

Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение высшего
профессионального образования

**«Уральский федеральный университет
имени первого Президента России Б.Н. Ельцина»**

Институт математики и компьютерных наук

Кафедра высокопроизводительных компьютерных технологий

**СПЕЦИАЛИЗИРОВАННАЯ СИСТЕМА
ВИЗУАЛИЗАЦИИ СЕТОЧНЫХ ДАННЫХ**

«Допущена к защите»

Зав.кафедрой
А.В. Созыкин

Квалификационная работа
на соискание степени магистра
по направлению
«Математика, компьютерные науки»
студентки гр. МКМ-240902,
Анненковой Ольги Геннадьевны

«___»_____2016 г.

Научный руководитель:
Авербух В.Л., к.т.н., доцент КИПУ

Научный консультант:
Васёв П.А., м.н.с. ИММ УрО РАН

Екатеринбург
2016

РЕФЕРАТ

Анненкова О.Г., СПЕЦИАЛИЗИРОВАННАЯ СИСТЕМА ВИЗУАЛИЗАЦИИ СЕТОЧНЫХ ДАННЫХ, квалификационная работа на степень магистра: стр. 42, рис. 19, библи. 10 назв., приложений 1.

Ключевые слова: научная визуализация, специализированные системы визуализации, трёхмерные структурированные сетки, интерактивная 3D-визуализация.

Объект исследования - трёхмерные структурированные расчётные сетки, используемые для численного решения дифференциальных уравнений методом конечных разностей. Цель работы - проектирование и реализация специализированной системы визуализации для визуального анализа качества сеточной конфигурации, формы ячеек и исследования особенностей внутренней структуры сеток. Результатом работы является интерактивное веб-приложение с трёхмерной графикой, применимое для исследовательской работы и решения практических задач прикладными математиками. Проект предоставляет возможности создания иллюстративных материалов для научных трудов и отчётов и вывода изображения на устройства виртуальной реальности.

Содержание

ВВЕДЕНИЕ	3
1. Сведения о расчётных сетках	6
2. Формальная постановка задачи	9
3. Технологии и инструменты	11
4. Устройство проекта и процесс загрузки	13
5. Реализация компонентов	15
5.1. Отображение узлов сетки	15
5.2. Генерация сеточных линий	16
5.3. Генерация псевдопрозрачных поверхностей	17
5.4. Фильтры для выделения особенностей	22
5.5. Настройки сцены и экспорт	24
ЗАКЛЮЧЕНИЕ	30
ССЫЛКИ И ЛИТЕРАТУРА	31
ПРИЛОЖЕНИЯ	33
Приложение 1. Руководство пользователя	33

ВВЕДЕНИЕ

Аппроксимации трёхмерных моделей расчётными сетками возникают в процессе численного решения уравнений математической физики, ставятся задачи представления, анализа и интерпретации научных данных.

Опыт, основанный на проектировании и разработке визуальных инструментов в ряде проектов ([1]), и анализ эффективности средств визуализации ([2]), показывают необходимость использования специализированных систем в нестандартных и исследовательских задачах, поскольку универсальные системы не способны обеспечить достаточную гибкость, возникают трудности при выходе за рамки типовых требований. Кроме того, результаты исследований и практика позволяют сформулировать основные принципы построения подобных систем: эффективность визуального представления и выделение характерных особенностей моделируемых объектов, интуитивный интерфейс человеко-машинного взаимодействия, архитектура системы в виде конструктора абстрактных модулей для создания представлений путём описания связей и зависимостей между ними, что способствует быстрому проектированию новых инструментов для решения конкретных задач, обеспечивает гибкое взаимодействие элементов и упрощает внедрение нового функционала.

Таким образом, целью работы является проектирование и реализация специализированной системы визуализации расчётных сеток. Актуальность, новизна и практическая ценность состоят в направленности проекта на текущие потребности прикладных математиков при решении задач сеточными методами.

Использование веб-среды как основы реализации системы мотивировано множеством преимуществ, среди которых кроссплатформенность приложений, интерактивные возможности и свободный удалённый доступ с отсутствием зависимостей в виде библиотек и компиляторов. Современные веб-

браузеры осуществляют поддержку библиотеки *WebGL* (Web-based Graphics Library), которая позволяет производить рендеринг 3D-графики на клиентской видеокарте и обеспечивает пути взаимодействия с устройствами виртуальной реальности.

В задачи данной работы входит разбор сеточных данных в предоставленном формате, разработка разнообразных функций и фильтров, включающих разреживание и выборки групп ячеек по индексам, отдельных сеточных линий, граней и узлов, изображаемых в соответствии с заданными скалярными значениями, построение управляющего интерфейса и необходимой графики с возможностью экспорта иллюстраций для дальнейшего использования в научных трудах.

Популярные на текущий момент графические пакеты ресурсоёмки и не могут в достаточной степени и удобном виде предоставить необходимый функционал и расширение пользовательскими функциями и компонентами. Платформа *ParaView*¹ ([3]), разработанная для визуализации больших объемов данных в исследовательских целях с использованием ресурсов удалённого сервера или распределённой среды, как сложный универсальный графический пакет с многочисленными настройками на практике вызывает трудности построения необходимых визуальных представлений. Утилиты семейства *Tecplot*² ([4]), специализирующиеся на моделировании физических процессов, в т.ч. задачах гидродинамики, входящих в сферу интересов наших пользователей, и активно ими применяемые, распространяются без исходных кодов и не могут быть доработаны.

Данный проект в отличие от существующих систем визуализации предполагает построение интерактивной трёхмерной сцены в веб-браузере с удобным интерфейсом построения выборок, и предоставит специфические функции, необходимые конкретным специалистам, а именно, пользователям тре-

¹<http://www.paraview.org/>

²<http://www.tecplot.com/>

буются возможности работы с коэффициентами качества ячеек различного типа, а также отображения выборок ячеек в контексте сетки.

Разрабатываемая система, соответствуя последним тенденциям и новым перспективным технологиям в области визуализации, предоставит возможности просмотра графики на устройствах виртуальной реальности, что является интересным экспериментом по использованию современных мультимедийных технологий в научных целях, поскольку это направление еще мало исследовано (обзор текущих проблем и вопросов в визуализации при использовании технологии VR (*virtual reality*) см. [5]). Очки виртуальной реальности позволяют выводить отдельные изображения для левого и правого глаза, а также осуществляют трекинг для отслеживания положения в пространстве, наклонов и поворотов, что способствует погружению пользователя в абстрактную среду. Такое явление располагает к достижению инсайт-эффекта, который считается целью визуальных представлений информации и основным аспектом в анализе качества систем визуализации. Разумеется, не предполагается, что использование устройств виртуальной реальности на практике станет более продуктивным подходом, чем другие методы визуализации, однако, имеет смысл предоставлять графику наиболее высокого доступного уровня реалистичности.

Работа реализована в виде модуля под интерфейс *Viewlang*³, разработанный научным консультантом, которому также принадлежат некоторые предложения по организации сцены. Основное устройство сцены в виде объектов, упомянутых в главе 4, разработка и реализация компонентов из главы 5 и их алгоритмов принадлежат автору. Тестовые сеточные модели предоставлены специалистами⁴. Методологические подходы основаны на исследованиях научного руководителя.

³<http://viewlang.ru/>

⁴Отдел прикладных задач ИММ УрО РАН, Н.А. Артёмова, О.В. Ушакова.

1. Сведения о расчётных сетках

Построение расчётных сеток применяется при использовании *метода конечных разностей* для решения дифференциальных уравнений путём замены производных разностными схемами. Основное преимущество применения данного метода в решении прикладных задач - высокая производительность вычислений за счёт широких возможностей распараллеливания.

Расчётная сетка - совокупность узлов, заданных в области определения некоторой функции, аппроксимирующих область для применения численного метода. В общем виде построение сетки означает построение непрерывного отображения из вычислительной области более простого вида на физическую.

Структурированная (регулярная) сетка предполагает наличие отношения порядка на множестве сеточных узлов, задается кусочно-гладким⁵ отображением из параметрического пространства P с целочисленными координатами узлов в физическое пространство G .

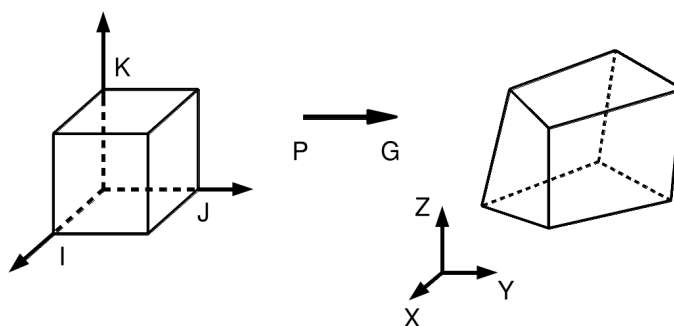


Рис. 1: Отображение для построения трёхмерной ячейки.

В данном проекте моделируемыми объектами являются *трёхмерные многоблочные регулярные сетки* (рис. 1, 2, 3), расчётная область представляет собой набор прямоугольных параллелепипедов, физическая область –

⁵Отображение непрерывно дифференцируемо на каждом из интервалов.

совокупность криволинейных шестигранников, разбиение и отображение имеют следующий вид:

$$P = \{\xi = (\xi_1, \xi_2, \xi_3), 0 \leq \xi_1 \leq I, 0 \leq \xi_2 \leq J, 0 \leq \xi_3 \leq K\}$$

$$s : P \rightarrow G; s = s(\xi_1, \xi_2, \xi_3),$$

где I, J, K - размерность сетки, $s_{i,j,k} = s(i, j, k)$ - дают координаты узлов, образы рёбер и граней кубических ячеек определяют линии и поверхности сетки.

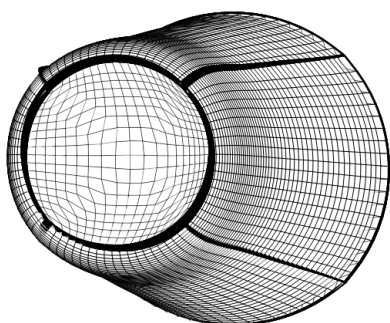


Рис. 2: Пример блока.

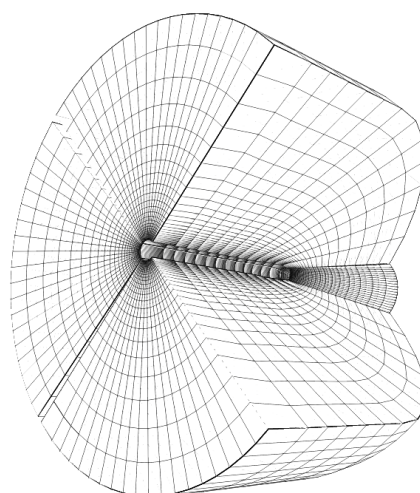


Рис. 3: Пример многоблочной сетки.

Невырожденность сетки (ячейки) означает, что отображение является биекцией (где инъективность исключает самопересечения и наложения ячеек, а сюръективность исключает зазоры и выход за пределы расчётной области). При этом важным показателем является значение якобиана используемого отображения в узлах сетки, который обычно используется как критерий невырожденности, отличность от нуля гарантирует взаимную однозначность в окрестности точки, а неотрицательность – сохранение ориентации векторов.

Значением якобиана является определитель матрицы частных производных:

$$J = \det\left(\frac{\partial s_i}{\partial \xi_j}\right)_{i=1,2,3;j=1,2,3}$$

Проблемы сеточного метода состоят в работе со случаями вырожденности (рис. 4), выкрученности и самопересечения ячеек (классификация ячеек см. [6]). На практике могут использоваться вырожденные сетки, существуют критерии выявления и алгоритмы диагностики и устранения вырожденных ячеек (условия невырожденности см. [7]), для особых случаев требуются модификации расчётов. В случаях вырожденных сеток системы алгебраических уравнений, заменяющие исходную дифференциальную задачу, являются плохо обусловленными, что оказывает влияние на точность решения.

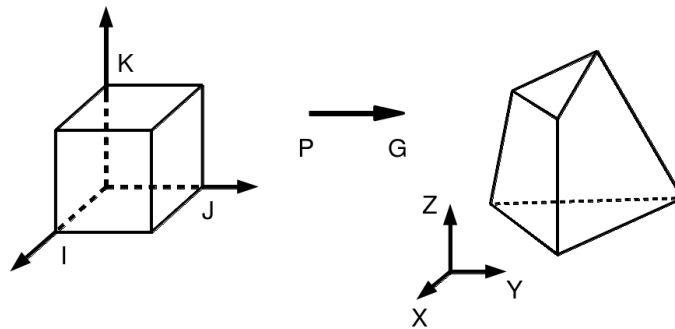


Рис. 4: Пример вырожденности.

Визуальные представления сеток необходимы для анализа качества сеточной конфигурации, структуры внутри блоков и на границах, выявления особых случаев и принятия решений для изменения сетки или применения различных методов расчетов.

Сложность визуализации сеток состоит в большом объеме графических данных, трудностях навигации и выделения особенностей, необходимостью управления многочисленными параметрами для построения разреженных выборок.

2. Формальная постановка задачи

Формат сеточных данных: *Tecplot ASCII* (*.dat).

Для каждого трёхмерного блока задается размерность I , J , K , далее N строк, где $N = I * J * K$, в каждой из которых описывается узел с текущими индексами i , j , k : координаты $x_{i,j,k}$, $y_{i,j,k}$, $z_{i,j,k}$ и набор скалярных значений.

Пример,

```
TITLE = "FINAL NODES, ELEMENTS "  
VARIABLES = "X", "Y", "Z", "D", "P", "M",  
ZONE I= 14, J= 20, K= 50, F=POINT  
-7.4611e+00 2.6450e-01 -2.5431e-01 8.8366e-01 8.4371e-01 5.1676e-01  
-8.0571e+00 3.6184e-01 -3.4790e-01 9.6576e-01 9.5247e-01 5.4574e-01  
-8.6695e+00 3.6256e-01 -3.4859e-01 1.0000e+00 1.0000e+00 5.0000e-01  
...  
TITLE = "FINAL NODES, ELEMENTS "  
VARIABLES = "X", "Y", "Z", "D", "P", "M",  
ZONE I= 20, J= 20, K= 50, F=POINT  
-7.4441e+00 -2.5544e-01 2.5544e-01 8.7689e-01 8.3477e-01 5.2378e-01  
-7.4407e+00 -2.7613e-01 2.3120e-01 8.7838e-01 8.3679e-01 5.2034e-01  
-7.4394e+00 -2.9306e-01 2.0853e-01 8.7919e-01 8.3788e-01 5.1793e-01  
...
```

Перечень необходимых функций и возможностей:

- Загружать сеточные данные из локальных файлов и по URL.
- Показывать все ячейки сетки, включая внутренние, выделять границу блоков, разреживать сетку по каждому направлению.
- Опционально отображать только внешние или только видимые элементы и линии в каждом направлении.
- Показать набор ячеек из указанного интервала с указанным шагом, ячейки выбранного блока по индексам.

