

Анимационная визуализация трехмерных векторных полей

Сельвачев Александр
Московский физико-технический институт (государственный университет)
Москва, Россия
Аксенов Андрей
Институт автоматизации проектирования РАН
Москва, Россия
Клименко Станислав
Институт физики высоких энергий
Протвино, Россия

Аннотация

Рассмотрена группа многочастичных анимационных методов и предложен единый подход к их описанию. Приводится описание системы визуального анализа, разработанной на базе этого подхода.

Ключевые слова: анимация, многочастичные методы, визуальный анализ.

1. ВВЕДЕНИЕ

Среди методов визуализации векторных полей можно выделить группу так называемых многочастичных анимационных методов [1] – [3] (в дальнейшем – МЧА методы). Их геометрическим представлением является движение множества маркеров в объектном пространстве. Методы этой группы находят применение при анализе решений задач из разных разделов науки. В данной работе они рассматриваются применительно к течению жидкости.

При всем внешнем многообразии, предложенные разными авторами методы имеют больше общих черт, чем различий. Настоящая работа направлена на выявление общих черт предложенных ранее методов (п.2) и ставит целью разработку единого подхода к рассмотрению и реализации МЧА методов визуализации векторного поля. Изложенные в статье идеи воплощены в подсистеме визуального анализа системы *FlowVision* (п.4).

2. ОБЩИЙ ПОДХОД К РАССМОТРЕНИЮ МЧА МЕТОДОВ

Рассмотрим расчетную область гидродинамического течения, представляющую собой произвольную геометрическую фигуру, вообще говоря, неодносвязную. Зададим по некоторому правилу конечное множество точек внутри этой области и поставим в соответствие каждой точке маркер. Будем называть эти точки *точками рождения* маркеров. Пусть в расчетной области находится N точек рождения маркеров. Выберем n из них, которые начнут движение в начальный момент времени. Назовем

множество этих маркеров *множеством “рожденных”* маркеров, а остальные $N-n$ – *множеством “нерожденных”* маркеров. Будем характеризовать каждый маркер еще двумя определяющими параметрами: *временем жизни* и *жизненным объемом*. Под жизненным объемом будем понимать некоторую (в общем случае, произвольную) геометрическую фигуру, являющуюся подмножеством расчетной области и содержащую точку рождения маркера. Отрисовка маркера на экране происходит при выполнении двух условий:

- ♦ Время, прошедшее с начала движения маркера не превысило его время жизни
- ♦ Маркер находится внутри своего жизненного объема

Если маркер из множества “рожденных” (“рожденный” маркер) перестает удовлетворять одному из этих условий, то он переносится в множество “нерожденных”. Взамен него какой-нибудь “нерожденный” маркер становится “рожденным”, то есть начинает движение из своей точки рождения. Посредством двух этих множеств моделируется случайное рождение маркеров в области.

Таким образом, создан подход к рассмотрению МЧА методов, в котором маркеры задаются своей точкой рождения, временем жизни и жизненным объемом. Большинство описанных в литературе МЧА методов можно трактовать как частные случаи созданного подхода. Для этого достаточно конкретизировать правила задания параметров маркера. В п.3 даны характерные примеры МЧА методов и для каждого из них указаны правила задания параметров.

3. НЕКОТОРЫЕ МЕТОДЫ МНОГООБЪЕМНОЙ АНИМАЦИИ

Приведем несколько методов многообъемной анимации. Названия всех описанных в этом пункте методов введены для облегчения ссылки на них.

- ♦ пример “Дождь”: Расчетная область рассечена набором параллельных эквидистантных плоскостей, на каждой из которых в узлах прямоугольной сетки выбраны точки рождения маркеров. Жизненный объем маркера ограничен

двумя соседними плоскостями и расчетной областью. Время жизни маркера не ограничено. Метод удобен для наблюдения возвратных и застойных зон в расчетной области.

