



УДК 519.6

С. И. Ройз, А. Ю. Власов, И. Б. Петров

**ВИЗУАЛИЗАЦИЯ РЕЗУЛЬТАТОВ
ЧИСЛЕННЫХ ЭКСПЕРИМЕНТОВ
С ПОМОЩЬЮ СИСТЕМЫ VISUALIZATION TOOLKIT**

Даны общие требования к системам визуализации и их роль в научных расчетах. Описаны архитектура и возможности Visualization Toolkit (VTK), форматы входных данных и примеры визуализации численных экспериментов. Сделан обзор систем, использующих VTK.

122

General requirements of visualization systems and their role in scientific calculations are covered. The architecture and capabilities of Visualization Toolkit (VTK) are described. Input data formats and some calculation results are described. List of mature visualization systems used VTK is considered.

Ключевые слова: визуализация, численное моделирование, *Visualization Toolkit*.

Key words: visualization, numerical modeling, *Visualization Toolkit*.

Введение

Благодаря развитию вычислительной техники и методов математического моделирования сформировалась новая дисциплина, которая получила название «Научная визуализация» [1]. Ее основная задача — обеспечение возможности визуального представления и анализа больших объемов данных. Модуль визуализации становится важной частью современного вычислительного комплекса, так как без нее невозможно правильно судить о моделируемых явлениях.

Реализация модуля визуализации не уступает по сложности реализации численного метода. Создание собственной полноценной системы визуализации требует значительных усилий. Так, ресурсоемкость разработки ядра твердотельного моделирования Parasolid оценивается в 300 человеко-лет [2]. Поэтому при конструировании вычислительного комплекса имеет смысл проанализировать специфику существующих систем визуализации. В большинстве случаев система визуализации должна удовлетворять требованиям по выполнению следующих возможностей:

- 1) построения изоповерхностей;
- 2) построения произвольных сечений;
- 3) построения изолиний на сечениях;
- 4) конфигурирования прозрачности полигонов;
- 5) визуализации значительного количества полигонов в приемлемое время — это было нужно при визуализации частиц;
- 6) интерактивного взаимодействия с объектами сцены, например вращение.

Существует много систем визуализации. Некоторые из них устарели и не поддерживаются, другие активно развиваются и повсеместно используются. Ниже приведен список нескольких систем.



1. На кафедре информатики МФТИ разработана *glmfc*. Главное преимущество — высокая производительность за счет применения особенностей видеокарт NVIDIA. Но данная программа не удовлетворяла всем требованиям, в настоящее время проект не поддерживается.

2. *OpenDX* — проект с открытым исходным кодом, основанный на Visualization Data Explorer (IBM). Решение является системой быстрого построения конечных программ визуализации путем представления процесса визуализации в виде блок-схемы. Каждая программа — набор связанных блоков (модулей). Недостаток *OpenDX* — ограниченность набора модулей, необходимость писать свои модули и невозможность визуализировать ранее полученные результаты в других системах. В настоящее время проект практически не поддерживается.

3. *VTK* — открытое программное обеспечение, распространяемое по лицензии BSD и предназначенное для трехмерного компьютерного моделирования, обработки и визуализации численных данных. *VTK* также включает поддержку интерактивного взаимодействия и параллельной обработки данных [3]. Система состоит из библиотеки классов, реализованных на C++, и обертки для Python, Perl, Java, Tcl. *VTK* используется в коммерческих приложениях, исследованиях и разработке промышленных приложений и в качестве основы для таких комплексных систем визуализации, как ParaView, VisIt, VisTrails, Slicer, MayaVi, OsiriX.

Представим обзор возможностей применения системы визуализации The Visualization Toolkit (*VTK*). Приведем примеры кода для визуализации тестовых данных, результаты моделирования реальных задач.

