

ScientificView - система параллельной постобработки результатов, полученных при численном моделировании физических процессов

А.Л. Потехин, В.И. Тарасов, С.А. Фирсов, И.В. Логинов, В.А. Никитин, М.Г. Кузнецов,
Н.В. Попова, А.К. Деманова, Ю.В. Козачек.

Институт теоретической и математической физики

РФЯЦ-ВНИИЭФ, Саров, Россия

{Potekhin, VTarasov, Firsov, I.V.Loginov, Yu.V.Kozachek}@vniief.ru

Аннотация

В работе описывается параллельная система постобработки ScientificView, реализованная в Институте теоретической и математической физики РФЯЦ-ВНИИЭФ. Данная система предназначена для фильтрации, отображения и табличного анализа результатов моделирования физических процессов на разностных сетках регулярного типа.

Описаны схема построения и архитектура системы; процесс обработки запросов; рассмотрены некоторые процедуры обработки данных, реализованные на основе предложенной схемы. Приведены графики изменения времени работы некоторых алгоритмов обработки, в зависимости от числа используемых процессоров; рассмотрены перспективы развития системы постобработки в целом.

Ключевые слова: параллельные системы постобработки, постобработка результатов научных вычислений, графическое отображение данных.

1. ВВЕДЕНИЕ

Мощные многопроцессорные ЭВМ позволяют более детально описывать физические процессы за счет применения более точных моделей, описываемых уравнениями математической физики, при решении которых разностными методами требуются существенные вычислительные ресурсы и большое количество счетных точек. Оценка качества проведенных расчетов требует развитых средств анализа и постобработки, в частности, систем визуализации способных быстро обрабатывать большой объем данных. Добиться этого с помощью однопроцессорных программ затруднительно по причине ограниченности ресурсов ПЭВМ.

В ИТМФ накоплен опыт использования параллельных систем визуализации, в частности проводится адаптация системы ParaView[1] для обработки данных, рассчитанных математическими методиками отделения. Работы по адаптации показали, что, несмотря на имеющийся большой инструментарий ParaView, реализация в нем алгоритмов, учитывающих особенности методик связана с большими трудозатратами (например, поддержка адаптивно встраиваемых сеток), либо просто невозможна (например, поддержка возможности обработки нерегулярных ячеек произвольной формы без их дробления на примитивы ParaView). Внедрение в ParaView для полноценной обработки расчетных данных неграфических средств (табличного просмотра, расчета интегральных характеристик и т.д.), также связано с большими трудозатратами. Поэтому было решено создать параллельную систему постобработки

ScientificView на основе собственных программных продуктов, а именно системы визуализации программы расчета начальных данных 3D-РНД[2], программы табличной обработки данных EFR-Viewer[3], библиотеки ЕФР[4] для доступа к данным на диске.

2. ВЫБОР СХЕМЫ ПОСТРОЕНИЯ СИСТЕМЫ

Наиболее широко используются три схемы построения параллельных систем постобработки. В этих схемах в параллельном режиме работают модули чтения и процедуры фильтрации данных. Отличие заключается в работе процедур подготовки изображения: они могут быть распараллелены (схема с максимальным распараллеливанием), работать в скалярном варианте на ПЭВМ (схема клиент-сервер) или на одном процессе, принадлежащего параллельному полю (схема удаленного клиента).

При реализации системы на основе программ, разработанных в ИТМФ, было решено использовать схему клиент-сервер с клиентской частью, работающей под Windows, и серверной частью, запущенной под Linux в параллельном режиме на многопроцессорной ЭВМ с распределенной памятью (рисунок 1).

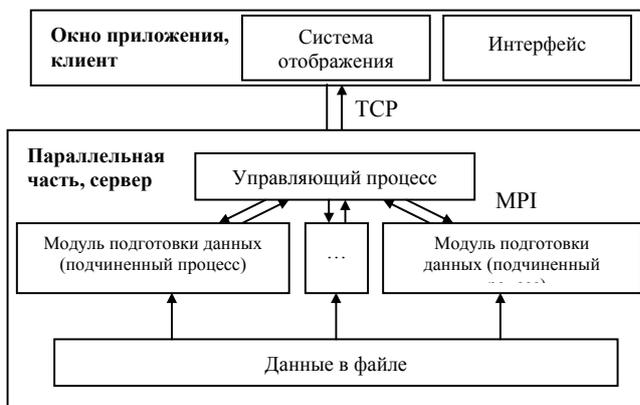


Рис 1: Схема построения ScientificView.

