РАЗДЕЛ II. ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Авдеев М.В., Бутенко М.А.

Визуализация многомерных данных в графическом пакете ParaView в пакетном режиме

ФГАОУ ВО «Волгоградский государственный университет»
(Россия, Волгоград)

doi:10.18411/sciencepublic-12-07-2019-02

idsp: sciencepublic-12-07-2019-02

Аннотация

В данной работе приведено описание разработанного конвертера для пакетной обработки данных численного моделирования, полученных бессеточным методом SPH. Разработан макрос для автоматизации визуализации в программе ParaView в пакетном режиме. Показаны результаты применения разработанного программного обеспечения для визуализации данных численного гидродинамического моделирования.

Ключевые слова: визуализация, численное моделирование, пакетная обработка, вычислительная гидродинамика, обработка данные.

Введение

Для огромного количества научно-технических задач, исследующих эволюцию какой-либо системы, невозможно получить точное решение. Для получения приближенного решения используется численное моделирование, результатами которого являются большие массивы данных. Для анализа данных численных экспериментов необходимо применять визуализацию, что делает ее весьма актуальной задачей. В данной работе представлено программное обеспечение для визуализации научных данных, полученных при моделировании газового галактического диска, вращающегося в гравитационном потенциале темного гало. Физическая постановка задачи и используемый для ее решения численный метод подробно описаны в работе [1]. Для моделирования использовался метод сглаженных частиц (SPH) [2], распараллеленный для работы на графических процессорах с помощью технологии CUDA [1].

Результатом численного гидродинамического моделирования является набор однотипных файлов, содержащих состояние системы в некоторый момент времени. В таких условиях актуальна пакетная обработка данных. Она позволяет автоматизировать повторяющиеся действия пользователя, что приводит к существенной экономии времени, особенно, когда стоит задача обработки терабайтов данных. Программа ParaView предоставляет возможности пакетной визуализации.

ParaView – это один из самых популярных пакетов для визуализации данных с открытым исходным кодом на сегодняшний день. Главным разработчиком является компания Kitware, которая занимается разработкой ПО для таких областей науки как компьютерное зрение, анализ данных, высокопроизводительные вычисления и визуализация и медицинская визуализация, где и находит применение ParaView [3].

Разработка пакетного конвертера

Результатом численных экспериментов, как было сказано ранее, является набор файлов состояния системы в определенный временной точке. Как правило, такие файлы не имеют общепринятого формата, их структура определяется разработчиком. В нашем случае данные записанные в бинарные файлы имеют следующую структуру:

1. Число SPH частиц – int;
2. Время эксперимента – double;

3. Перечисление параметров для каждой частицы, тип данных – double:

- a. Значение координаты x;
- b. Значение координаты y;
- c. Значение координаты z;
- d. Значение параметра плотности ρ ;
- e. Параметр вектора скорости u;
- f. Параметр вектора скорости v;
- g. Параметр вектора скорости w;
- h. Значение параметра энергии e.

Такая структура данных не пригодна для обработки в ParaView непосредственно. Чтобы визуализировать данные в ParaView, необходимо преобразовать их в понятный для программы формат. Для решения этой задачи был написан пакетный конвертер, с помощью которого бинарные файлы с данными преобразовывались в формат Unstructured Grid, который является встроенным форматом библиотеки VTK.

Пакетная обработка подразумевает что данные необходимо сгруппировать в пакет. ParaView поддерживает формат данных ParaView Data (PVD), структура которого представлена в листинге 1. Данный формат специально предназначен для хранения информации о пакетах файлов [4]. Формат базируется на XML и понятен для человека. Каждый файл пакета описывается элементом DataSet внутри Collection. С помощью атрибутов file, part, group и timestep указываются соответственно: местоположение файла пакета, номер части данных, идентификатор данных и временная метка файла.

Листинг 1 – Структура ParaView Data

```
<?xml version="1.0"?>
<VTKFile type="Collection" version="0.1"
        byte_order="LittleEndian"
        compressor="vtkZLibDataCompressor">
  <Collection>
    <DataSet timestep="0" group="" part="0"
            file="nir/g0000.vtu"/>
    <DataSet timestep="1" group="" part="0"
            file="nir/g0001.vtu"/>
  </Collection>
</VTKFile>
```

Для реализации конвертера использовался язык программирования C++ 17, поддерживающий библиотеку Path, с использованием сторонних библиотек: Visualization Toolkit и TinyXML2.

Преобразование бинарных данных выполнялось с помощью комбинации функции побитового чтения `read(char* s, streamsize n)` и оператора приведения типов: `reinterpret_cast<type>(object)` для преобразования байт в числа [5]. Код преобразования представлен в листинге 2.