- Показывать выбранные ячейки в контексте сетки.
- Показывать значения в узлах цветом с разными способами окрашивания для дискретных и вещественных скалярных значений.
- По клику на узел показывать информацию о нем - номер блока, индексы, координаты, скалярные значения.
- Предоставить возможности управления цветовой палитрой.
- Опциональное отображение осей координат с заданием подписей.
- Экспорт изображений в растровом и векторном виде (включая легенду с палитрой цветов узлов).

Цели и задачи специализированной системы:

- Быстрые алгоритмы построения графики.
- Интерактивные возможности для навигации по сцене и выбора узлов, интуитивно понятный интерфейс выбора опций и настройки параметров, минимизирующий пользовательские усилия для построения разнообразных видов отображения, удобные автодополнения и клавиши быстрого доступа.
- Реализация специфического функционала генерации палитр для работы с коэффициентами оптимальности ячеек и построения псевдопрозрачных поверхностей для обзора внутренней структуры блоков и выборок интересующих ячеек в контексте сетки.
- Возможность просмотра сцены с использованием устройств виртуальной реальности.

3. Технологии и инструменты

Рендеринг трёхмерной графической сцены производится в веб-браузере с поддержкой кроссплатформенной технологии *WebGL*⁶ (Web-based Graphics Library), предоставляющей низкоуровневые средства управления ресурсами видеокарты без использования дополнительных плагинов. Для построения 3D-графики используется библиотека *Three.js*⁷, которая позволяет определить настройки сцены, источников света, положения камеры, описать создание и изменение графических примитивов и материалов и осуществляет real-time рендеринг.

Определение объектов и описание связей между ними осуществляется на языке *QML* (Qt Meta-Object Language) на основе декларативно-императивной парадигмы, поскольку данный язык является синтаксически минималистичным и следует реактивному подходу, т.е. позволяет автоматически производить распространение изменений и пересчёт зависимых параметров, что является важным свойством при разработке фильтров, осуществляющих выборку элементов в соответствии с запросами пользователя как ключевом моменте в задачах исследования и анализа структуры отдельных частей моделируемого объекта. Разбор и выполнение qml-конструкций осуществляется с помощью библиотеки *QmlWeb*⁸.

Интерфейс *Viewlang* (см. [8, 9]) используется в качестве обёртки для *Three.js* и *QML*, унифицирующей проектирование специализированных систем визуализации для представления трёхмерных научных моделей. *Viewlang* предоставляет конструктор модулей, общие настройки *three.js*-сцены, функции для динамической асинхронной загрузки дополнительных скриптов и объектов. *Qml*-блоки используются в *viewlang*-сценах не только для управляющих элементов, но и для параметрического задания графических объектов. При-

⁶<https://www.khronos.org/webgl/>

⁷<http://threejs.org/>

⁸<https://github.com/qmlweb/qmlweb>

менение в данном проекте обусловлено необходимостью единой поддержки общего функционала для подобных задач (см. [10]).

Вывод изображения на устройства виртуальной реальности *VR HMD* (Head-Mounted Display) типа *Oculus Rift*⁹ возможен через интерфейс *WebVR API*¹⁰.

Основная функциональность описывается посредством языка *CoffeeScript*¹¹, который представляет собой синтаксический сахар для написания компактного кода, транслируемого в JavaScript. Сборка организуется с использованием утилиты *cake*.

Дополнительные модули, включенные в проект: графический интерфейс управления параметрами *dat.GUI*¹², библиотека для локального экспорта файлов *FileSaver.js*¹³, рендерер векторной –графики для Three.js *SVGRenderer.js*.

Таким образом, все используемые технологии и инструменты являются платформо-независимыми и самодостаточными, зависимостью для пользователя является только веб-браузер, весь код выполняется на JavaScript на клиентской стороне, производительность зависит от возможностей клиентского устройства. Все необходимые библиотеки свободно распространяемы, открытые исходные коды доступны под некоммерческими лицензиями. Проект является легковесным, но производительным решением, получаемая трёхмерная графика ускорена клиентским GPU, не задействуя серверных ресурсов.

⁹<https://www.oculus.com/en-us/rift/>

¹⁰В настоящее время доступны экспериментальные реализации спецификации WebVR API в тестовых сборках браузеров: <https://webvr.info/>

¹¹<http://coffeescript.org/>

¹²<https://github.com/dataarts/dat.gui>

¹³<https://github.com/eligrey/FileSaver.js/>

4. Устройство проекта и процесс загрузки

Иерархия объектов сцены под интерфейс `viewlang` и связи между ними: см. рис. 5, в схему не включены некоторые пользовательские управляющие элементы. Основные компоненты: оси координат, линии, поверхности, узлы сетки и загрузчик данных описаны объектами, унаследованными от имеющихся в интерфейсе, с реализацией конструктора с добавлением `three.js`-объектов на сцену, деконструктора и распространения изменений значений параметров. Основная императивная логика формирования графических примитивов реализуется функциями, содержащимися в отдельном скрипте.

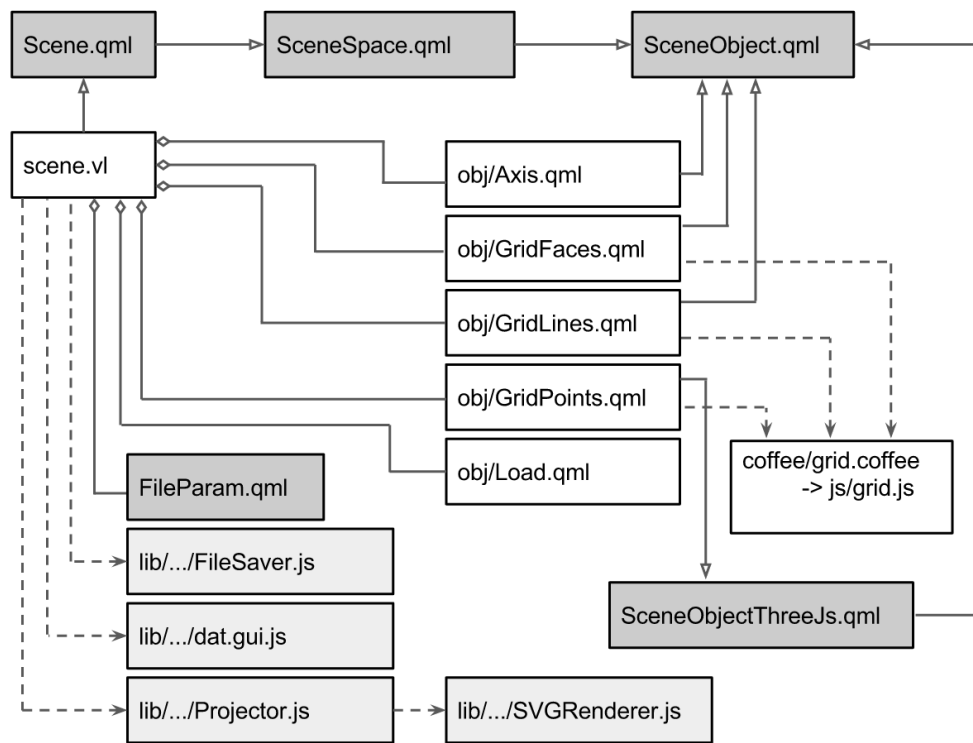


Рис. 5: Диаграмма объектов сцены.

В начале работы пользователем запрашивается стартовая страница со скриптами библиотек `three.js` и `qmlweb`, в `get`-переменной указан путь до основного файла сцены. Исходный код сцены и зависимости асинхронно скачиваются на клиентское устройство непосредственно из веб-интерфейса репозитория. Присутствующие среди зависимостей `qml`-объекты, входящие в интерфейс `viewlang`, подгружаются с его сервера, для поиска загружаются `qml`-файлы со списками объектов директорий. Сеточные данные могут загружаться из любого источника локального или удалённого, возможен множественный выбор. Пути удалённых файлов с данными и атрибуты позиции камеры могут указываться в URL `hash`-параметрами в `json`-формате.

Процесс загрузки объектов, скриптов и данных: см. рис. 6.

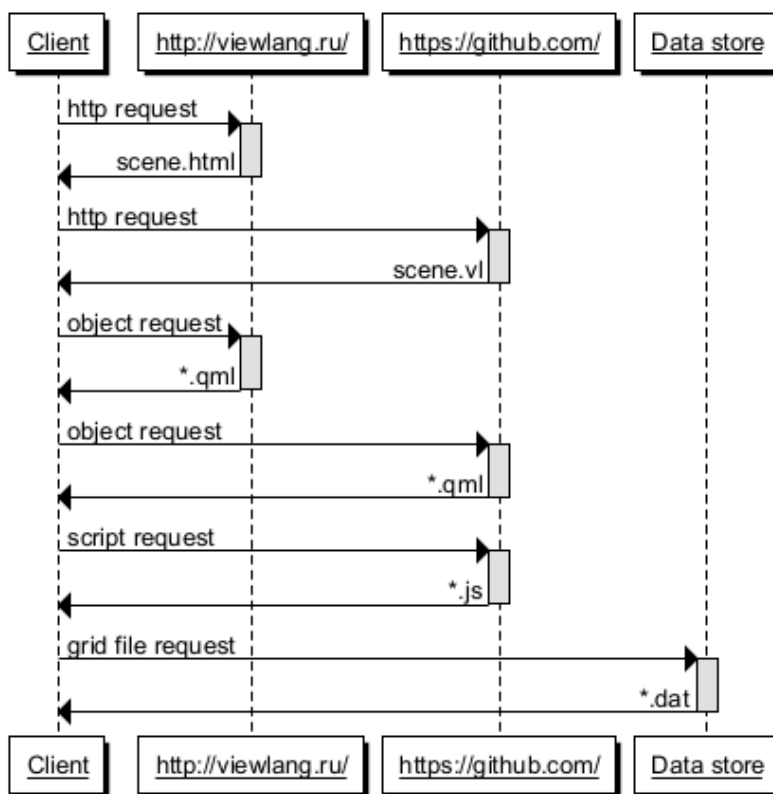


Рис. 6: Схема загрузки.

5. Реализация компонентов

Все графические элементы сцены имеют отдельные стилевые настройки, доступные для изменения пользователем, в том числе линии и поверхности в каждом направлении настраиваются отдельно, что позволяет строить сечения сетки в пространстве индексов. Для выделения интересующей выборки используются фильтры, для удовлетворяющих фильтру элементов применяется отдельный аналогичный набор настроек. Универсальность такого подхода состоит в том, что комбинации параметров позволяют получать большое количество разнообразных вариантов визуальных представлений.

Детализация сетки настраивается указанием шага для индексов каждого направления, что изменяет количество линий и поверхностей. При изменении набора отображаемых блоков построенные элементы не удаляются, а отмечаются как невидимые.

5.1. Отображение узлов сетки

Узлы сетки изображаются цветными квадратными билбордами (т.е. всегда обращены в сторону камеры) в интерактивном виде: по двойному клику выводится информация об узле, при этом увеличивается размер выбранного элемента. Цвет определяется в соответствии с указанным скалярным индексом. Во время парсинга сеточных данных сохраняются характеристики для каждого скалярного поля (глобальные для всех блоков): минимум и максимум, перечень целочисленных значений или `null`, если нашлось вещественное число. Таким образом, если скаляр оказался вещественнозначным, узлы окрашиваются градиентом, т.е. цвета получаются интерполяцией по заданному списку (интерполируется каждая компонента RGB-модели соседних цветов). В дискретном случае генерируется палитра перебором в круге Hue-компоненты HSL-модели с шагом в $1/3$ круга (с небольшими поправками

для разных случаев делимости и изменением других компонент для синих и зелёных цветов). Пользователю доступны настройки палитры редактированием списка цветов.

5.2. Генерация сеточных линий

Линии соединяют соседние узлы, отличающиеся одним индексом, по умолчанию окрашиваются в соответствии с отображением индексов блока на RGB-куб. Производительность графики на современных видеокартах для сеток порядка сотен тыс. блоков около 60 FPS¹⁴ (см. рис. 7).

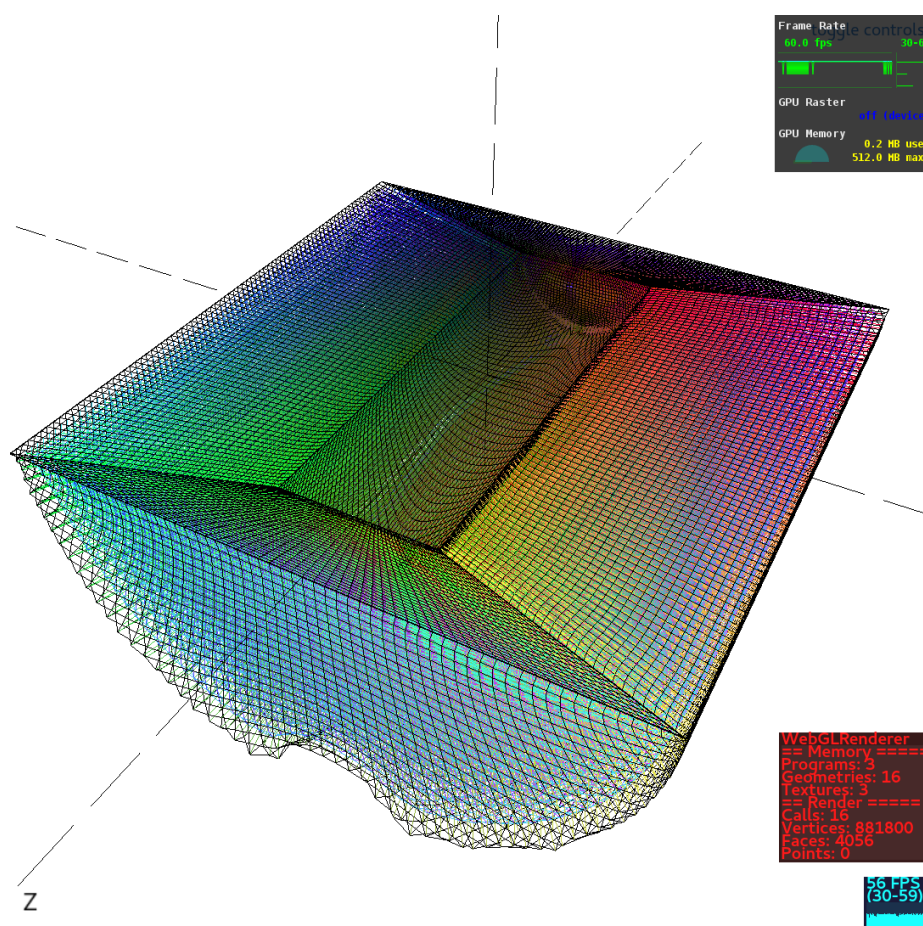


Рис. 7: Все линии сетки: около 56-60 FPS для сетки 12.8Мб (блок размерностью $61 * 81 * 30$).