Эта схема оказалась проще в реализации, поскольку для создания клиентской части (интерфейса пользователя, системы отображения) достаточно было выделить соответствующие процедуры из имеющихся скалярных программ. Серверная часть была создана за счет выделения в отдельное кроссплатформенное приложение и распараллеливания имеющихся в скалярных программах процедур чтения и обработки данных.

Не смотря на простоту, реализация системы по данной схеме позволила увеличить объемы и скорость обработки

расчетных данных по сравнению со скалярными версиями программ.

Связь клиента и серверной части осуществляется средствами TCP/IP. Для этого среди процессов серверной части выделяется специальный *управляющий* процесс. Управляющий процесс осуществляет прием запросов от клиента, распределяет нагрузку по данному запросу на остальные (*подчиненные*) процессы сервера, собирает от них результаты обработки.

Управляющий процесс не проводит чтение и обработку сеточных данных, поскольку это приводит к замедлению работы всей серверной части. В будущем, при реализации процедур формирования изображения в параллельном режиме, каждый подчиненный процесс будет формировать часть изображения по имеющимся у него данным, а итоговое изображение предполагается формировать на управляющем процессе.

Поскольку связь клиента на ПЭВМ осуществляется лишь с одним процессом серверной части, многопроцессорная ЭВМ может быть неоднородной и не до всех ее узлов обязательно существование прямого соединения с ПЭВМ пользователя.

3. АРХИТЕКТУРА СИСТЕМЫ

Необходимо было разработать архитектуру системы, которая позволяла бы с минимальными трудозатратами провести первичную реализацию системы и упростить ее развитие. Поскольку речь идет о графической системе, под развитием в первую очередь понимается создание новых возможностей по обработке данных (различных алгоритмов фильтрации и анализа данных), а также возможностей по обработке данных новых типов (хранящихся в новых форматах файлов или данных, полученных в результате моделирования новых, ранее не рассматриваемых физических процессов). Разработанная архитектура, должна также обеспечить кроссплатформенность программной реализации.

При проектировании были выделены несколько информационных уровней, каждый из которых представляется некоторой программной реализацией (рисунок 2).

1. Геометрические примитивы: точки, вектора, отрезки, многоугольники. Каждый примитив хранит часть информации, принадлежащей либо исходным сеточным данным (математическим областям), либо результатам обработки (в первую очередь результатам фильтрации). Примитивы, как правило, самодостаточны и не нуждаются в информации о своем реальном окружении (в частности одна точка ничего не «знает» о соседней).
2. Контейнер примитивов. Контейнер предназначен для хранения однотипных примитивов (например, многоугольников), а также для их обработки: добавления, удаления, сортировки, поиска дубликатов и т.д. В каждом контейнере хранится информация о взаимосвязях находящихся в нем примитивов.
3. Элементы сцены. Под элементами сцены понимаются объекты, хранящие результаты работы фильтрующих алгоритмов или расчетные данные. Используют контейнеры для хранения своих геометрических примитивов.

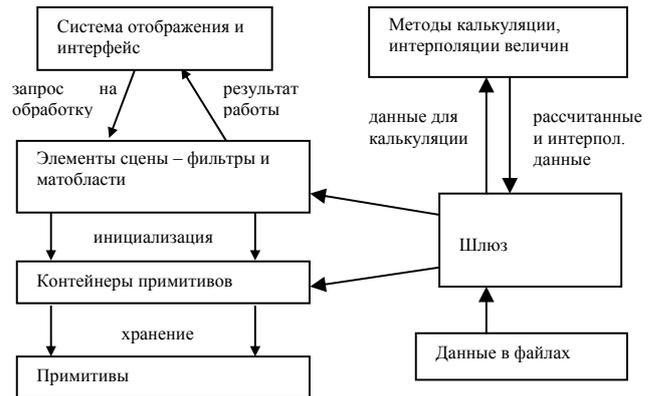


Рис 2: Условная схема архитектуры системы.

Доступ к данным в файлах или оперативной памяти осуществляется через шлюзовую систему. Она же обеспечивает доступ к данным, полученным интерполяцией или калькуляцией по специальным формулам.