VTK

Центральное место в архитектуре *Visualization Toolkit* занимает конвейер от источника данных к конечному изображению на экране. Обычно приложение состоит из нескольких конвейеров, соответствующих каждому элементу на экране. Это могут быть различные представления одних и тех же данных, различные их выборки или совершенно независимые наборы.

Конвейерная архитектура. Архитектура *VTK* состоит из модулей:

- 1) источник — источник исходных данных для обработки;
- 2) фильтры — предварительная обработка данных, например выборка, интерполяция, усреднение и так далее;
- 3) отображение — преобразование данных в объекты для визуализации: многогранники, поверхности, изолинии, линии тока;
- 4) настройки — настройки визуального представления: расположение камеры, цвета, прозрачность, освещение и другое;
- 5) визуализация — непосредственная отрисовка на экране или другом устройстве ввода. На этом этапе возможно интерактивное взаимодействие с пользователем для изменения настроек сцены.

Входные данные. *VTK* обрабатывает широкий набор типов входных данных, которые могут получаться в результате численного моделирования. Рассмотрим подробно эти типы и их структуру.

Регулярные вершины. Формат хранения данных в точках, расположенных в узлах регулярной решетки. Характеризуются количеством



точек по трем измерениям, координатами начальной точки и расстояниями между соседними точками. Пример входного файла дан ниже.

```
DATASET STRUCTURED_POINTS
DIMENSIONS nx ny nz
ORIGIN x y z
SPACING sx sy sz
```

Регулярная решетка. Формат хранения данных в узлах регулярной решетки. Характеризуется количеством узлов по трем измерениям и количеством вершин. Пример входного файла:

```
DATASET STRUCTURED_GRID
DIMENSIONS nx ny nz
POINTS n dataType
p0x p0y p0z
...
```

124

Линейная решетка. Формат хранения данных в узлах решетки с переменными расстояниями между соседними узлами. Характеризуется количеством узлов по трем измерениям и расстояниями между соседними узлами по каждому измерению. Пример входного файла:

```
DATASET RECTILINEAR_GRID
DIMENSIONS nx ny nz
X_COORDINATES nx dataType
x0 x1... x(nx-1)
Y_COORDINATES ny dataType
y0 y1... y(ny-1)
Z_COORDINATES nz dataType
z0 z1... z(nz-1)
```

Полигональные данные. Формат хранения связанных данных на произвольной поверхности. Характеризуется набором вершин и связями в отрезки и многогранники. Пример входного файла:

```
DATASET POLYDATA
POINTS n dataType
p0x p0y p0z
...
VERTICES n size
numPoints0, i0, j0, k0, ...
...
LINES n size
numPoints0, i0, j0, k0, ...
...
POLYGONS n size
numPoints0, i0, j0, k0, ...
...
TRIANGLE_STRIP n size
numPoints0, i0, j0, k0, ...
...
```

Неструктурированная решетка. Формат хранения данных в несвязанных ячейках. Характеризуется набором вершин и описанием ячеек. Пример входного файла:



```

DATASET UNSTRUCTURED_GRID
POINTS n dataType
p0x p0y p0z
...
CELLS n size
numPoints0, i, j, k, l, ...
...
CELL_TYPES n
type0
...
    
```

Примеры применения VTK

Ниже приведены примеры применения VTK для обработки результатов расчетов моделирования высокоскоростных соударений и взрывов методом гладких частиц (SPH) [4–5].

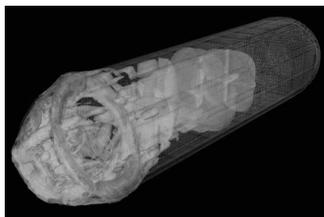


Рис. 1. Изоповерхности давления в перфорированной конструкции при взрыве

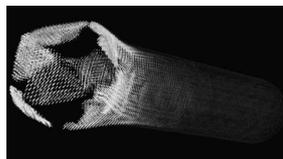


Рис. 2. Распределение векторов скоростей при взрыве в перфорированной конструкции