¹⁴Frames Per Second - Кадровая частота.

5.3. Генерация псевдопрозрачных поверхностей

Рендеринг графики осуществляется методом рейкастинга (raycasting), быстрым вариантом рендеринга без полной трассировки лучей, поэтому эффект прозрачности зависит от порядка добавления элементов. Алгоритм добавления полигонов (*polygon mesh*), являющихся совокупностями треугольников, без сортировки связан с имеющимся на сетке отношением порядка, что позволяет получать симметричный эффект псевдопрозрачности достаточный для обзора внутренней структуры (см. рис. 12). При построении используется реализованный в three.js "merge" геометрий, на сетке порядка десятков тыс. блоков производительность графики составляет около 40 FPS для поверхностей одного направления, включая внутренние, и около 12 FPS для всех сеточных поверхностей (см. рис. 8). Обновление материалов (цветов и уровня прозрачности) происходит без полного перестроения объектов.

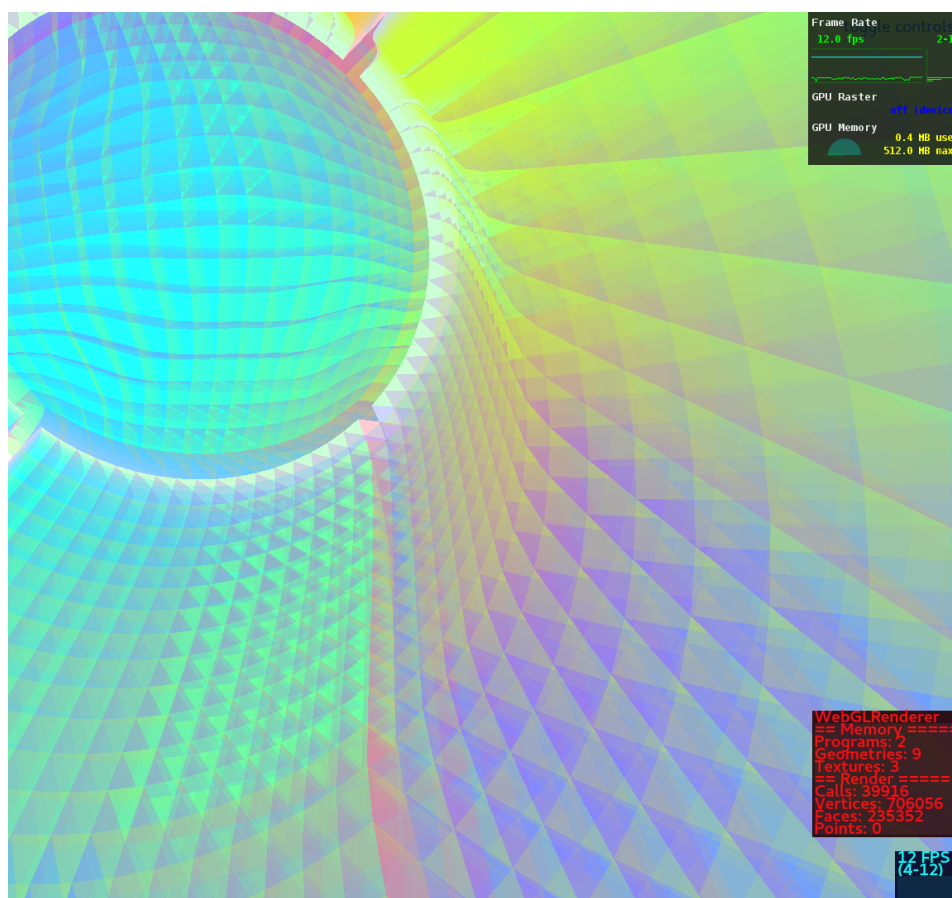


Рис. 8: Поверхности всех направлений: около 11-12 FPS для сетки 2.1Мб.

Алгоритм 1. Быстрый алгоритм генерации псевдопрозрачных полигонов блока регулярной сетки для рейкастинг рендеринга.

Для построения используется разбиение поверхностей на компоненты разного размера: в направлении K в полигон включается сечение сетки для фиксированного k , в направлении J полигоном является часть сечения для одного ряда ячеек, в направлении I в полигон входит грань одной ячейки. На каждом шаге добавляется односторонний многоугольник, эффект прозрачности в новом полигоне корректно отображается только для ранее добавленных элементов, обратная сторона является полностью прозрачной. Поверхности одного направления строятся с обеих сторон от краев блока нормальными, направленными в сторону противоположного края (см. рис. 9).

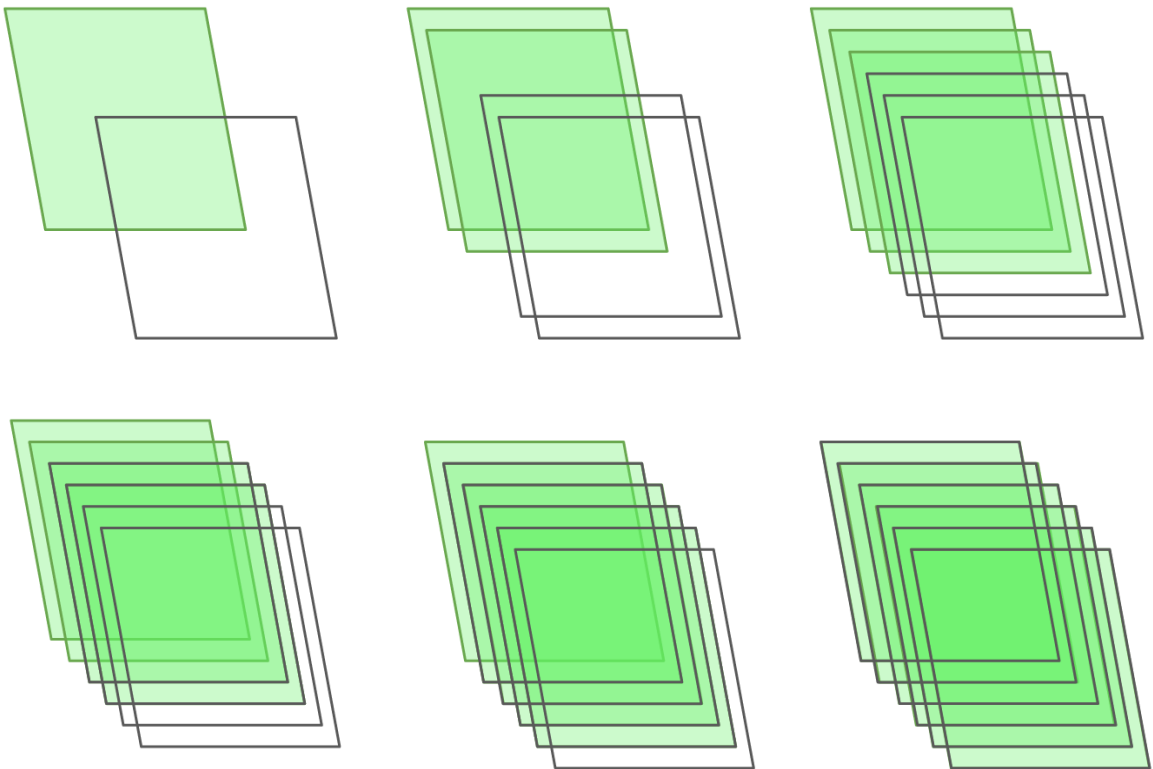


Рис. 9: Поверхности одного направления.

Для поверхностей двух направлений в середине построения полигонов первого направления, между следующими элементами размерностью $I * J$ узлов включается аналогичное построение многоугольников размера $K * 2$ от краев блока (см. рис. 10).

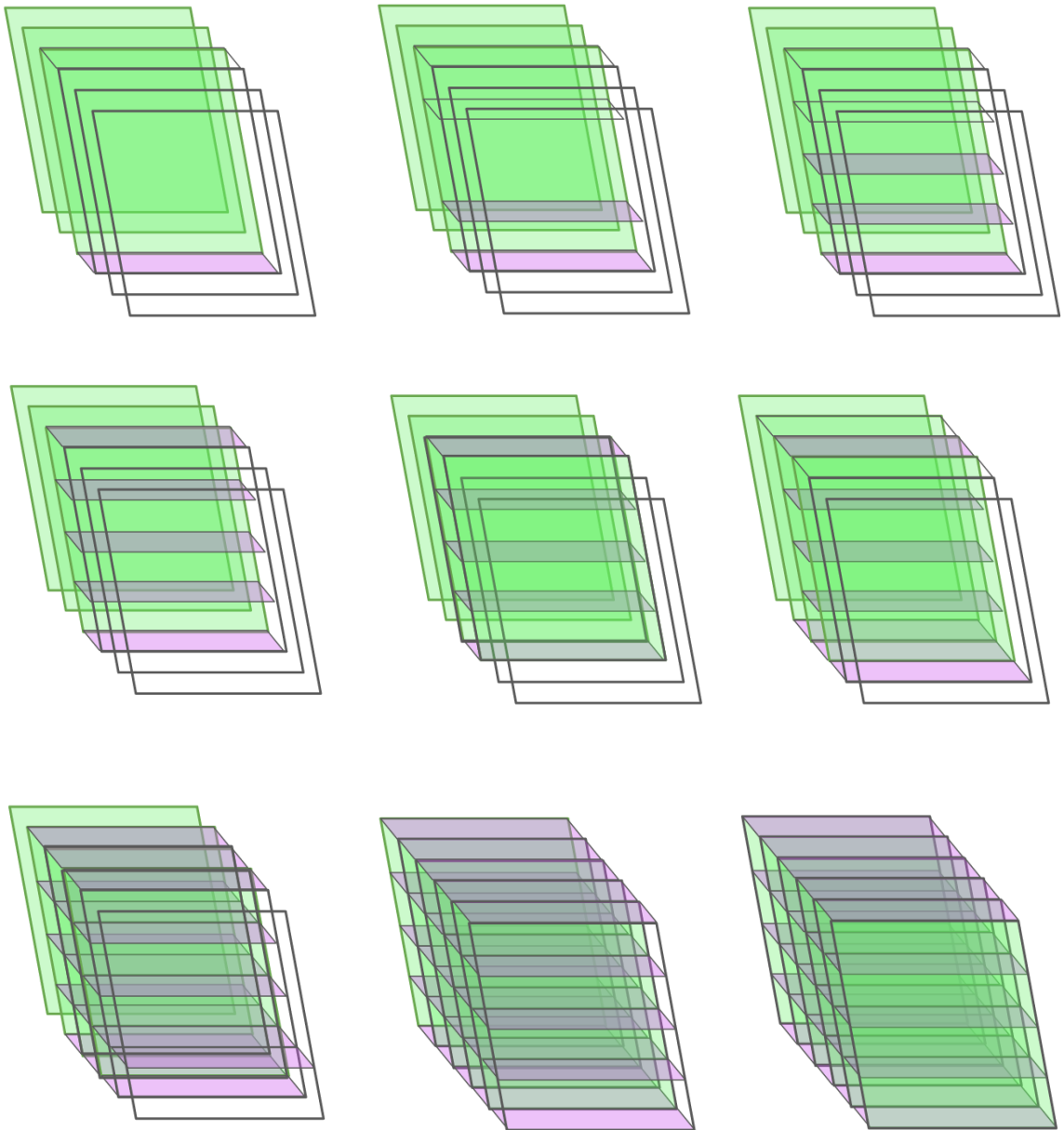


Рис. 10: Поверхности двух направлений.

В случае полного построения блока маленькие полигоны из 4 узлов (2 треугольника) генерируются во время построения первой половины элементов размера $K * 2$ нормальными внутрь блока, и наоборот для второй половины (см. рис. 11).

