delta t = t_{k+1} - t_k$ 。 $\Delta t$  的取值不能太小，否则粒子大部分时候都处在网格的同一个单元格中，也不能取值过大，否则会造成粒子在一个时间段内越过运动轨迹上某些单元格的情况。

在这种步进式的计算过程中，不光需要确定下一时刻粒子所处的新位置，还需要确定粒子在新位置处的运动动量。本文中对新位置处运动动量的确定采用平均权重线性插值法。所谓平均权重线性插值，即首先确定粒子的新位置落在不规则网格的哪个单元格之中，然后不管新位置处于该单元格的哪个区域，都将单元格的各顶点处的水流动量赋予同样权重进行线性插值，以确定粒子位置处的动量。简单来说，就是落在某个单元格内的粒子的动量都等于该单元格中心点在相应时刻的动量。对于三角单元格而言，假定三个顶

点处的动量分别为  $\vec{v}_1, \vec{v}_2, \vec{v}_3$ ，则单元格中心

点处的动量  $\vec{v}$  通过下式计算可得：

$$\vec{v} = \frac{1}{3} \cdot (\vec{v}_1 + \vec{v}_2 + \vec{v}_3)$$

如图 15 所示是三角单元格的平均权重线性插值方法，图中满足  $\vec{v}(p_{k+1}, t_{k+1}) = \vec{v}$ 。

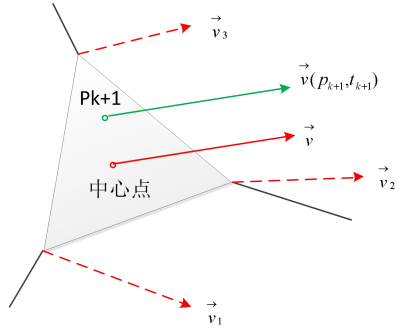


图 15 三角单元格平均权重线性插值

在判断粒子新位置处于哪个单元格内时，存在两种边界情况需要特殊处理。

1) 粒子落在单元格的边界上：当进行计算时，如果无法判断粒子位于哪个单元格内时，有可能粒子落在了单元格的边界上，这时按照原来的运动方向将粒子往前移动一小段距离，如果此时能够判断粒子位于哪个单元格内则回归正常处理，否则粒子一定已经落在研究区域  $D$  外了，则转到第二种情况处理；

2) 粒子落在研究区域  $D$  外：如果断定粒子已经落在研究区域  $D$  之外，那么粒子不应该继续存活，立即移除该粒子，即使它的预先设定的存活时间还未耗尽。