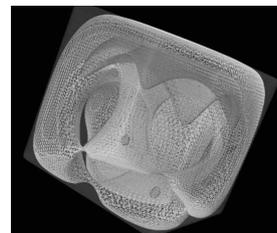


Рис. 3. Изоповерхности давления при взрыве в сплошной среде

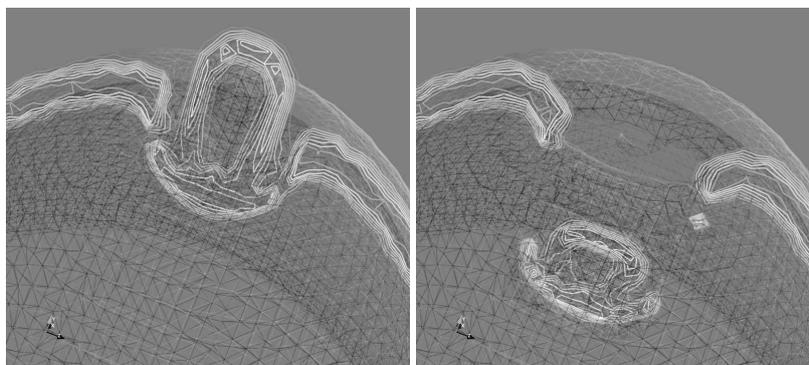


Рис. 4. Соударение полусферической оболочки (слева) и цилиндрического деформируемого ударника (справа). Изображены изолинии давления на поперечном срезе

Заключение

Приведем краткий список проектов, использующих VTK.
BioImageXD – открытый проект по обработке данных пространственной микроскопии.



VTK Designer — инструмент для обработки графического конвейера в виде блок-схемы. Предоставляет возможности графического представления настроек всех модулей и не требует программирования.

Virtual Creatures — проект для просмотра и манипулирования компьютеризированными моделями животных в высоком разрешении. Предназначен для образовательных целей.

Demoviewer — программа трехмерной визуализации ландшафтов.

NCSA Cave Visualization — система визуализации изоповерхностей и астрономических данных.

ParaView — открытая платформа визуализации / обработки данных.

Slicer3D — открытый пакет визуализации / обработки данных, используется в медицинских исследованиях для обработки снимков компьютерной томографии и реконструкции трехмерной поверхности.

Список литературы

1. Бондарев А. Е., Галактионов В. А. Научная визуализация в задачах вычислительной физики: концепции, методы, перспективы // Новые информационные технологии в автоматизированных системах: матер. XIII науч.-практ. семинара. М., 2010. С. 212–217.
2. Баранов Л. А. Место и роль геометрического ядра в современной САПР // САПР и графика. 2002. №6. С. 111–113.
3. *Visualization Toolkit. Applications.* URL: <http://www.vtk.org/VTK/resources/applications.html>
4. Блажевич Ю. В., Иванов В. Д., Петров И. Б., Петвиашвили И. А. Численное моделирование высокоскоростного соударения методом гладких частиц // Мат. моделирование. 1999. Т. 11, №1. С. 88–100.
5. Петров И. Б., Тормасов А. Г., Холодов А. С. Об использовании гибридизированных сеточно-характеристических схем для численного решения трехмерных задач динамики деформируемого твердого тела // ЖВМ и МФ. 1990. Т. 30, №8. С. 1237–1244.

Об авторах

Сергей Иванович Ройз — асп., Московский физико-технический институт.

Андрей Юрьевич Власов — асп., Московский физико-технический институт.

Игорь Борисович Петров — д-р физ.-мат. наук, проф., Московский физико-технический институт, e-mail: petrov@mipt.ru.

Authors

Sergey Roiz — PhD student, Moscow Institute of Physics and Technology.

Andrey Vlasov — PhD student, Moscow Institute of Physics and Technology.

Professor Igor Petrov — Moscow Institute of Physics and Technology, e-mail: petrov@mipt.ru.