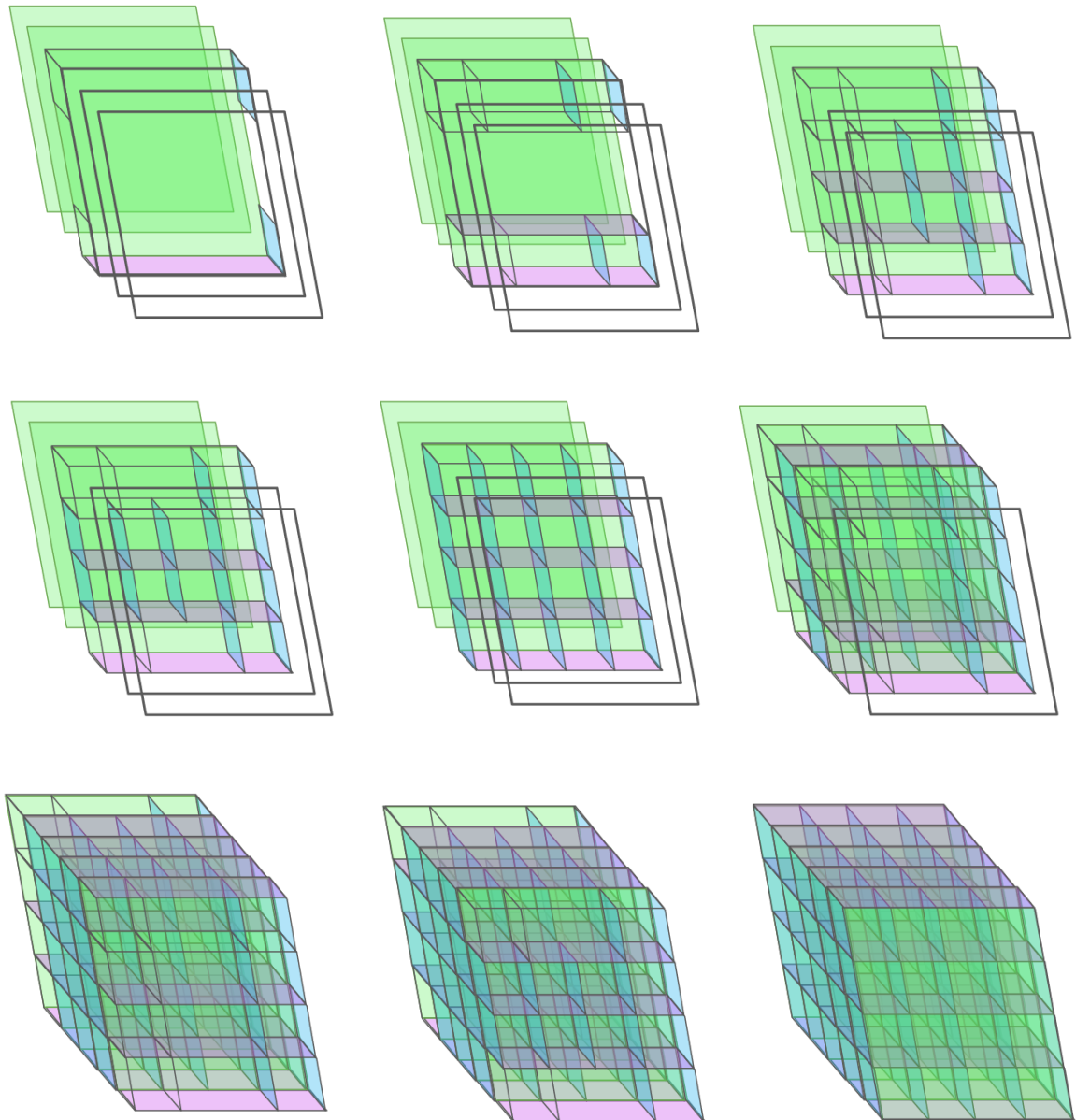
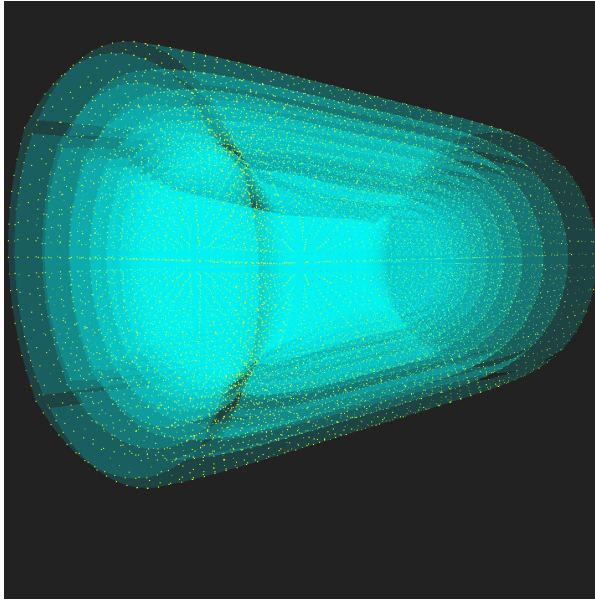
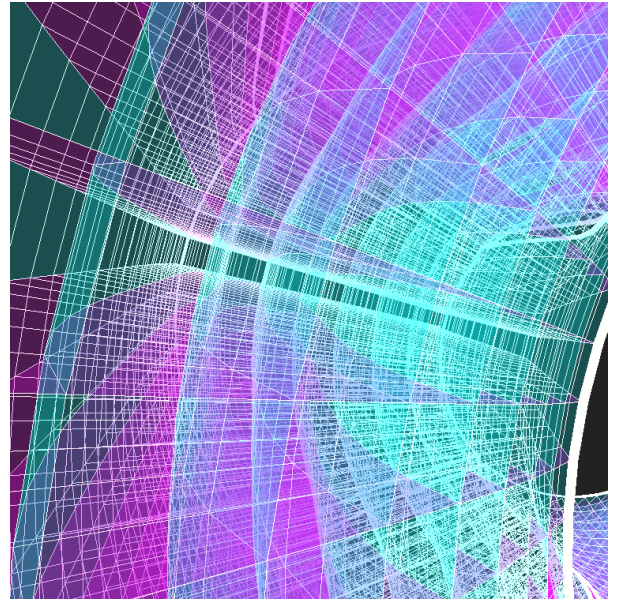


Рис. 11: Поверхности трёх направлений.

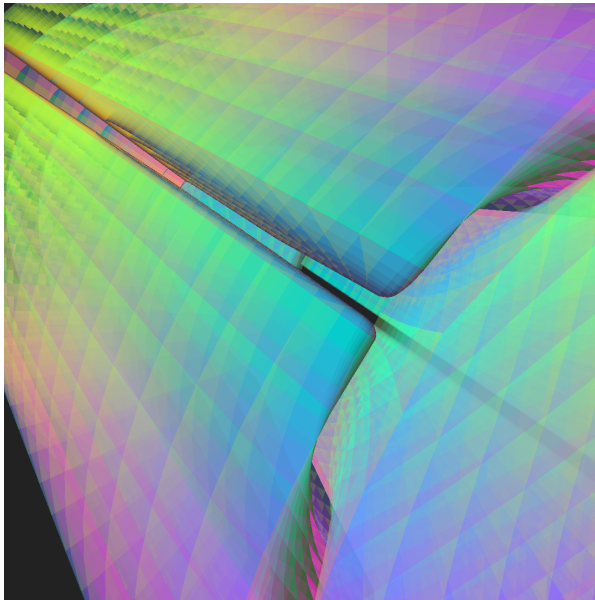
Для использования полигонов максимального размера, что является более производительным вариантом, чем несколько меньших, в соответствующих случаях осуществляются перестановки индексов i, j, k .



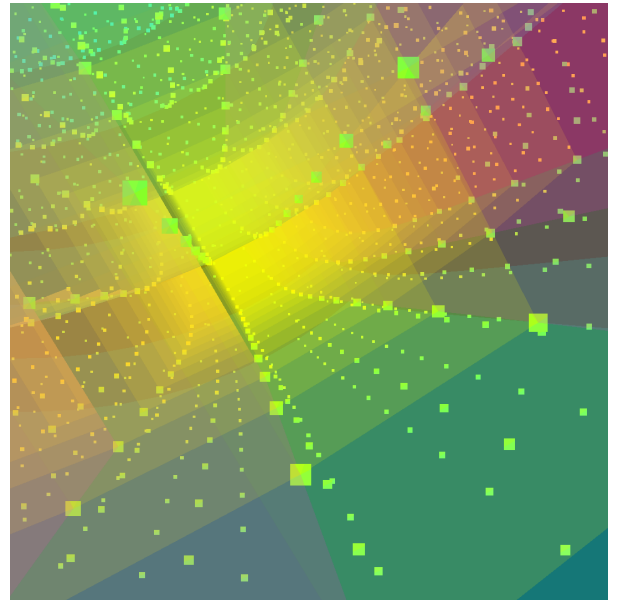
а).



б).



в).



г).

Рис. 12:

а), б) - поверхности одного и двух направлений.
в), г) - наружный и внутренний вид сетки.

5.4. Фильтры для выделения особенностей

Для выделения интересующей области используются фильтры, которые задаются функцией, проверяющей принадлежность узла сетки на основе его индексов, скалярных значений и заданных пользователем параметров. Для отфильтрованных элементов не применяются стили основной части, а настраиваются отдельно. Реализовано в виде трёх вариантов выборов.

- Фильтр в заданном блоке по заданному интервалу и шагу для индексов каждого направления.

Позволяет выделить ограниченное подмножество сетки: линию, поверхность, грань, ячейку, ряд или слой ячеек (см. рис. 13).

Для анализа различий сетки на разных этапах деформации два файла загружаются как отдельные блоки, один из которых целиком выделяется фильтром (см. рис. 14).

- Фильтр по отрезку скалярных значений.

Пользователь указывает номер скаляра и минимальное и максимальное значения. Фильтр выделяет ячейки целиком, т.е. проверка включает в себя просмотр предыдущих соседних вершин по каждому индексу, кроме нулевого, и следующих для предпоследних индексов.

Фильтр может использоваться для выделения деформированных ячеек (см. рис. 15), заданных дискретными коэффициентами, или особенностей значений функции, заданных вещественными числами (см. рис. 16).

- Фильтр по списку ячеек.

Пользователем задается список параметров для каждой ячейки: номер блока и индексы i , j , k . Для проверки принадлежности создается маска

в соответствии с размерностью сетки, ячейки выделяются целиком с проверкой соседних вершин.

Фильтр используется для выборок, которые нельзя задать другими фильтрами, например, просмотр соседних ячеек к вырожденным случаям (см. рис. 17) и совместимость ячеек на границах соседних блоков (см. рис. 18).

Поскольку фильтры могут задавать выборки ячеек с разреженностью отличной от основной части, списки индексов для циклов построения линий и поверхностей формируются как отсортированные объединения индексов основной части и индексов фильтра с разными детализациями.

В процессе формирования поверхностей полигоны для внутренних элементов фильтра и внешних строятся на едином массиве вершин и разных массивах треугольников. Треугольник из трёх номеров вершин добавляется в поверхность, к которой применяются стили фильтра, если удовлетворяет условиям выборки и индекс слоя входит в детализацию фильтра или добавляется как внешний к фильтру, если не подходит под условия и принадлежит детализации основной части, и не добавляется вообще в остальных случаях. Таким образом, на каждом шаге формируется два списка треугольников, которые образуют поверхности в соответствии с отдельно заданными параметрами отображения.

При генерации линий для отрезка аналогично проверяется условие фильтра и принадлежность детализации при фиксированных индексах двух направлений, в результате чего применяются соответствующие стилевые настройки.

Пример выборки ячеек с псевдопрозрачными поверхностями, заданной фильтром с разреженностью отличной от основной части: см. рис. 19.

5.5. Настройки сцены и экспорт

Сцена содержит координатные оси с текстовыми подписями. Поскольку подписи должны иметь одинаковый размер в экранных координатах, поправка масштаба рассчитывается как произведение расстояния от подписи до камеры на константу.

Экспорт снимка сцены в растровом виде осуществляется через контекст canvas-элемента, содержащего three.js-сцену. Для векторного представления используется аналогичный рендерер с полной копией сцены. Для устранения различий рендеринга неэкспортируемые объекты добавляются отдельно, для этого некоторые элементы генерируются в двух версиях, при обновлениях изменяются все дубликаты. В частности, подписи осей в WebGL представляются текстурой, а в векторном виде простым текстом. Экспорт легенды с палитрой скалярных значений также возможен в двух вариантах, при этом для дискретных значений подписывается каждый цвет, а для вещественных изображается градиент с подписями минимального и максимального значений.

Справка по использованию пользовательского интерфейса:
см. Приложение 1.

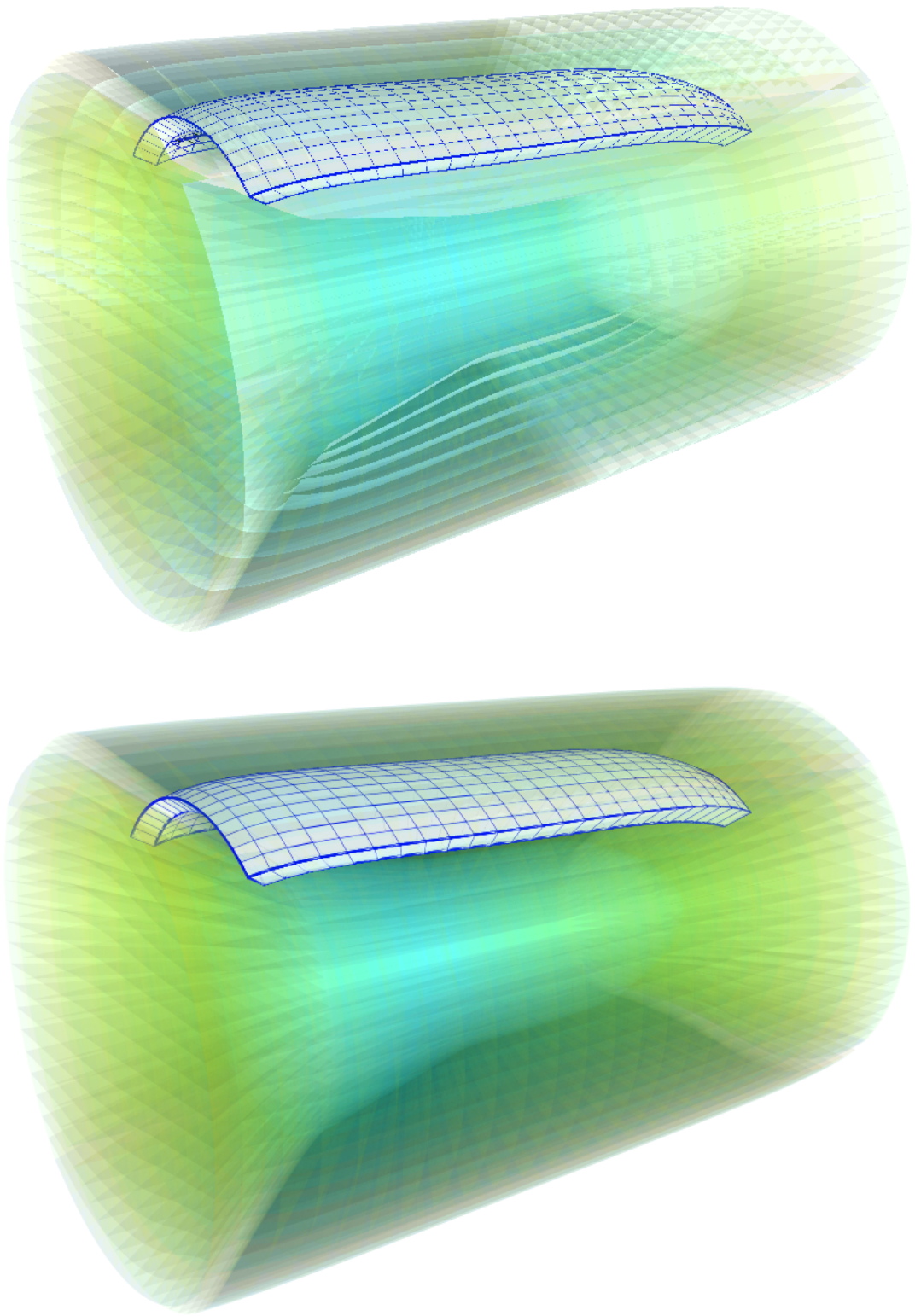


Рис. 13: Слой ячеек в контексте сетки в виде псевдопрозрачных поверхностей в WebGLRenderer и прозрачных в SVGRenderer.