综上所述，对于以图 14 所示不规则三角网格中的关键点  $K_1$  为种子点的粒子在某个时刻的运动轨迹如图 16 所示，图中实线部分为该粒子的运动轨迹。

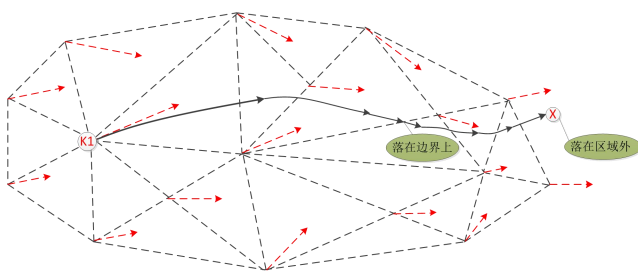


图 16 粒子运动轨迹示意图

## 6 实验分析

本文以中国湖南省洞庭湖区域钱粮湖蓄洪坑作为研究目标区域，实现了一个洪水淹没模拟三维可视化系统，实验系统框架如图 17 所示。

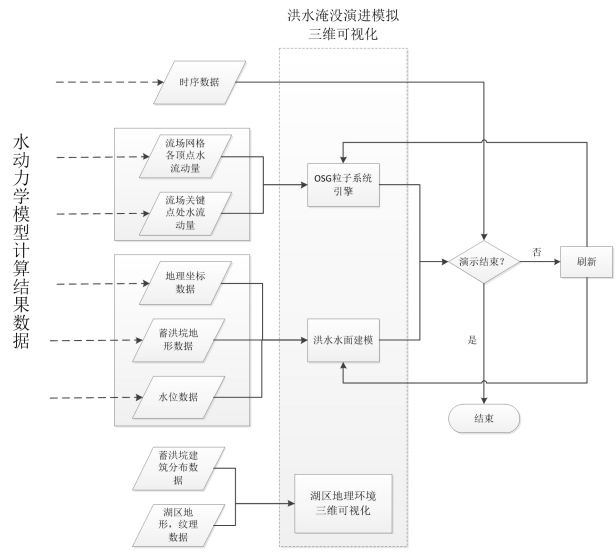


图 17 实验系统框架图

实验系统可视化效果如图 18，图 19 所示。图 18 展示了蓄洪坑地理环境三维可视化中的城镇建模渲染效果。图 19 展示了蓄洪坑在进水闸口处水位保持为 34.5 米时蓄洪 38 小时后的洪水演进分布和流场的可视化渲染效果。

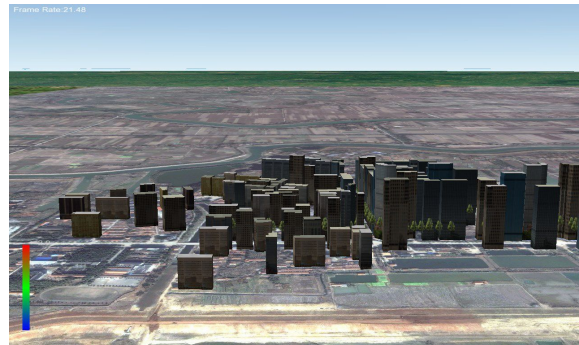


图 18 地理环境三维可视化

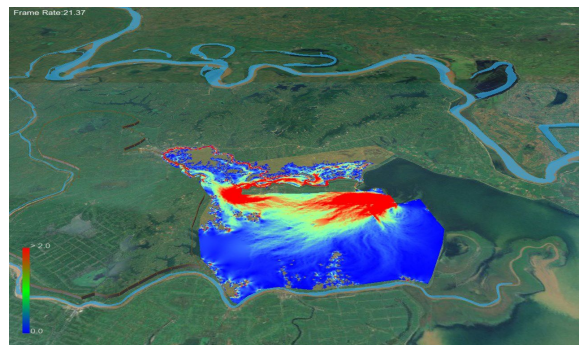


图 19 洪水淹没演进和流场可视化

从这两个图中可看出，实验系统实现了蓄洪坑区域的地理环境三维可视化，洪水淹没演进动态可视化和流场动态可视化。在保证良好可视化





效果的同时保持帧率在 20 以上,可以保证流畅地进行交互式漫游。

## 7 结论

本文针对洪水淹没模拟三维可视化中的地理环境三维可视化,洪水淹没演进动态可视化和流场动态可视化三个问题进行了详尽探讨,并分别提出了基于 OSG 和 osgEarth 三维渲染引擎的地理环境三维可视化解方案,基于 OSG 三维渲染引擎的洪水淹没演进动态可视化解方案,以及基于 OSG 粒子系统的流场动态可视化解方案。通过应用本文提出的解方案,实现了一个洪水淹没模拟三维可视化系统,并以洞庭湖区域钱粮湖蓄洪垸作为研究目标区域,对实验系统进行验证,实验结果表明系统较为逼真流畅地实现了洪水淹没模拟三维可视化效果。