Листинг 1 - Чтение и преобразование бинарных данных

```
for (int i = 0; i < N; i++) { // Цикл по числу точек N
    double * data = new double[8]; // Указатель для хранения
    данных

    input.read(reinterpret_cast<char*>(data),
sizeof(double)*8);
    // Заполнение массивов данных в соответствии с
    структурой бинарного файла
```

```

xP[i] = data[0];
yP[i] = data[1];
zP[i] = data[2];
Rhot[i] = data[3];
uP[i] = data[4];
vP[i] = data[5];
wP[i] = data[6];
eP[i] = data[7];

delete[] data; // Освобождение памяти
}

```

Библиотека VTK применялась для формирования и записи выходных файлов, содержащих данные в формате нерегулярной сетки. Применялись следующие классы [6]:

- `vtkUnstructuredGrid` – контейнер для данных нерегулярной решетки;
- `vtkFloatArray` – контейнер для хранения скалярных данных типа «число с плавающей точкой»;
- `vtkPoints` – контейнер для хранения координат точек в пространстве;
- `vtkCellArray` – контейнер для хранения примитивов;
- `vtkVertex` – примитив – вершина;
- `vtkUnstructuredGridWriter` – реализует запись файла в формате нерегулярной решетки.

Формирование ParaView Data – файла реализовано с помощью инструментов библиотеки TinyXML2. Эта библиотека позволяет очень просто создавать новые элементы и добавлять их в XML документ, а также реализует сохранение полученного файла в формате XML.

Процесс обработки файлов начинается с чтения директории ввода и заполнения списка путей к конвертируемым файлам. Следующий шаг – последовательная обработка файлов в функции `Convert()`, реализующей преобразование данных. После обработки всех файлов выполняется функция `WritePVD()`, реализующая запись ParaView Data – файла. Алгоритм работы конвертера показан на рисунке 2.

Рисунок 1 - Блок-схема функции `Convert()`

Результатом работы конвертера является набор файлов данных в формате нерегулярной сетки VTU и PVD-файл, содержащий информацию о файлах пакета для импорта в ParaView.

Разработка макроса

Следующим этапом работы является разработка макроса для автоматизации процесса визуализации в ParaView и вывода результатов в графический формат. ParaView предоставляет возможность работы с данными через встроенный интерпретатор Python Shell, для отображения окна Python Shell необходимо выбрать соответствующий пункт в меню View, расположенном на верхней панели интерфейса. Для работы с данными пользователь может писать команды на python прямо в консоль, или запустить готовый python-сценарий, или сохранить сценарий в качестве макроса и использовать его из меню Macros [7].

В ходе выполнения работы был написан python-сценарий, выполняющий следующие задачи по работе с данными:

1. Применение фильтра SPH Volume Interpolator.
2. Составление композиции изображения.
3. Экспортирование изображений.

Блок-схема полученного сценария приведена на рисунке 2.

Рисунок 2 - Блок-схема автоматизирующего макроса

Результаты работы макроса

Результатом работы полученного макроса является набор изображений, где каждое изображение представляет визуализацию соответствующего файла из пакета.

На рисунке 3 приведен результат визуализации одного из файлов из пакета данных, соответствующего некоторому моменту времени t . Цветовая шкала отражает значение плотности ρ интерполированных SPH-частиц.

Полученное в результате визуализации изображение содержит:

1. трехмерный вид газового галактического диска под некоторым углом в текущий момент времени (слева);
2. срез диска вдоль оси Ox (снизу);
3. срез диска в плоскости $z = 0$ (справа);
4. цветовую шкалу, связанную с параметром плотности ρ .

Рисунок 3 - Результат работы макроса

Заключение

В ходе выполнения работы был проведен анализ форматов данных, доступных для работы в программном пакете ParaView. На основании результатов анализа выбран формат данных с нерегулярной сеткой VTK Unstructured Grid для преобразования данных, полученных в процессе численного моделирования. Разработан пакетный конвертер, с помощью которого была произведена конвертация данных, полученных при моделировании галактического диска методом сглаженных частиц. Для автоматизации визуализации данных был разработан макрос, выполняющий композицию и экспорт изображений в файлы графического формата. Изменить композицию изображения на выходе, например, угол обзора или плоскости для среза, можно с помощью незначительных изменений в python-скрипте. Приведенный в работе подход к визуализации с применением разработанного программного обеспечения для конвертации данных позволяет существенно ускорить процесс визуализации для большого объема данных вычислений.

Исследование выполнено при финансовой поддержке РФФИ и Администрации Волгоградской области в рамках научного проекта №18-42-343005 (p_мол_a).

1. Butenko M., Khoperskov A., Khrapov S. Modeling stellar-gaseous disks: rows in spiral patterns of galaxies // Journal of Physics: Conference Series, 2019. – V. 1203. – pp. 012062.
2. Monaghan J.J. Smoothed particle hydrodynamics // Annual review of astronomy and astrophysics, 1992. – V. 30. – pp. 543–574.
3. The Official ParaView's page [Электронный ресурс] // The Official ParaView's page: <https://paraview.org>.
4. ParaView/Data formats – KitwarePublic [Электронный ресурс] // Kitware Public Wiki. 2016. URL: https://www.paraview.org/Wiki/ParaView/Data_formats.
5. Lischner R. C++ In a Nutshell. – 1st ed. – O'Reilly Media. – 2003. – 704 pp.
6. Kitware Inc. File Formats // VTK - The Visualization Toolkit. 2010. URL: <https://vtk.org/wp-content/uploads/2015/04/file-formats.pdf>.
7. Quammen C. Scientific Data Analysis and Visualization with Python, VTK, and ParaView // Proceedings of the 14th Python in Science Conference (SciPy 2015). – 2015. – pp. 31–38.