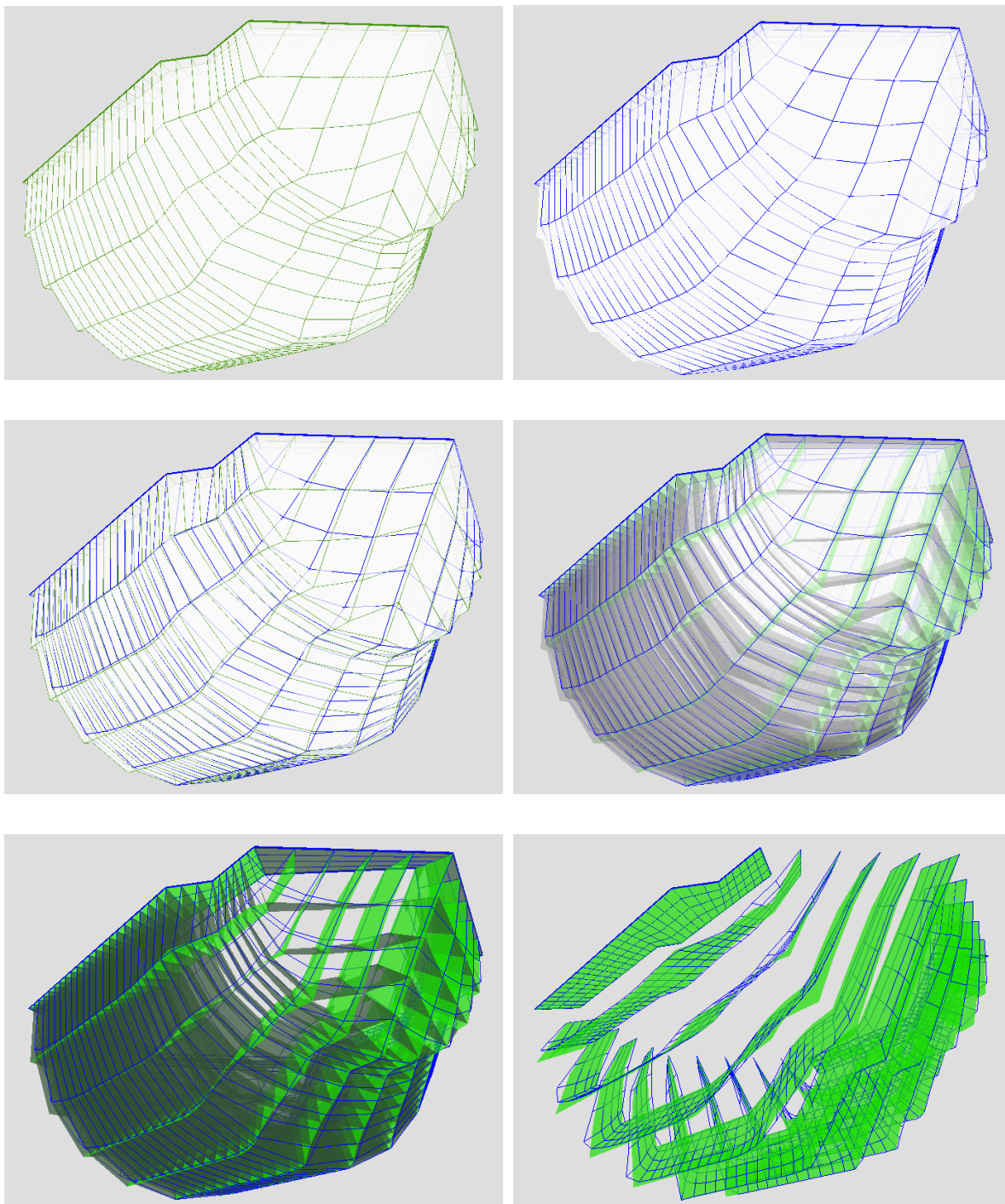


Рис. 14: Два варианта сетки и совместные отображения.

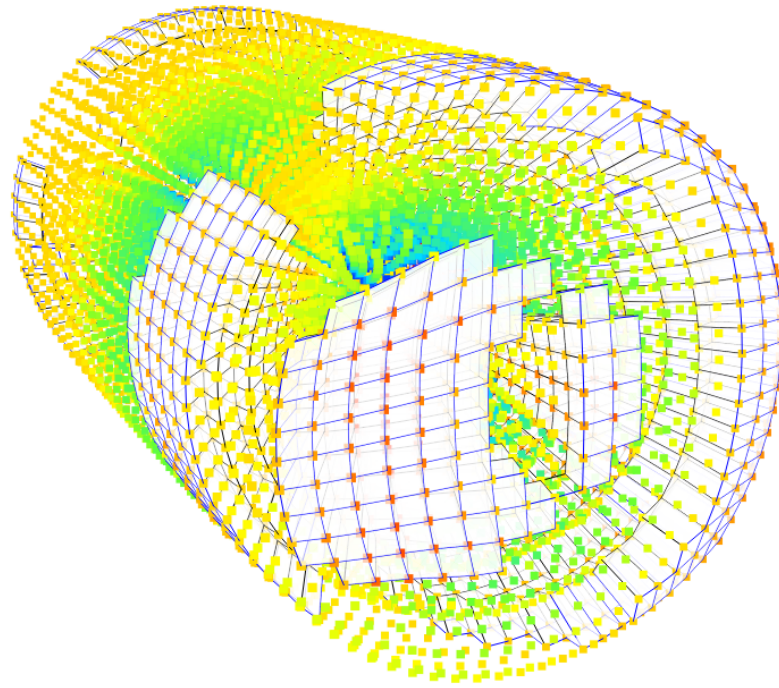


Рис. 15: Выборка ячеек по значению функции.

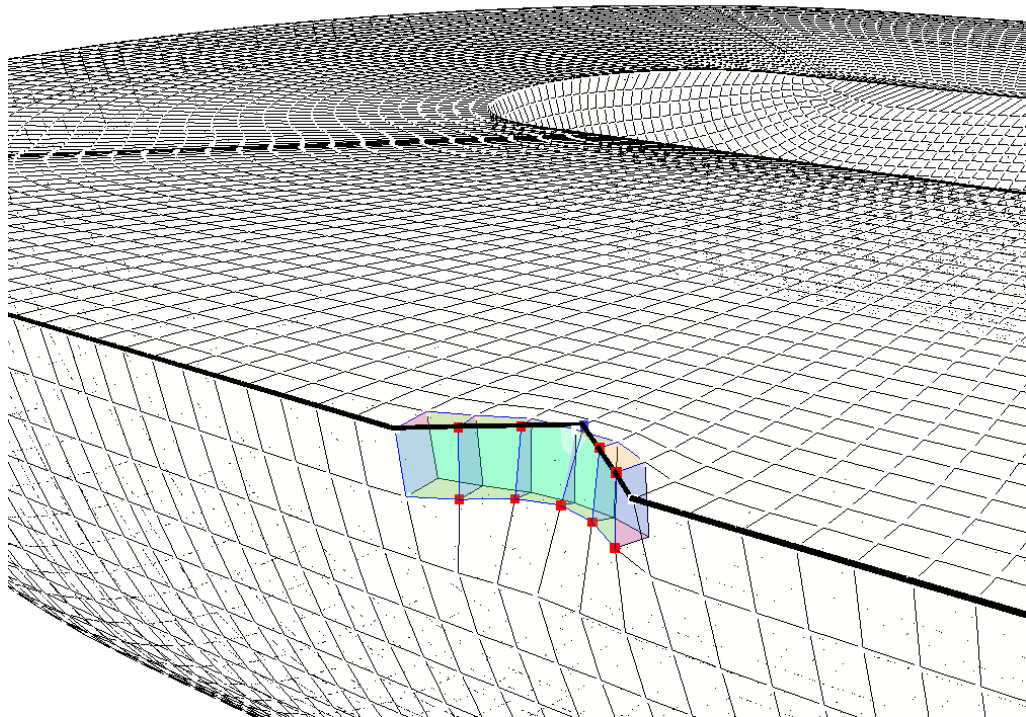


Рис. 16: Деформированные ячейки.

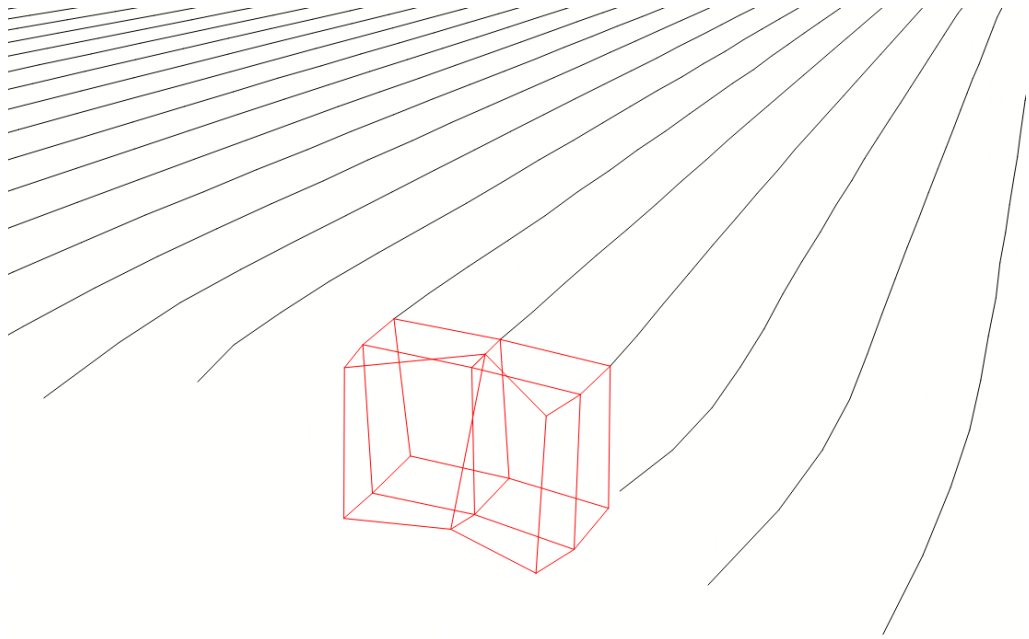


Рис. 17: Выкрученные ячейки пересекаются с правильными шестигранными.

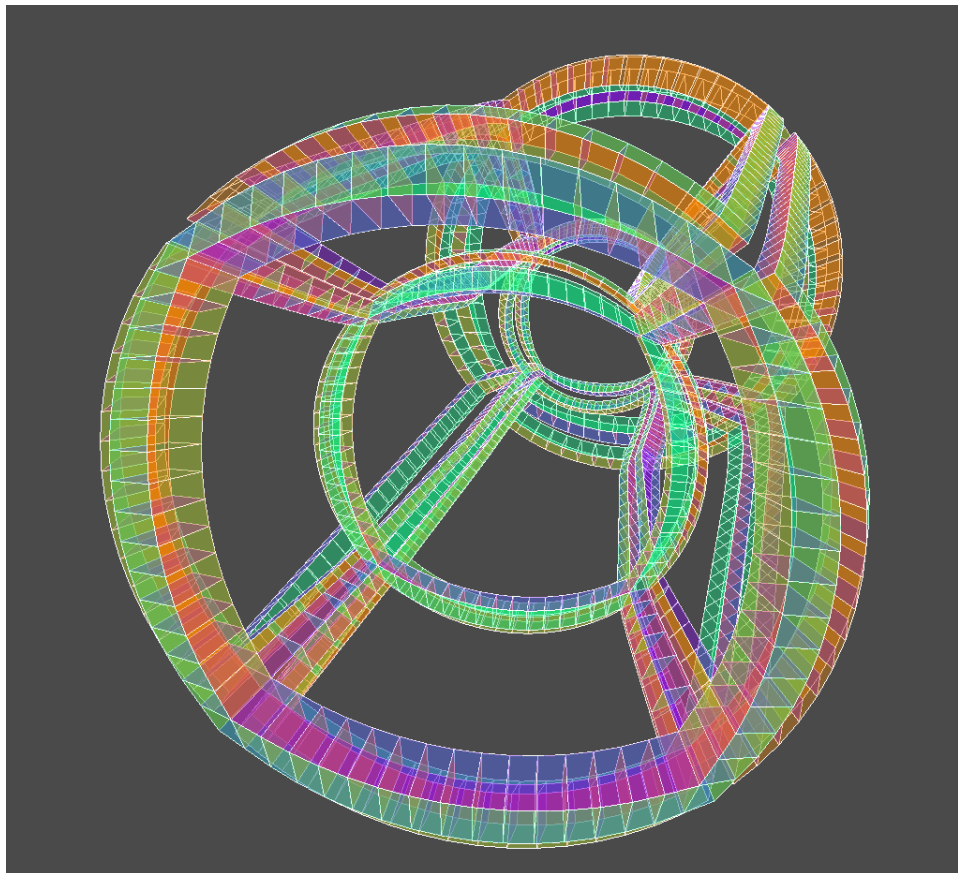


Рис. 18: Фильтр ячеек на границах блоков.