Взаимодействие пользователя с ScientificView организовано через интерфейсную часть и систему отображения, осуществляющую вывод графической информации.

Потоки данных, связывающие модули и информационные слои обрабатываются ядром системы визуализации. Это позволяет при разработке новых алгоритмов воспользоваться ранее реализованными базовыми конструкциями, например, процедурами инициализации объектов, получения данных из файла и отображения результатов обработки.

4. ПРОЦЕДУРЫ ОБРАБОТКИ ЗАПРОСОВ

Взаимодействие клиентской и серверной части приложения осуществляется на основе запросов, которые группируются в пакеты четырех типов. Первый тип пакета – «информационный». Он указывает принимающей стороне (как правило, серверу) какую операцию необходимо выполнить, а также размер данных, которые будут переданы в качестве параметров операции следующим типом пакета – «пакетом с данными». Содержимое «пакета с данными» зависит от проводимой операции, правила формирования пакета согласованы между отправляющей и принимающей стороной. Если сторона, принимающая запрос – сервер, а результат обработки будет передан несколькими пакетами, сначала сервер посылает клиенту «процедурный» пакет. Пакет содержит информацию, необходимую для организации приема обработанных данных. Сами данные также передаются «пакетом с данными». На «информационный» и «процедурный» типы пакетов принимающая сторона отвечает «пакетом отклика», что является подтверждением факта приема.

Пакеты обрабатываются процедурами собственной разработки. Простейший сценарий обработки запроса приведен на рисунке 3. Поскольку взаимодействие с клиентом осуществляет только управляющий процесс сервера, он должен разослать полученную от клиента информацию всем остальным процессам сервера, а также собрать результаты обработки данных для передачи клиенту. Это осуществляется за счет скрытого взаимодействия процессов серверной части при обработке запроса. Например, функция приема информационного пакета (рисунок 3, пункт

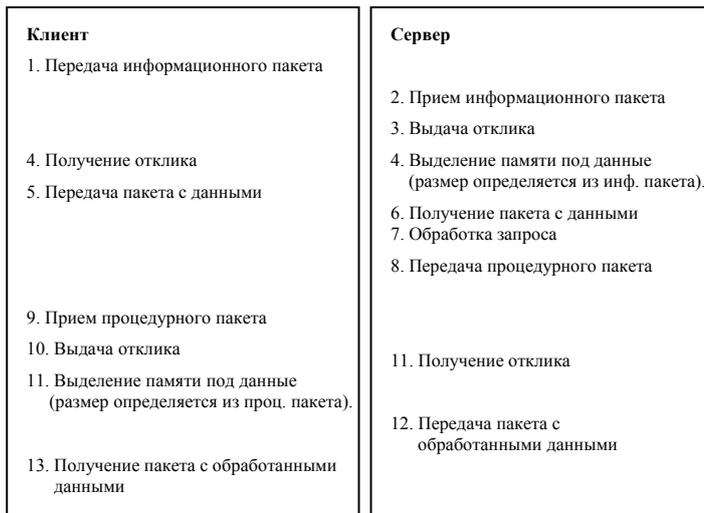


Рис 3: Простейший сценарий обработки запроса.

2, справа) на управляющем процессе проводит передачу содержимого пакета на подчиненные процессы сервера с помощью MPI. Функция передачи процедурного пакета (рисунок 3, пункт 8, справа), вызванная на подчиненном процессе передаст данные с помощью MPI управляющему, а тот, в свою очередь – клиенту. Реализация скрытого взаимодействия позволила использовать одинаковые алгоритмы обработки на управляющем и подчиненных процессах. На основе системы запросов проведено распараллеливание основных алгоритмов обработки данных, реализованных в скалярной версии системы визуализации программы 3D-РНД, которые обсуждаются далее.

5. РЕЖИМЫ ОТОБРАЖЕНИЯ

Интерпретация значений величин цветом при отображении исходных сеточных данных и результатов их обработки возможна несколькими способами: заливка полигонов, «цветная» сетка или «цветные» узлы сетки, отображение внешних границ сетки (рисунок 4).

В качестве величины для интерпретации выступает любая величина, определенная в ячейках или узлах сетки (рисунок 5).

При отображении объектов возможно применение эффекта прозрачности, упрощающего анализ сеточных данных, находящихся «внутри» объекта (рисунок 6).