- ◆ пример “Капля”: В расчетной области выбирается точка и вблизи нее со сферически-симметричной функцией распределения размещаются точки рождения маркеров. Жизненный объем каждого маркера – вся расчетная область, время жизни не ограничено. Метод дает возможность наблюдать линии тока, выходящие из некоторого объема жидкости.
- ◆ пример “Вспышки”: Внутри всей расчетной области генерируются точки рождения маркеров. Принципы генерации могут быть различными. В существующей реализации, это – узлы прямоугольной сетки, слегка смещенные случайным образом. Как и в предыдущем примере, жизненный объем маркера – вся расчетная область; время жизни ограничено заранее заданным значением, общим для всех маркеров. В начальный момент времени “рождается” заранее заданный процент маркеров, остальные остаются “нерожденными”. Выбор “родившихся” маркеров производится случайным

образом. Метод дает представление о движении жидкости в целом и является аналогом экспериментальной визуализации движения прозрачной жидкости с отражающими свет частицами.

Рассмотренные методы реализованы в программном комплексе *FlowVision* в рамках общего подхода к группе МЧА методов. В следующем пункте находится описание *FlowVision* и рассматриваются детали реализации, связанные с МЧА методами.

4. ВИЗУАЛИЗАЦИЯ ГИДРОДИНАМИЧЕСКОГО ТЕЧЕНИЯ С ПОМОЩЬЮ МЧА МЕТОДОВ В СИСТЕМЕ *FLOWVISION*

Программный комплекс *FlowVision* создан для решения задач вычислительной гидродинамики (ВГД), возникающих в промышленных приложениях. При решении задач используется декартовая расчетная сетка с локальным измельчением.

Подсистема визуального анализа системы *FlowVision* (далее – визуализатор) использует **OpenGL** для отображения 3D-сцен.

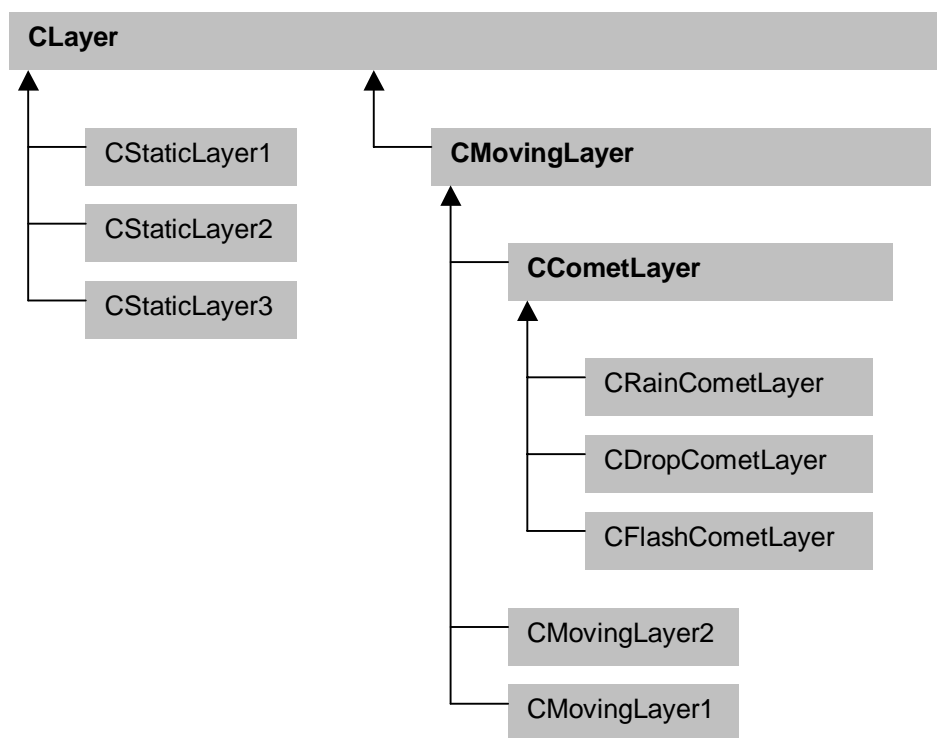


Рисунок 1. Организация классов-слоев

Каждому Windows-окну визуализатор сопоставляет объект класса **CPainter**, который работает с контекстом окна и осуществляет весь вывод в него. Каждый объект класса **CPainter** имеет свой список

слоев отображения. После установки параметров сцены, **CPainter** вызывает поочередно методы отрисовки каждого из своих слоев.