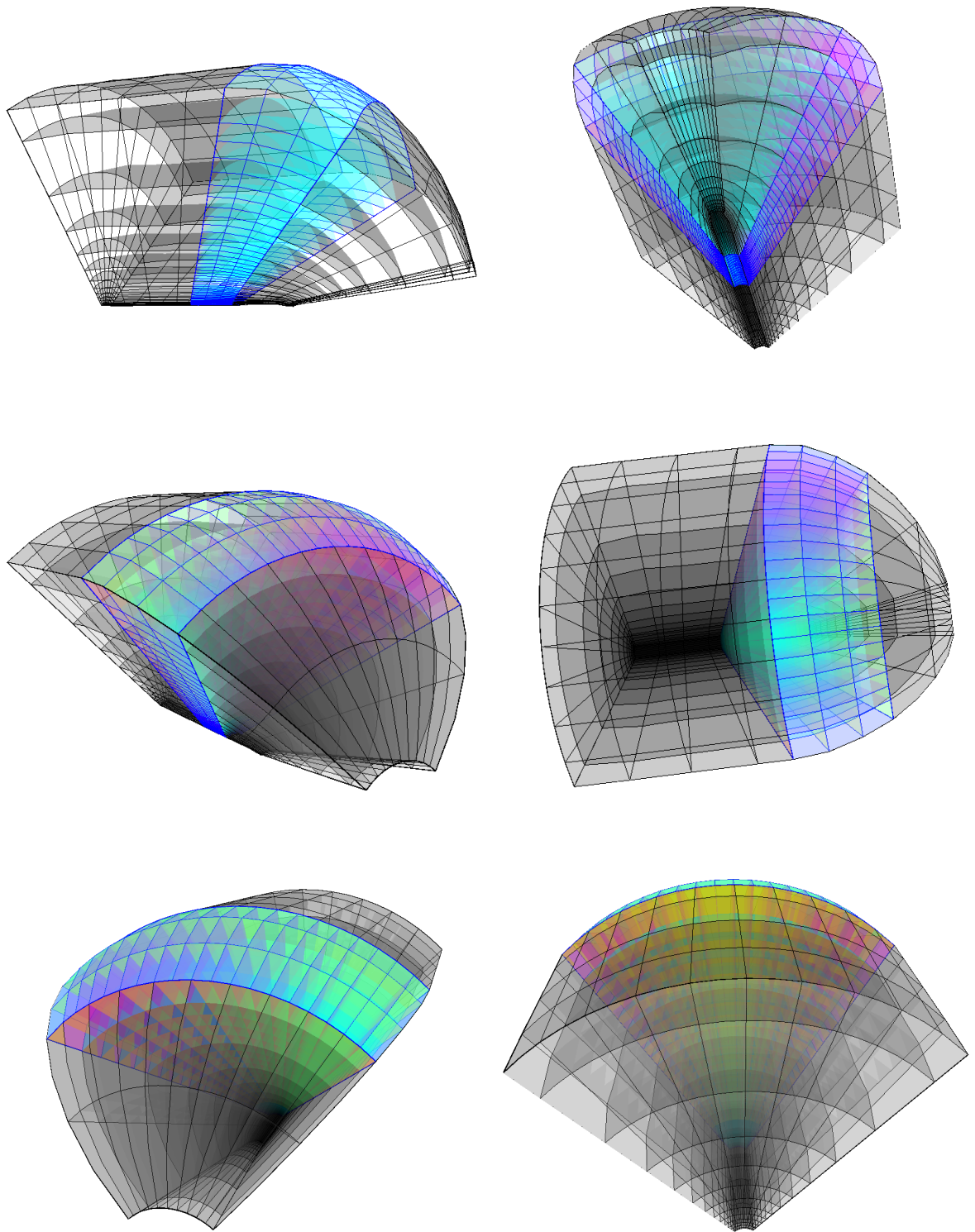


Рис. 19: Выделение ограниченного подмножества блока с более подробной детализацией.

ЗАКЛЮЧЕНИЕ

Разработанная система визуализации¹⁵ ориентирована на уникальных пользователей – научных сотрудников, позволяет исследовать структуру регулярных сеток и создавать иллюстративные материалы. В ходе работы реализованы оригинальные алгоритмы построения графических представлений сеточных данных с выделением особенностей. Инструмент спроектирован в соответствии с принципами легковесности, доступности и гибкой расширяемости, функционал покрывает поставленные задачи. Как правило, при создании специализированных инструментов визуализации основной проблемой является то, что корректно интерпретировать построенные сущности и качество визуального представления способен исключительно автор моделей, эффективный результат может быть достигнут только в постоянном диалоге с пользователями, поэтому на различных этапах были проведены тестовые демонстрации, которые определили детали поставленных задач, план реализации и дальнейшие перспективы. Данный проект открывает перспективы для разработок сразу в нескольких направлениях, среди возможных задач: интерактивная коррекция сеток, online-мониторинг деформаций, рассчитываемых на кластере, внедрение жестовых интерфейсов для управления и взаимодействия с объектами в пространстве виртуальной реальности. Возникают также вопросы полезности и комфортности использования VR-режима, насколько проявляется негативное влияние аппаратных задержек обновления изображения на человеческое восприятие, может ли использование такой технологии принести какие-то преимущества в научной работе.

¹⁵<http://viewlang.ru/grid>

ССЫЛКИ И ЛИТЕРАТУРА

1. Авербух В.Л., Анненкова О.Г., Бахтерев М.О., Манаков Д.В., Задачи визуализации программного обеспечения параллельных и распределенных вычислений // Международная научная конференция Параллельные вычислительные технологии 2014. Южный федеральный университет (1-3 апреля 2014 г., г. Ростов-на-Дону). Сборник трудов. Стр. 7–18.
2. Авербух В.Л., Анненкова О.Г., Бахтерев М.О., Манаков Д.В., Анализ и оценка систем визуализации программного обеспечения параллельных вычислений // Вопр. атомной науки и техники (ВАНТ). Сер. Мат. моделирование физ. процессов. 2015. Вып.4. С. 58-70.
3. Ahrens J., Geveci B., Law C., ParaView: An End-User Tool for Large Data Visualization // Visualization Handbook, Elsevier, 2005, pp. 717-731.
4. Scott T. Imlay, Craig Mackey, Improved Performance of Large Data Visualization using Sub-Zone Load-On-Demand // Tecplot Inc., Bellevue, WA, 2013.
5. Olshannikova, E., Ometov, A., Koucheryavy, Y. and Olsson, T., Visualizing Big Data with augmented and virtual reality: challenges and research agenda. // Journal Of Big Data, 2(1), 2015, pp.1-27.
6. Ушакова О.В., Классификация шестигранных ячеек // Журнал вычислительной математики и математической физики, 2008, том 48, номер 8, 1406-1428.
7. Ушакова О.В., Условия невырожденности трехмерных ячеек. Формула для объема ячеек // Журнал вычислительной математики и математической физики, 2001, том 41, номер 6, 881-894.

8. Васёв П.А., Бахтерев М.О., Декларативно-императивный метод конструирования сцен научной визуализации // XV Международная конференция «Супервычисления и Математическое Моделирование». Тезисы. ФГУП «РФЯЦ ВНИИЭФ». Саров. 2014, стр. 46.
9. Авербух В.Л., Бахтерев М.О., Васёв П.А., Манаков Д.В., Стародубцев И.С., Развитие подходов к разработке специализированных систем компьютерной визуализации // GraphiCon2015 : Междунар. науч. конф., Протвино, 22-25 сентября 2015: труды / АНО Науч. о-во «ГРАФИКОН»; Ин-т физ.-техн. информатики. Москва; Протвино, 2015. С. 17-21.
10. Васёв П.А., Опыт применения конструктора сцен научной визуализации // Труды международной научной конференции “Параллельные Вычислительные Технологии” (ПаВТ’2015). Екатеринбург, 31 марта - 2 апреля 2015. Челябинск, Издательский центр ЮУрГУ. 2015. Стр. 500.

Специализированная система визуализации сеточных данных.

<http://viewlang.ru/grid>

Описание

Интерактивное веб-приложение для исследования внутренней структуры многоблочных регулярных сеток с шестигранными ячейками на основе библиотеки трёхмерной визуализации `three.js` и интерфейса описания сцен `viewlang`.

Формат данных: Tecplot ASCII.

Требования: Веб-браузер с поддержкой WebGL.

Желательно Google Chrome (<https://www.google.ru/chrome/>) или Mozilla Firefox (<https://www.mozilla.org/en-US/>) последних версий.

Допустимые размеры сеток и производительность зависят от количества свободной оперативной памяти и возможностей видеокарты.

В случае нехватки ресурсов браузер может вернуть ошибку, после чего нужно обновить страницу.

Управление:

Навигация по сцене:

- `left click` – вращение вокруг центра,
- `scroll` – изменение расстояния до центра вращения,
- `right click` – смещение центра.

Подтверждение ввода текстового поля: **Enter**.

Просмотр по одной ячейке:

- `b`, `B` – переход по блокам,
- `k`, `K`, `j`, `J`, `i`, `I` – сдвиг по каждому направлению.

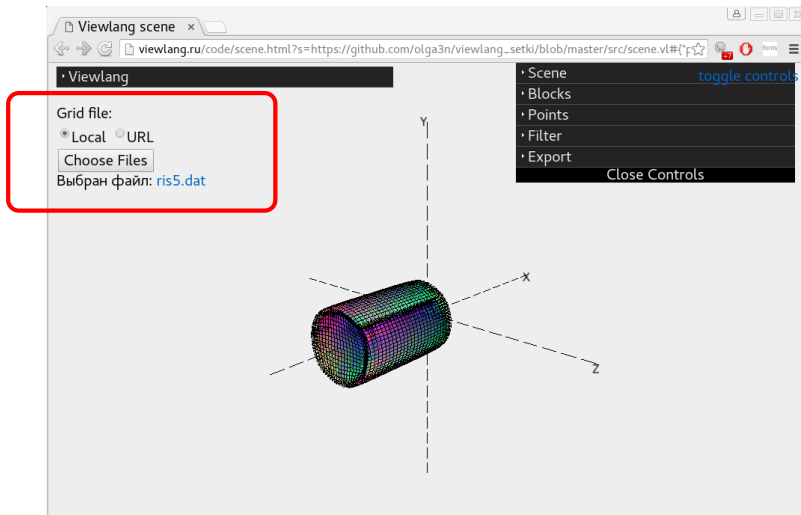
Выбор точки:

- `double click` – просмотр информации по выбранному узлу сетки.

Скрыть управляющие элементы и меню: **Ctrl-Q**, **h**.

Импорт

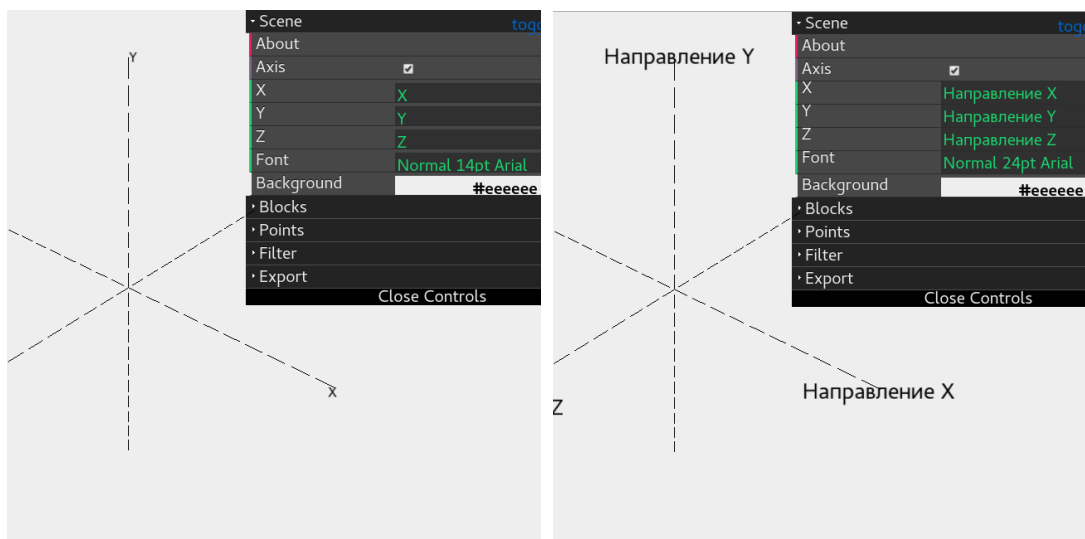
Поддерживаются файлы формата Tecplot ASCII, размером до нескольких десятков МБ, доступна загрузка по URL-адресам и чтение локальных файлов (выбор нескольких элементов с **Ctrl**).