## 参考文献

- [1] 胡四一, 施勇, 戚晨等. 1998 年长江中下游洪水系统反演及高洪水位形成原因探讨[J]. 水科学进展, 1999, 10(3): 242-250.
- [2] 张尚弘, 王兴奎, 易雨君等. 防洪数字仿真平台总体框架及关键技术研究[J]. 应用基础与工程学报, 2009, 17: 49-57.
- [3] 李云, 范子武, 吴时强等. 大型行蓄洪区洪水演进数值模拟与三维可视化技术[J]. 水利学报, 2005, 36(10): 1158-1164.
- [4] Trent Polack. Focus On 3D Terrain Program[M]. Ohio, USA: Premier Press, 2003.
- [5] 张娟. 大规模复杂地形模型的实时绘制技术研究[D]. 南京: 河海大学, 2007
- [6] Peter Lindstrom, David Koller, William Ribarsky, *et al.* Real-Time, Continuous Level of Detail Rendering of Height Fields[C]. In Computer Graphics (Proceedings SIGGRAPH '96), pages 109-118, 1996.
- [7] 万定生, 龚汇丰. 一种基于二叉树的大规模地形实时生成算法[J]. 计算机工程与应用, 2005, 33: 186-188.
- [8] 梁雅港, 周扬, 张勇等. 基于 OSG 的火星形貌 3 维建模与绘制[J]. 测绘与空间地理信息, 2011, 34(1): 53-56.
- [9] 杜莹, 武玉国, 王晓明等. 全球多分辨率虚拟地形环境的金字塔模型研究[J]. 系统仿真学报, 2006, 18(4): 955-958, 967
- [10] 杨建思, 杜志强, 彭正洪等. 数字城市三维景观模型的建模技术[J]. 武汉大学学报(工学版), 2003, 36(3): 37-40.
- [11] 孙君, 奚赛英, 尤迪等. 基于 ArcGIS 的洪水淹没分析与三维模拟[J]. 城市地质, 2012, 7(3): 31-33.
- [12] Aixia Zhou, Lianfeng Gao. A "volume method" for simulation of flood scene[C]. 水资源综合管理地理信息国际会议, January 2012.
- [13] Yu.V.VASSILEVSKI, K.D.NIKITIN, M. A. OLSHANSKI, *et al.* CFD technology for 3D simulation of large scale hydrodynamic events and disasters[J]. Russian journal of numerical analysis and mathematical modelling, 2012, 27(4).
- [14] Rui Wang, Xuelei Qian. OpenSceneGraph 3.0 Beginner's Guide[M]. Birmingham, UK: Packt Publishing Ltd, 2010
- [15] Rui Wang, Xuelai Qian. OpenSceneGraph 3 Cookbook[M]. Birmingham, UK: Packt Publishing Ltd, 2012.
- [16] 王锐, 钱学雷. OpenSceneGraph 三维渲染引擎设计与实践[M]. 北京, 中国: 清华大学出版社, 2009.
- [17] Tutorial: Introduction to 2D Flow Field Visualization[OL]. <http://www.cs.mtu.edu/~chaoliw/2dflowvis.html>.
- [18] Brian Carbral, Leith Leedom. Imaging Vector Fields Using Line Integral Convolution[C]. In Computer Graphics (Proceedings SIGGRAPH '93), volume 27, pages 263-272, August 1993.
- [19] Detlve Stalling, Hans-Christian Hege. Fast and Resolution Independent Line Integral Convolution[C]. Proceedings of the 22nd annual conference on Computer graphics and interactive techniques, pages 249-256, September 1995.
- [20] R. Wegenkittl, E. Groeller, W. Purgathofer. Animating Flowfields: Rendering of Oriented Line Integral Convolution[C]. In Computer Animation '97, pages 15-21, June 1997.
- [21] Tony McLoughlin, Robert S. Laramée, Ronald Peikert, *et al.* Over Two Decades of Integration-Based, Geometric Flow Visualization[J]. Computer Graphics Forum, 2010, 29(6): 1807-1829.
- [22] Numerical Methods for Particle Tracing in Vector Fields[OL]. <http://graphics.cs.ucdavis.edu/~joy/ecs277/other-notes/Numerical-Methods-for-Particle-Tracing-in-Vector-Fields.pdf>