6. НЕКОТОРЫЕ АЛГОРИТМЫ ГРАФИЧЕСКОЙ ОБРАБОТКИ ДАННЫХ

В параллельном режиме были реализованы «классические» алгоритмы построения векторных полей, сечений плоскостью и изоповерхностей (рисунок 7) [1, 5], а также алгоритмы, предназначенные для обработки регулярных сеточных данных: выделение слоя граней с постоянным значением одного из индексов (рисунок 8 слева); выделение подобласти, заданной интервалами изменения индексов (рисунок 8 справа).

Для анализа топологии сетки реализован алгоритм, наиболее часто применяемый при обработке нерегулярных данных (фильтр «Проба»). Он позволяет просмотреть окрестность

выбранного узла или ячейки - номера граней и узлов примыкающих ячеек и т.д. (рисунок 9).

Для выделения ячеек или узлов, величины в которых находятся в заданном интервале, реализован фильтр «Интервал». Он, в частности, позволяет выделить ячейки с определенным веществом (рисунок 10).

В последних версиях системы появилась возможность применения одного алгоритма фильтрации к результатам работы другого. Это позволяет, например, выделить часть счетного пространства, построить там сечение и на нем построить векторное поле (рисунок 11).

7. АЛГОРИТМЫ ТАБЛИЧНОЙ ОБРАБОТКИ ДАННЫХ

Большинство алгоритмов табличной обработки изначально было реализовано в программе EFR-Viewer. Посредством гипертекстовых интерфейсов (рисунок 12) имеется возможность:

- просмотра общих характеристик задачи, числа, типа и параметров математических областей;
- просмотра набора массивов, списков, структур, определенных для каждой области;
- расчета интегральных характеристик, сравнение балансов сеточных величин из двух файлов-разрезов;
- внесения изменений в любую составляющую файла-разреза

Большинство этих алгоритмов внедрено в ScientificView.

8. РЕЗУЛЬТАТЫ ТЕСТИРОВАНИЯ

При численном моделировании физических процессов по программам ИТМФ РФЯЦ-ВНИИЭФ в трехмерном случае, как правило, используются разностные сетки с числом ячеек около 100 млн. В дальнейшем число ячеек должно достигнуть 500-1000 млн. Поэтому, для проверки работоспособности ScientificView при обработке таких объемов данных был создан набор тестовых задач с числом ячеек 500 млн.

Тестирование проводилось на 8-48 процессорах серверной части. При использовании 48 процессоров время открытия файлов и построения основных фильтров составило порядка 40 секунд. Система постобработки обеспечила при этом отображение 3-5 кадров в секунду, потребовав менее 1Gb оперативной памяти на ПЭВМ и менее 400 Mb на каждом процессоре сервера. ScientificView, как показало тестирование, обеспечивает приемлемые времена обработки. Наиболее слабым местом системы является ее клиентская часть. Это, как отмечалось выше, связано с ограниченными ресурсами ПЭВМ, необходимыми для работы системы отображения. Решение этой задачи видится в двух направлениях – увеличение мощности и памяти ПЭВМ и распараллеливание процедур отображения.

На рисунке 13 приведены графики изменения времени чтения данных из файла и построения фильтров «Сечение плоскостью» и «Векторное поле» в зависимости от числа процессоров в серверной части.

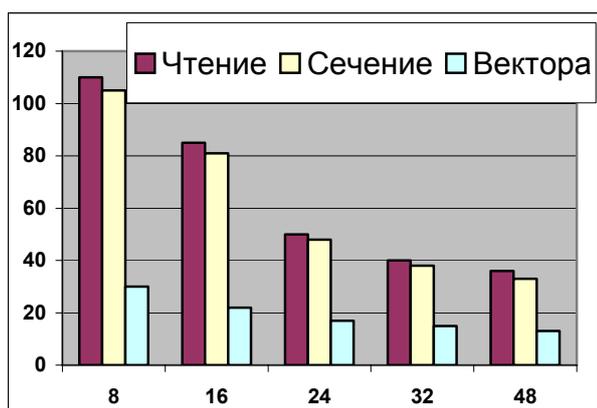


Рис 13: Графики изменения времени работы (в секундах) алгоритмов в зависимости от числа используемых процессоров в серверной части.