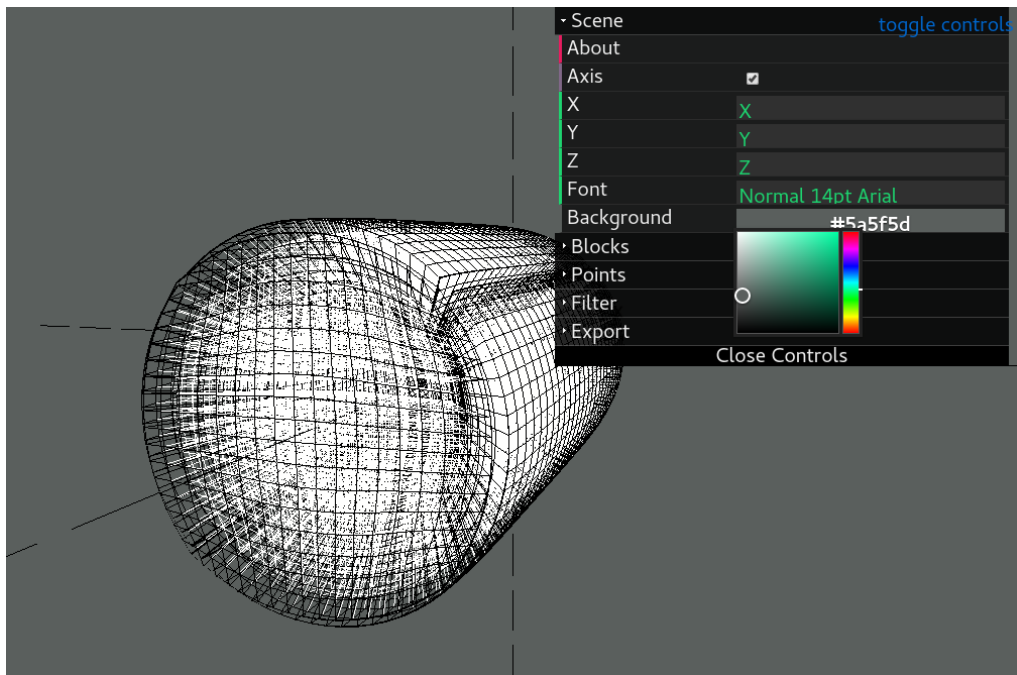
Опции

1. Настройки сцены:

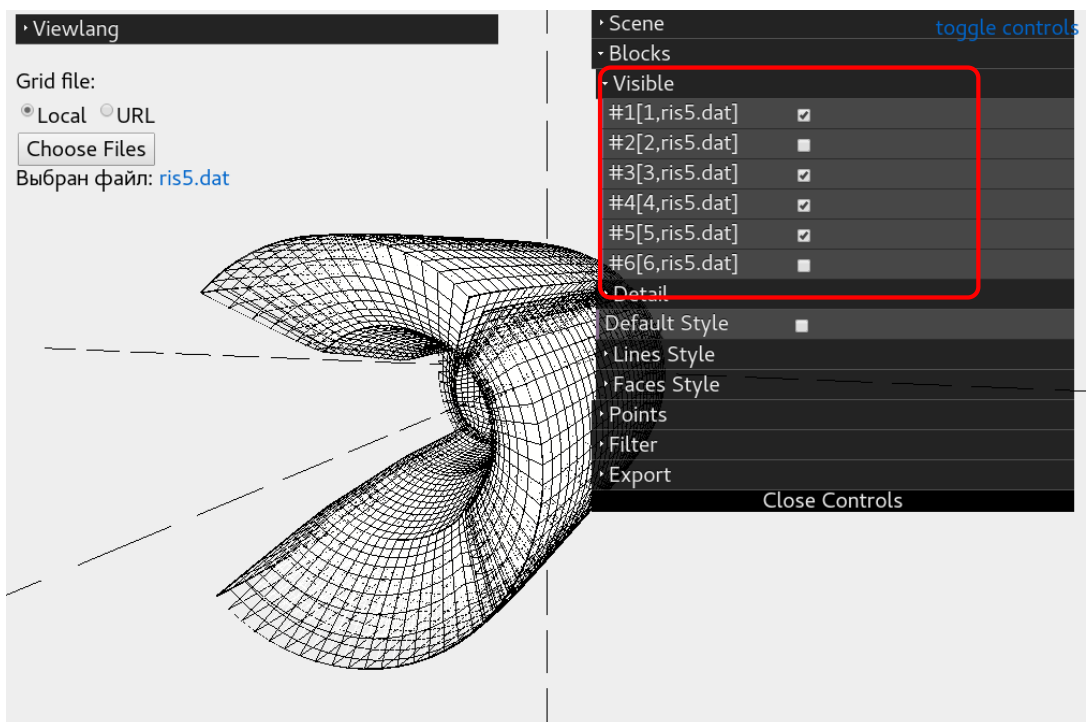
В разделе **Scene** содержится справочная информация по клавишам быстрого доступа, управление подписями осей координат и настройками шрифтов. (Отключение отображения осей координат: **Scene** -> **Axis**)



Указание цвета фона сцены: Scene -> background.

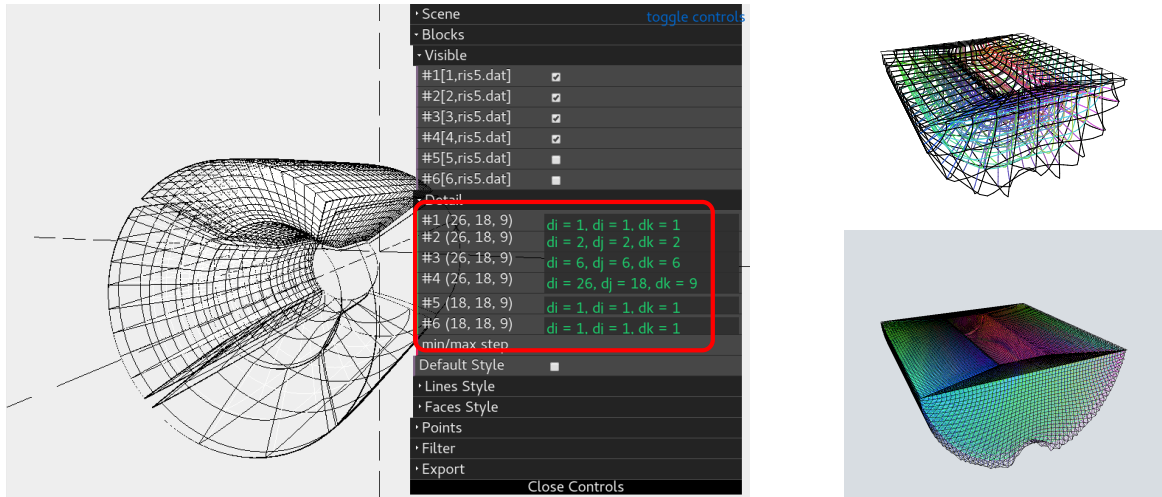


2. Выбор блоков: Blocks -> Visible



3. Разрезивание: Blocks -> Detail

Указание шага по каждому направлению для каждого блока (в скобках выводятся размерности блока I, J, K).



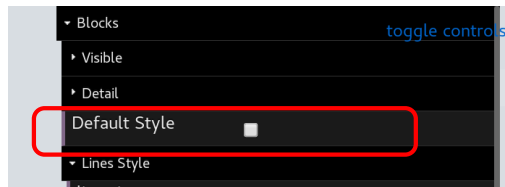
4. Стили линий и поверхностей

Для задания стилей отключить настройки по умолчанию:

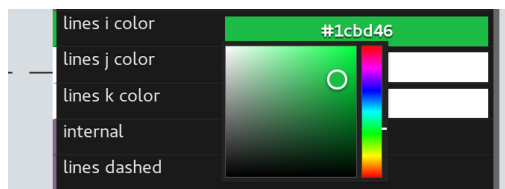
Blocks -> Default Style.

Меню настроек для для линий и поверхностей основной части:

Blocks -> Lines Style, Blocks -> Faces Style.



Цвета элементов для каждого направления задаются с помощью *Color Picker* или вводом шестнадцатеричной записи цвета в RGB-модели.



Для линий доступно пунктирное начертание, отключение отображения внутренних линий и линий по каждому направлению, для поверхностей аналогично задаются направления и отображение внутренних элементов, а также возможно указание уровня прозрачности.

Варианты отображения линий и поверхностей сетки.

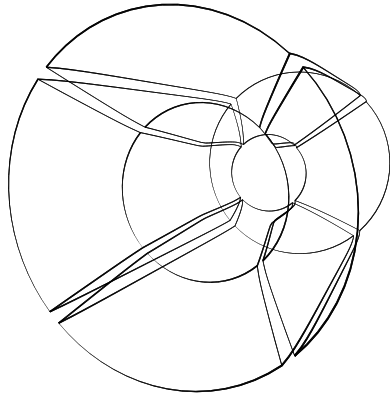


Рис. 20: Границы блоков.
Указать максимальную разреженность (шаг равный размерности блока)
(Blocks -> Detail).

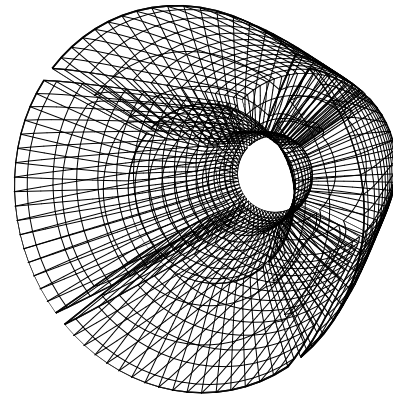


Рис. 21: Внешние линии блоков.
Отключить отображение внутренних линий:
Blocks -> Lines Style -> internal.

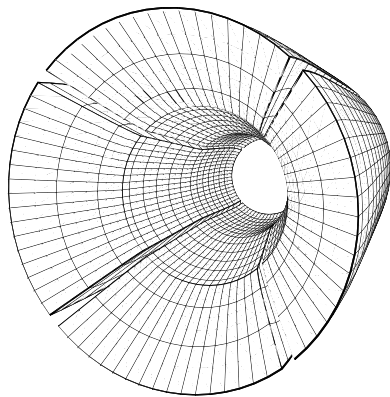


Рис. 22: Видимые линии сетки.
Добавить непрозрачные внешние поверхности
(Blocks -> Faces Style, faces opacity: 1).

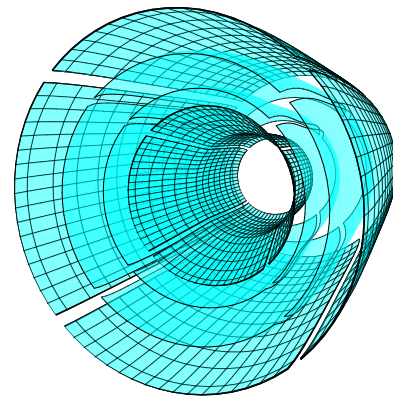


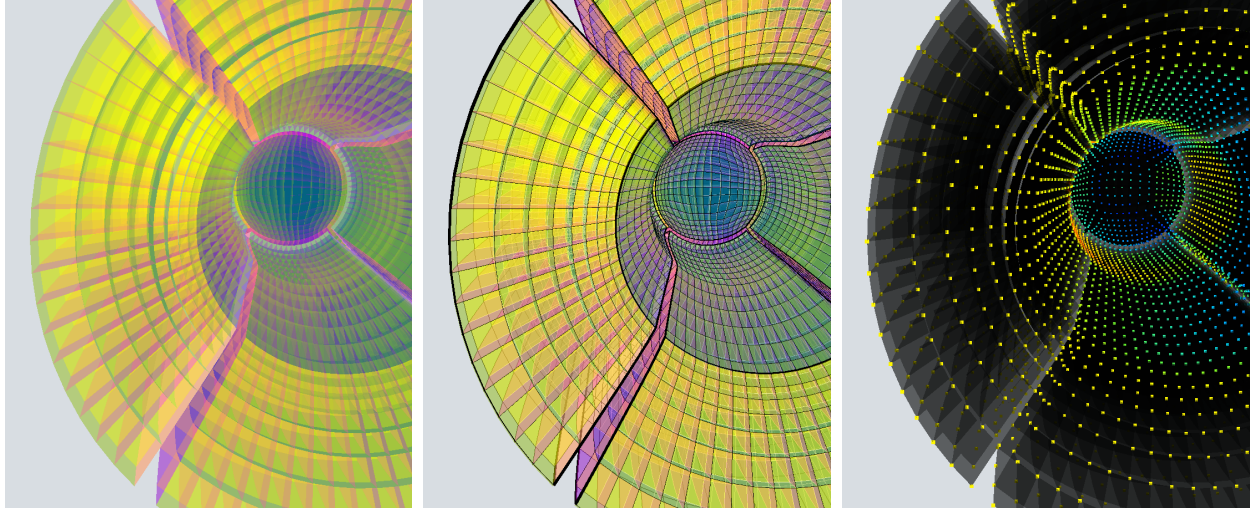
Рис. 23: Слои направления.
Отображение поверхностей одного направления и линий двух соответствующих направлений.

Поверхности всех направлений

Включить отображение внутренних поверхностей:

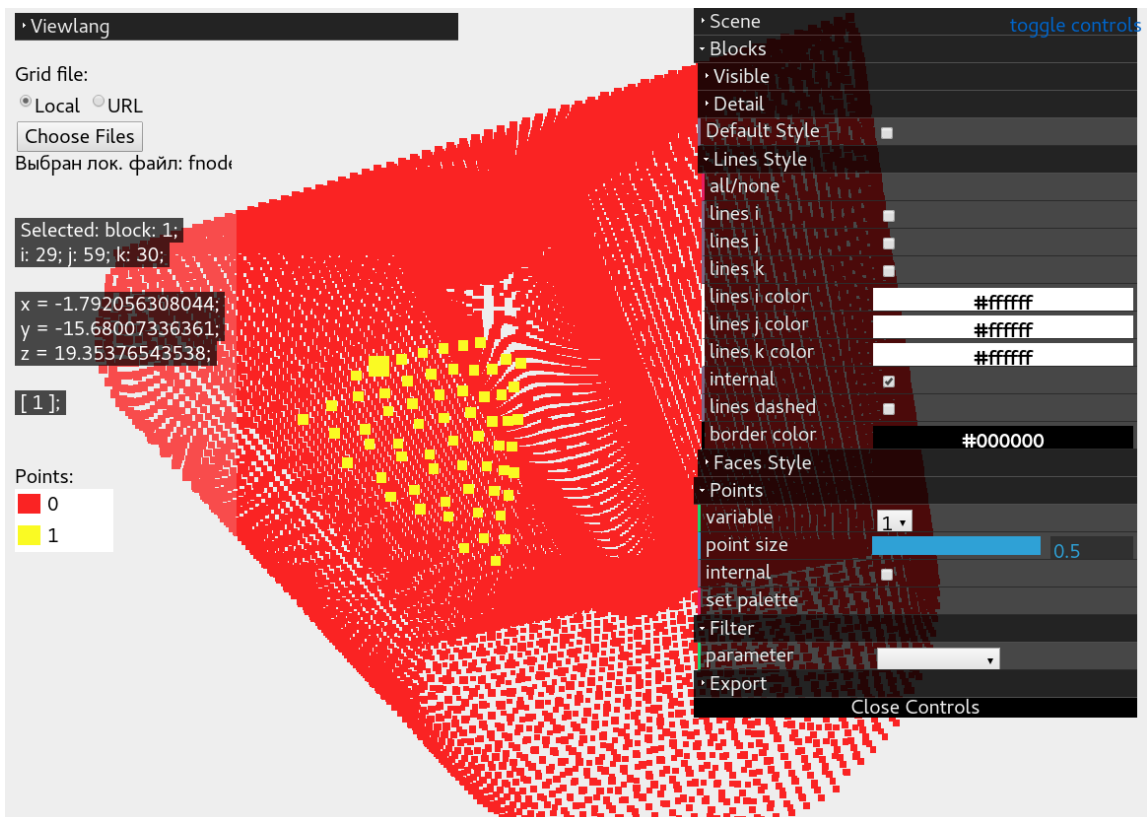
Blocks -> Faces Style -> internal faces.