(В данном случае, слой – структура данных, имеющая свое геометрическое представление и метод отрисовки). Все слои – наследники абстрактного класса **CLayer**.

```
class CLayer
{
public:
    virtual void Draw(void) = 0;
    virtual void Recreate(CRegion&) = 0;
};
```

Метод **Draw()** вызывается при отрисовке сцены, к которой данный слой принадлежит. Метод **Recreate()** вызывается при изменении параметров слоя (включая метод отрисовки). Класс **CRegion** содержит информацию о геометрии расчетной области и расчетных данных задачи.

На **рис.1** классы **CLayer1**, **CLayer2**, **CLayer3** заключают в себе статические слои, то есть слои без анимации, например: плоскость отсечения, изоповерхность, вся расчетная геометрия. Абстрактный класс **CMovingLayer** – предок всех слоев, содержащих анимацию.

```
class CMovingLayer
{
public:
    virtual void Step(CRegion&) = 0;
};
```

Виртуальный метод **Step()** вызывается у слоев – потомков **CMovingLayer** при сигнале таймера, встроенного в **CPainter**. От частоты сигналов таймера зависит частота смены кадров в окне.

Абстрактный класс **CCometLayer** содержит список маркеров с ассоциированными параметрами – точкой рождения, временем жизни и жизненным объемом. Список заполняется в методах **Recreate()** наследников **CCometLayer**. В классе **CCometLayer** дана конечная реализация методов **Step()** и **Draw()**:

⇒ метод **CCometLayer::Step()** смещает маркер в пространстве в соответствии со значением векторного поля в точке нахождения маркера и проверяет условия видимости маркера. При нарушении условий маркер переносится из множества “рожденных” в множество “нерожденных”. Вместо него начинается движение один из “нерожденных”.