9. ПЕРСПЕКТИВЫ РАЗВИТИЯ

В ближайшее время планируется провести распараллеливание алгоритмов обработки неструктурированных данных и ряда дополнительных фильтров для обработки структурированных: выделение границы раздела веществ, построение линии профиля. Необходимо также провести дополнительную оптимизацию процедур чтения и обработки данных с целью их ускорения. В долгосрочной перспективе планируется реализовать возможности подготовки изображения в параллельном режиме (параллельный рендеринг), возможности альтернативного представления данных (воксельная и стерео графика), управления камерой посредством специальных манипуляторов («виртуальных перчаток»).

10. БИБЛИОГРАФИЯ

- [1] Henderson A., Ahren J., etc. *The ParaView guide. Published by Kitware, Inc. 2004*
- [2] Потехин А.Л. *Интерактивная система визуализации в программе трехмерного расчета начальных данных 3D-РНД. Труды всероссийской конференции "Прикладная геометрия, построение расчетных сеток и высокопроизводительные вычисления". Москва, 2004.*
- [3] Попова Н.В., Деманова А.К. *Программа EFR VIEWER как средство быстрого анализа результатов численного моделирования физических процессов; Материалы конференции «Молодежь в науке». Саратов, 2007.*
- [4] Волгин А.В., Красов А.В., Кузнецов М.Ю., Тарасов В.И. *Библиотека EFR для универсального представления расчетных данных. Труды РФЯЦ-ВНИИЭФ. Выпуск 11. Саратов, 2007.*
- [5] Зибаров А.В. *Cutting-edge CFD solutions for science and industry [Электронный ресурс]. Режим доступа: [http://www.cfd.ru 10.02.2006].*

Об авторах

Потехин Антон Львович, начальник сектора лаборатории общего сервиса Института теоретической и математической физики, Российского федерального ядерного центра, Всероссийского научно-исследовательского института экспериментальной физики. Почта: potekhin@vniief.ru

Тарасов Виктор Иванович, кандидат физ.-мат. наук, начальник лаборатории общего сервиса ИТМФ, РФЯЦ-ВНИИЭФ. Почта: vtarasov@md08.vniief.ru

Фирсов Сергей Александрович, кандидат физ.-мат. наук, сотрудник технического отдела ИТМФ, РФЯЦ-ВНИИЭФ. Почта: firsov@vniief.ru

Никитин Вячеслав Александрович, Кузнецов Михаил Геннадьевич, Логинов Иван Владимирович, Попова Наталья Васильевна, Деманова Анна Константиновна, Козачек Юлия Владимировна, сотрудники лаборатории общего сервиса ИТМФ, РФЯЦ-ВНИИЭФ.

Почта: [{ I.V.Loginov, Yu.V.Kozachek}@vniief.ru](mailto:{I.V.Loginov, Yu.V.Kozachek}@vniief.ru)

ScientificView - parallel system for postprocessing results, got under numerical modeling of the physical processes

Abstract

This paper is told about parallel postprocessing system, developed in Institute theoretical and mathematical physics RFYAC-VNIIEF. The system is intended for visual, filtering and tabular analysis of result modeling of the physical processes on regular and non-regular nets. Building scheme, system architecture, request processing is shortly described in paper.

The paper is also told about results got on cluster machine during of system testing.

Keywords: parallel visualization systems, processing of science calculations, data rendering.

About the authors

Anton L. Potekhin, leader of visualisation team in mathematical department ITMF, RFYAC-VNIIEF. E-mail: potekhin@vniief.ru

Victor I. Tarasov, PhD at mathematical department ITMF, leader of general-purpose systems group, RFYAC-VNIIEF. E-mail: firsov@vniief.ru

Vyacheslav A. Nikitin, Michael G. Kuznetsov, Ivan V. Loginov, Nataliya V. Popova, Anna K. Demanova, Julia V. Kozachek, mathematicians of visualisation team in mathematical department ITMF, RFYAC-VNIIEF. E-mail: [{ I.V.Loginov, Yu.V.Kozachek}@vniief.ru](mailto:{I.V.Loginov, Yu.V.Kozachek}@vniief.ru)