При большом количестве элементов требуется разреживание.



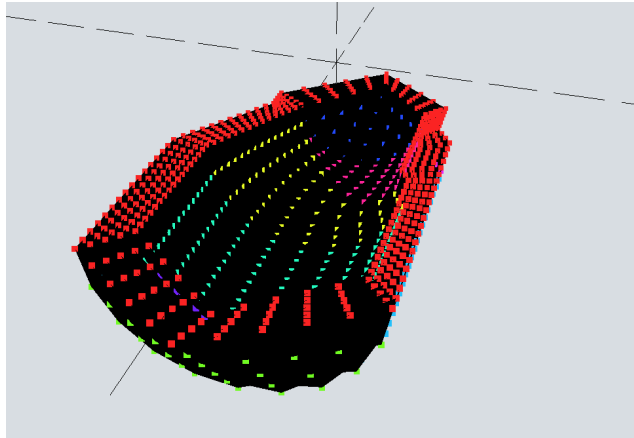
5. Узлы

Для отображения точек выбрать номер скаляра в разделе **Points** и задать размер, опционально выбирается отображение внутренних точек блока. Просмотр информации по точке по двойному клику.

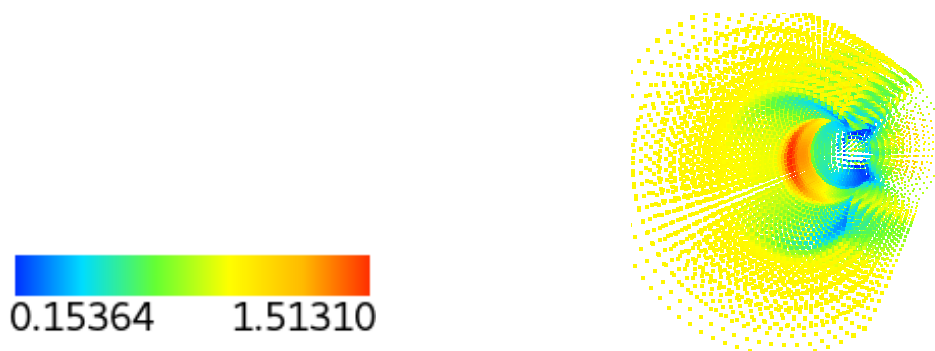


Видимые узлы:

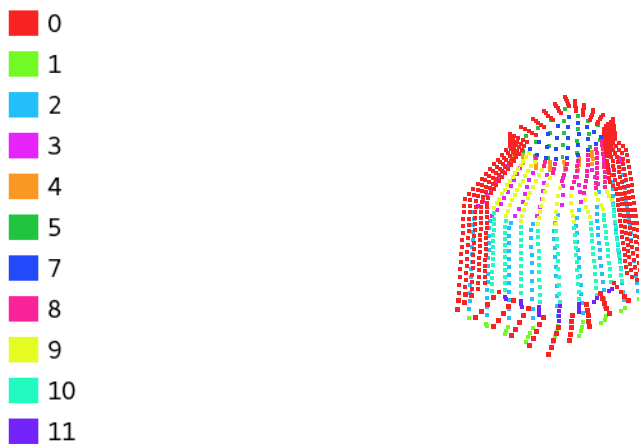
включить отображение непрозрачных внешних поверхностей.



Для вещественных значений скаляров узлы окрашиваются градиентной палитрой (в легенде выводятся минимальное и максимальное значения).



Для целочисленных значений скаляров формируется дискретная палитра.



Управление палитрой редактированием списка цветов:
Points -> set palette.

6. Фильтры

Возможно создание фильтра для выбора интересующей области для заданного блока. Для ячеек, удовлетворяющих фильтру, стили задаются отдельно аналогичным образом.

Фильтр по индексам

Указать номер блока, индексы первого, последнего элемента и шаг для каждого направления. (В скобках выводится размерность выбранного блока для направления.)

Переходы к соседним ячейкам: b, B, i, I, j, J, k, K.

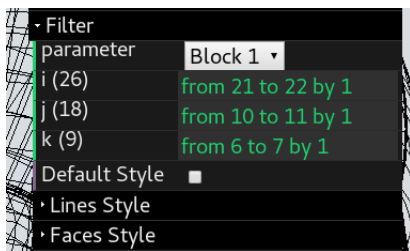


Рис. 24: Меню настройки фильтра

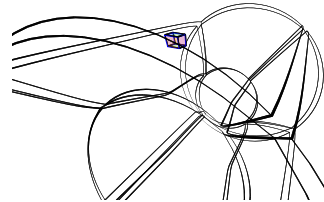


Рис. 25: Шестигранная ячейка

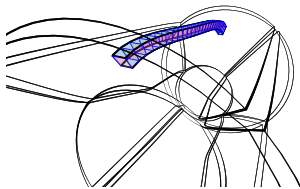


Рис. 26: Ряд ячеек
Все индексы (от 1 до размерности) для одного направления.

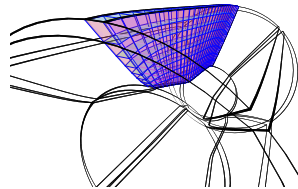


Рис. 27: Слой ячеек
Все индексы для двух направлений.

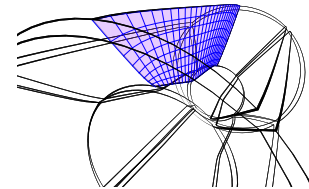


Рис. 28: Поверхность
Совпадающие индексы в нужном направлении.

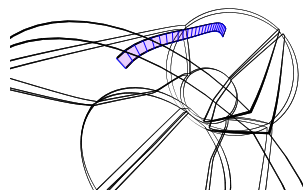


Рис. 29: Часть поверхности
Указать интересующие ограничения.



Рис. 30: Кривая
Совпадающие индексы для двух направлений.

Дополнение аргументов по-умолчанию:

- 0 аргументов - выбираются все индексы направления,
- 1 аргумент - выбирается ячейка,
- 2 аргумента - выбранный промежуток с единичным шагом.

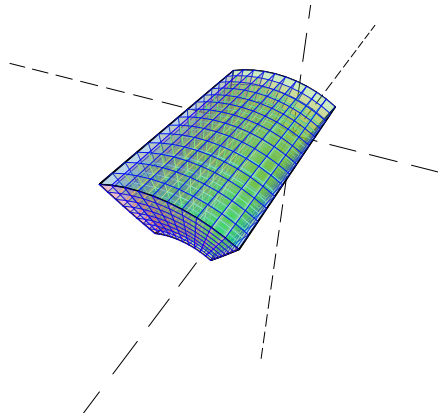


Рис. 31: Область, заданная фильтром:
отключить отображение элементов в основной части.

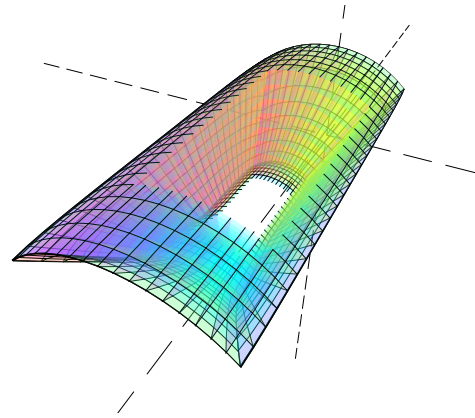
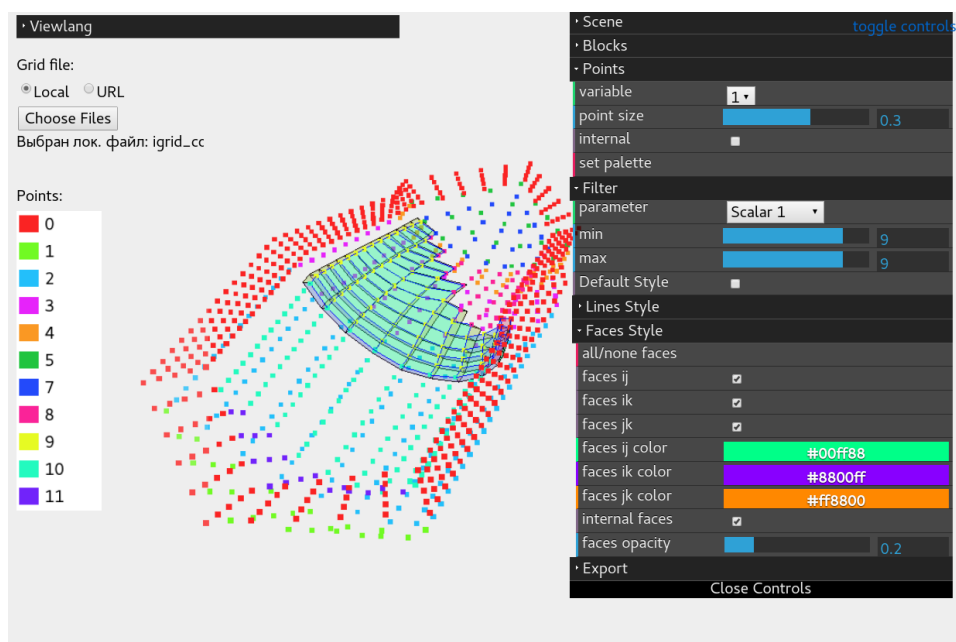


Рис. 32: Отверстие, заданное фильтром:
отключить отображение элементов в фильтре,
включить в основной части.

Фильтр по скалярным значениям

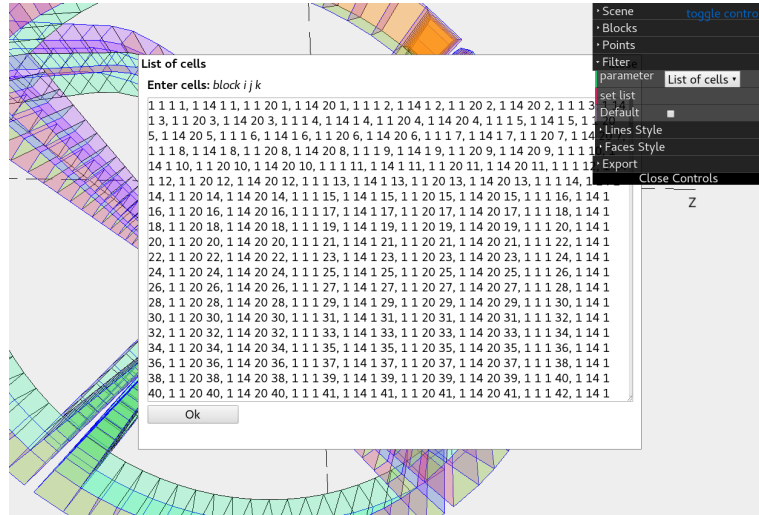
Выбор ячеек из заданного диапазона по скалярным значениям в узлах.

Указать в качестве параметра скаляр по номеру и выбрать минимальное и максимальные значения.



Фильтр по списку ячеек

Filter -> parameter List of cells, set list
(номер блока, индексы i, j, k)



Экспорт

Границами окна браузера можно выбрать область изображения.

Растровое изображение - *.png

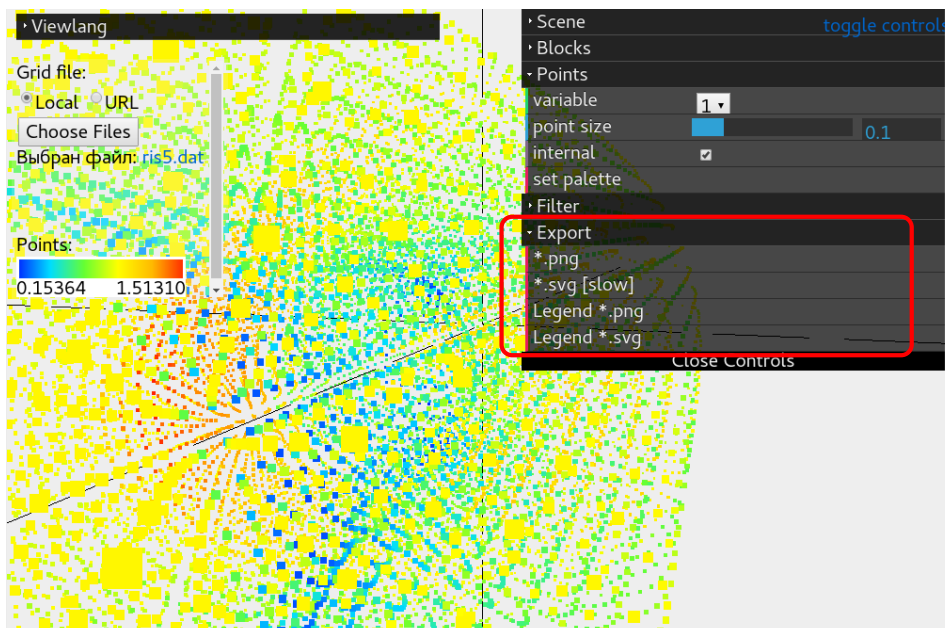
Векторное изображение - *.svg

Примечания по формату SVG:

Векторное представление может отличаться от исходного изображения.

Получаются файлы большого объема. При возникновении ошибок браузера можно попробовать уменьшить количество элементов разреживанием.

При выбранном отображении узлов доступен экспорт легенды.



Abstract

Annenkova O.G., The Specialized System of Visualization of Numerical Grid Data. Thesis for the degree of Master of Science, p. 42, pictures 19, bib.titles 10, appendix 1.

Keywords: SCIENTIFIC VISUALIZATION, SPECIALIZED VISUALIZATION SYSTEM, THREE-DIMENSIONAL STRUCTURED GRID, INTERACTIVE 3D-VISUALIZATION.

The object of study - three-dimensional structured grid used for the numerical solution of differential equations by finite difference method. The aim of the work - design and implementation of a specialized system for visual analysis quality of grid configuration, form of cells and overview of the internal structure of the grid. The result of the work is an interactive web application with three-dimensional graphics, applicable to research and solve practical problems of applied mathematics. The project offers the possibility of creating illustrative materials for scientific papers and reports and viewing graphics on a virtual reality device.