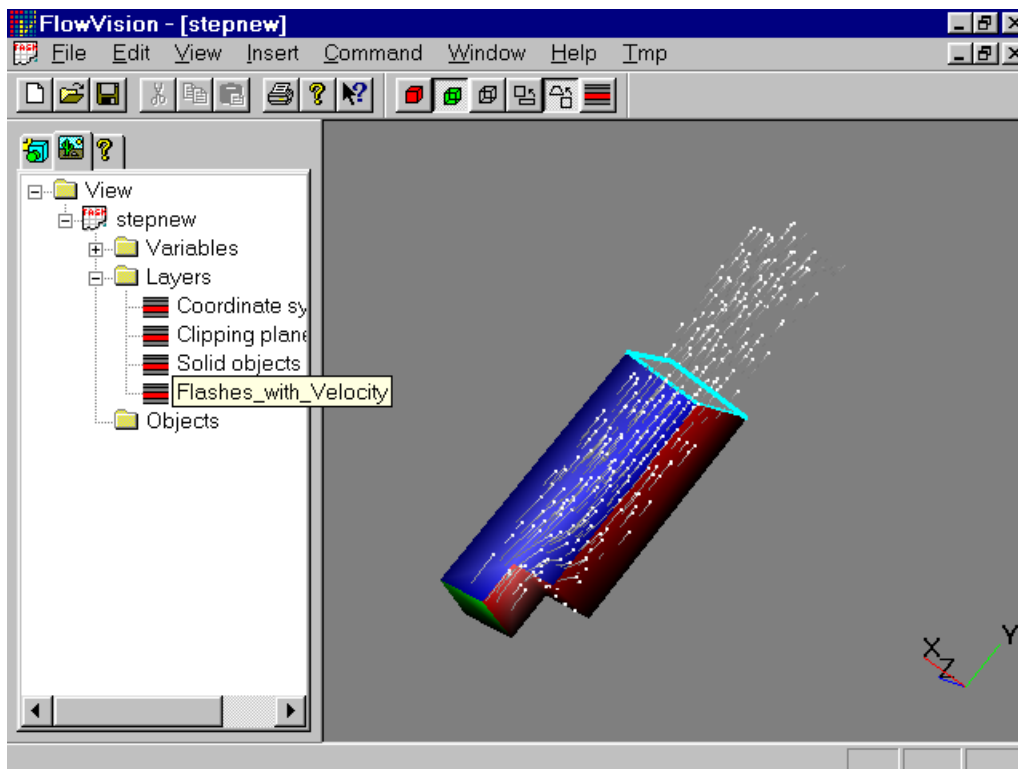


Рисунок 2. Метод “Вспышки”

⇒ метод `CCometLayer::Draw()` отображает все “рожденные” маркеры в **OpenGL** контекст. В существующей реализации маркеры визуализируются в виде ярких точек с затухающими “хвостами”, лежащими на пройденной маркером траектории. Такой вид маркера создает 3D-эффект, достаточный для определения направления поля в точке, а также позволяет сравнивать абсолютные значения поля в двух точках по длине “хвоста”.

Примерам п.2 соответствуют:

- ◆ примеру “Дождь” – `CRainCometLayer`
- ◆ примеру “Капля” – `CDropCometLayer`
- ◆ примеру “Вспышки” – `CFlashCometLayer`. На рис.2 показано, как выглядит реализация метода из этого примера применительно к задаче T-образного смесителя газа.

В этих классах дана конечная реализация метода `Recreate()`, который занимается заполнением списка маркеров в соответствии с изложенными в примерах принципами.

Таким образом, иерархия классов слоев использует общий подход к рассмотрению МЧА методов, описанный в п.2. Добавление нового метода достигается порождением еще одного класса-потомка `CCometLayer` и конкретизации его метода `Recreate()`.

5. ЗАКЛЮЧЕНИЕ

В работе был предложен новый взгляд на группу многочастичных анимационных методов. Было разобрано, как уже реализованные методы встраиваются в новую схему. Описана готовая реализация этой схемы и даны спецификации ключевых классов. Достигнутые результаты позволяют существенно упростить как определение новых методов группы, так и их реализацию.

6. ЛИТЕРАТУРА

[1] < Hin, A.J.S., 1994 >. < *Visualization of Turbulent Flow*. CIP-DATA KONINKLIJKE BIBLIOTHEEK, DEN HAAG >.

[2] < Sims, K., 1990 >. < *Particle animation and Rendering Using Data Parallel Computation*. Computer Graphics 24(4), pp. 405-413 >.

[3] < Stolk, J., J.J. van Wijk, 1992 >. < *Surface-Particles for 3D Flow Visualization*, Advances in Scientific Visualization, Springer Verlag, pp. 119-130 >.

Авторы:

Александр Сельвачев, студент
Московского физико-технического института
Институтский пер. 9, Долгопрудный М.о., Россия
Телефон: (095) 250-92-51

E-mail: flowvision@glas.apc.org

Андрей Аксенов, старший научный сотрудник
Института автоматизации проектирования РАН
2-я Брестская, 19/18, Москва, Россия
Телефон: (095) 250-92-51

E-mail: flowvision@glas.apc.org

Станислав Клименко, нач. отдела
Института высоких энергий РАН
Протвино, Россия

Телефон: (277) 1-39-38

E-mail: klimenko@mx.ihep.su

Alexander Selvachov, Andrey Aksenov,
Stanislav Klimenko

Visualization of 3D vector fields by animation

The group of multiparticle animation methods has been considered and new approach has been offered for its description. Upon the base of this approach a system of visual analysis has been created.

Keywords: *animation, multiparticle methods, visual analysis.*