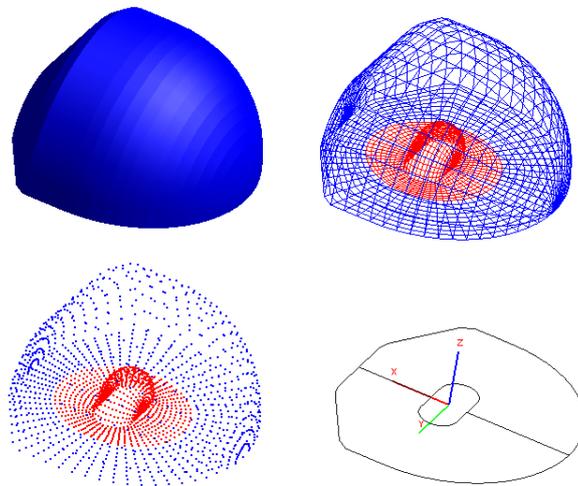


Рисунок 4: Способы интерпретация величин цветом: заливка полигонов, «цветная» сетка или «цветные» узлы сетки, отображение границ.

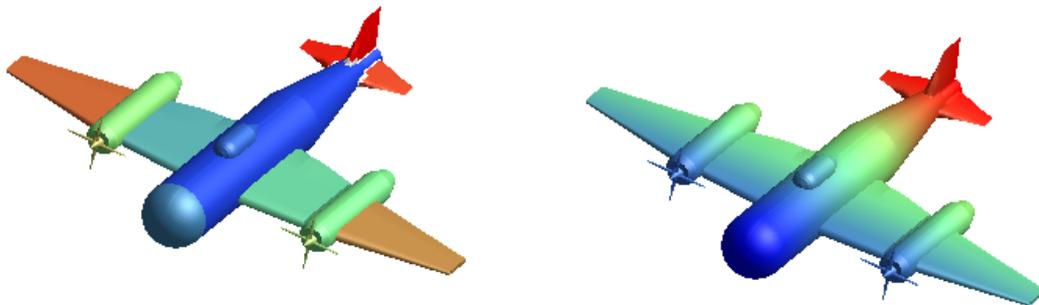


Рисунок 5: Заливка многоугольников по величинам из ячеек (вещество, слева) и узлов сетки (координата X, справа)

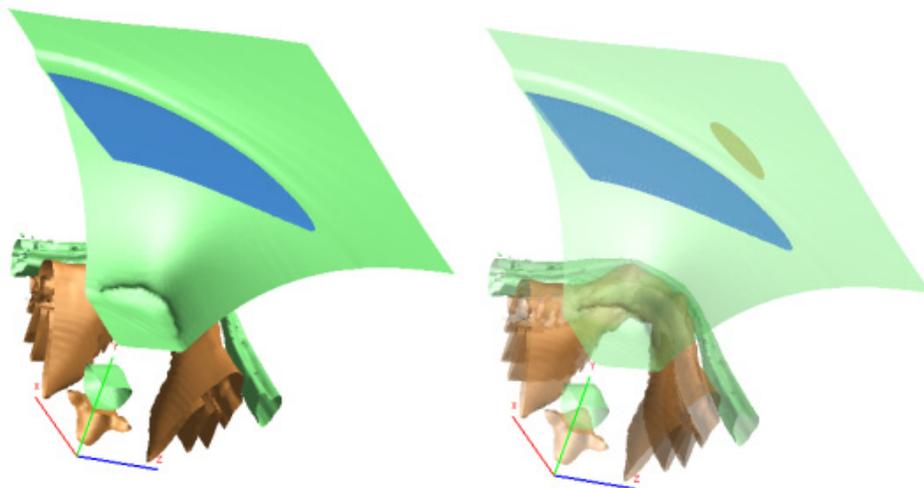


Рисунок 6: Отображение изоповерхности обычное и с применением прозрачности

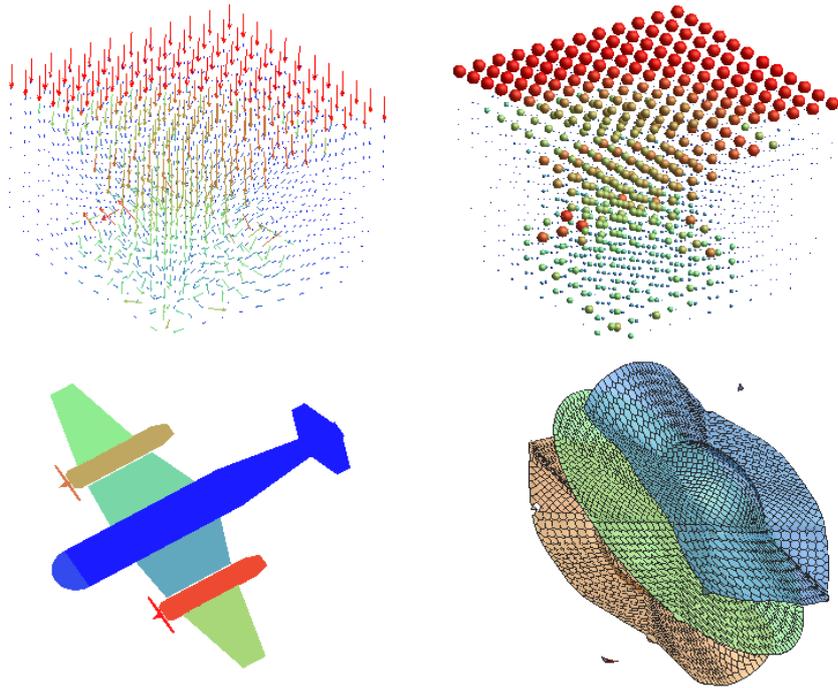


Рисунок 7: Результат работы фильтров «Векторное поле», сверху (отображение в двух режимах), «Сечение» и «Изоповерхность», внизу.

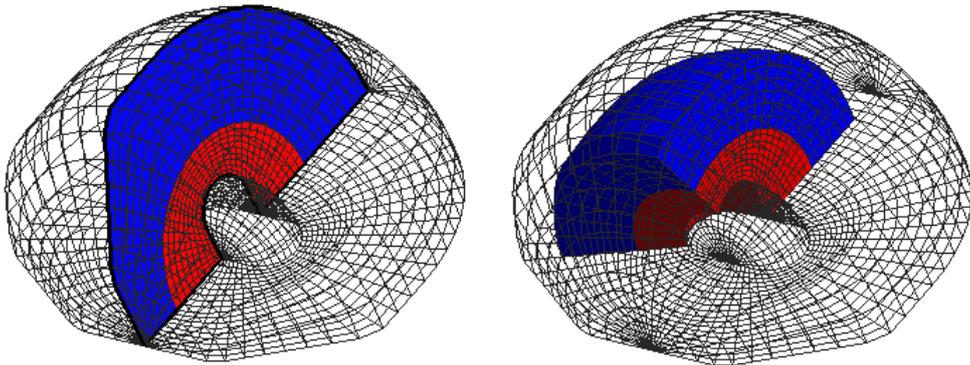


Рисунок 8: Построение слоя грани и регулярной подобласти

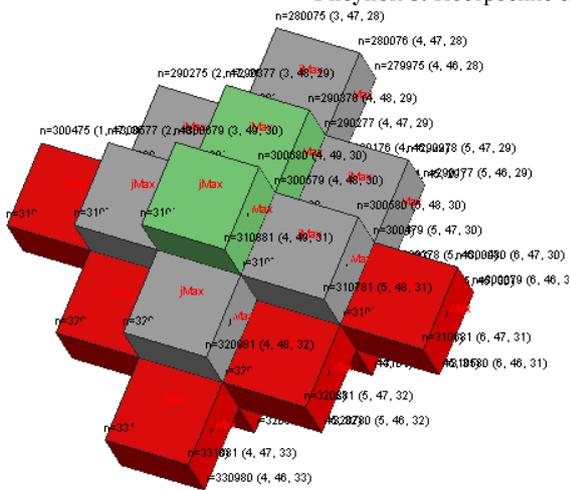


Рисунок 9: Анализ топологии с помощью фильтра «Проба»

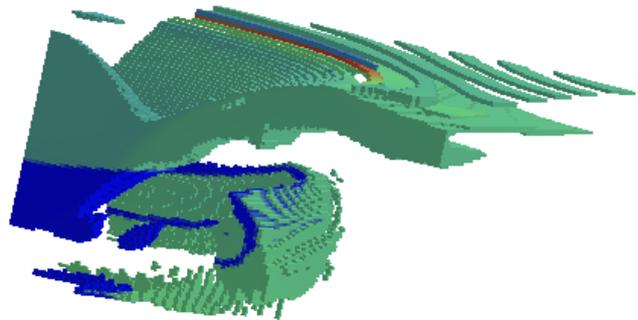


Рисунок 10: Результат работы фильтра «Интервал»

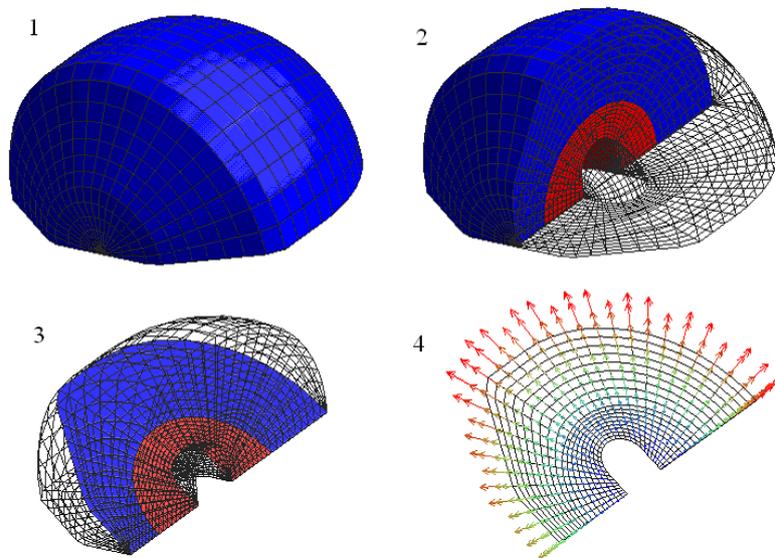


Рисунок 11: Применение иерархической фильтрации

EFR Viewer 3.1.2 - D:_efr\2d\Chukhmanov_NV\118_calculate_cell.efr

Файл Вид Правка Визуализация Поиск Пересчет Сервис Справка

Параметры разреза | Просмотр и редактирование данных | Сохранение фрагмента данных в отдельный файл EFR | Поиск

0

- 2D
 - 3000 X
 - 3001 Y
 - 3011
 - R² Радиус
- 1
 - 3000 X
 - 3001 Y
 - 3011
 - R² Радиус
- 1
 - 1D
 - 1109
 - 1110
 - 1111 MF
 - 1112 KF
 - 1113 LF
 - 1114 NFOF
 - 1121 MO
 - 1129 XO
 - 1130 XOP
 - 1131 YO
 - 1132 YOP
 - 1133 SL
 - 1134 SLP
 - 1135 QU
 - 1136 QUP
 - 1137 SQ
 - 1138 SQP
 - 1139 CQ
 - 1140 CQP

Параметры просмотра

Постоянный индекс: NK

Значение индекса:

Кратности: 1

Оси: I J K

Развернуть оси

Параметры редактирования

Укажите новое значение:

Имя текстового файла:

R² Радиус

	I 1	I 2	I 3
J 1	5.06250000000000e+0;	5.06250000000000e+0;	5.116053460112345e+0;
J 2	5.062397716288951e+0;	5.062397716288951e+0;	5.111750348808725e+0;
J 3	5.062250430651529e+0;	5.062250430651529e+0;	5.107415219587236e+0;
J 4	5.062333514946807e+0;	5.062333514946807e+0;	5.107468187066166e+0;
J 5	5.062326591434611e+0;	5.062326591434611e+0;	5.107470834491667e+0;
J 6	5.062269313347808e+0;	5.062269313347808e+0;	5.107432880642820e+0;
J 7	5.062300469137919e+0;	5.062300469137919e+0;	5.107446705959027e+0;
J 8	5.062241618758996e+0;	5.062241618758996e+0;	5.107409603430050e+0;
J 9	5.062310854932617e+0;	5.062310854932617e+0;	5.107459309291684e+0;
J 10	5.062290084422280e+0;	5.062290084422280e+0;	5.107441723558991e+0;
J 11	5.062298266998117e+0;	5.062298266998117e+0;	5.107444755940202e+0;
J 12	5.062315575757740e+0;	5.062315575757740e+0;	5.107463309691071e+0;
J 13	5.062242877599257e+0;	5.062242877599257e+0;	5.107414815225284e+0;
J 14	5.062298266094165e+0;	5.062298266094165e+0;	5.107445107891991e+0;
J 15	5.062263648529233e+0;	5.062263648529233e+0;	5.107424625995561e+0;
J 16	5.062337291832472e+0;	5.062337291832472e+0;	5.107477294541650e+0;

Фрагм

I 1

J 1

K 1

Средний

Рисунок 12: Диалог просмотра значения массивов выбранной области