

# О Создании Методов Многомерной Визуализации

П.А. Васёв, Д.С. Перевалов

Институт Математики и Механики УрО РАН, Екатеринбург

## АННОТАЦИЯ

Работа посвящена теории и практике многомерной визуализации. Разработана классификация методов визуальных представлений, изложены принципы создания сложных визуальных систем. Большое внимание уделено проблемам и рекомендациям по взаимодействию разработчика системы визуализации и конечным пользователем системы. В работе содержатся примеры действующих систем, созданных авторами.

## 1. ВВЕДЕНИЕ

При исследованиях в естественных науках важную роль играет задача интерпретации данных. Визуализация является одним из мощных средств, позволяющих эффективно решать такую задачу. Однако в большинстве случаев данные - это не график или набор чисел, а сложный многомерный объект. Возникает проблема поиска подходящих способов его графического представления.

Этим вопросом занимается многомерная визуализация. Как дисциплина она начала формироваться в 50-х годах прошлого века вместе с появлением первых устройств графического вывода. К настоящему времени разработано много программных и алгоритмических средств визуализации многомерных структур. Прежде всего, к ним относятся универсальные системы визуализации Data Explorer, RockWorks99 и Slicer Dicer.

Однако в ряде задач не удаётся успешно применить обычную технику визуализации. Это связано с тем, что исследователя, как правило, интересуют специфические свойства объектов, которые не удаётся выявить с помощью стандартных подходов. В этом случае возникает потребность в разработке специальных видов представления, ориентированных на конкретную задачу.

В работе освещаются различные аспекты создания специализированных систем многомерной визуализации, а именно, теоретические основы, практические рекомендации и возможные трудности. Статья является подведением итогов успешной работы авторов по созданию нескольких систем визуализации [1, 2].

## 2. ТЕОРЕТИЧЕСКИЕ ОСНОВЫ

Исследователю, решающему задачу на основе данных реального или численного эксперимента, зачастую требуется интерпретировать значительные массивы числовых данных, чтобы получить лаконичный ответ о справедливости выдвинутой гипотезы, адекватности модели и т.п. Задача интерпретации легко решается, если известно, как и какую именно информацию необходимо извлечь.

Другое дело, если исследователю нужно проанализировать геометрическую структуру одного или нескольких множеств в  $n$ -мерном пространстве. Типичный пример - задача кластерного анализа, где в точечном множестве требуется обнаружить области сгущения, или

кластеры [6]. Для решения таких задач разработано много алгоритмов кластеризации. Но для их успешного применения и контроля правильности необходима априорная информация о форме и приблизительном числе кластеров. Это можно сделать, лишь получив визуальное представление входного множества.

Особенности человеческого сознания не позволяют визуально воспринимать геометрические объекты размерности больше трех. Это приводит к необходимости трансформации множеств в  $n$ -мерном пространстве в 1, 2 и 3-мерные визуальные представления. Такие способы преобразования называются методами визуализации, или видами отображения [3].

### 2.1. Проблема искажения

При интерпретации данных исследователей интересуют самые разнообразные свойства объектов. Но пространства разных размерностей топологически неэквивалентны, а значит любой вид отображения искажает геометрию многомерного множества - либо теряется часть информации, либо не выполняется свойство непрерывности представления. Это приводит к тому, что не удаётся построить методы, достаточные для всех задач визуализации. Приходится разрабатывать специализированные способы графического представления, выделяющие интересующую информацию. Кроме того, оказывается целесообразным использовать несколько методов, объединённых в систему визуализации.

Таким образом, многомерная визуализация занимается изучением и созданием методов представлений многомерных структур и вопросами объединения этих методов в системы визуализации, позволяющие изучить многомерный объект "с разных сторон".

В дальнейшем предполагается, что объектом визуализации является конечное множество точек  $M$ , лежащее в  $n$ -мерном евклидовом пространстве. Этот случай часто имеет место при анализе экспериментов, а изучение непрерывных структур - многообразий и полиэдров - сводится к анализу их дискретизированных моделей.

### 2.2. Классификация методов многомерной визуализации

В этом пункте описываются основные приёмы, применяемые в многомерной визуализации. Одновременно приводится их классификация, весьма полезная при проектировании сложных визуальных систем.

При построении визуального представления  $M$  можно преследовать две цели - изучить геометрию множества или представить в визуальной форме его свойства. В связи с этим естественно разделить методы на *геометрические* и *структурные*.

К геометрическим представлениям относятся фольклорные методы сечений и проекций. В них

результатом визуализации является двумерная фигура или трёхмерное тело, передающее геометрию объекта в двух или трех выбранных измерениях. В качестве примера на рис. 1 (все рисунки, упоминаемые в тексте, находятся в приложении) изображена проекция пятимерного куба.

Структурные методы включают в себя различные графики и диаграммы, передающие распределение диаметров и мощности сечений, расположение их центров и т. п. [2]. К ним же относится и метод скалярных гистограмм, характеризующий распределение расстояний между парами точек [1], см. рис. 2.

Выше отмечалось, что 2 или 3-мерный объект-образ, полученный в результате трансформации исходного множества  $M$ , не сохраняет его метрические свойства. Поэтому геометрические представления разделяются по способу искажения: *непрерывные*, передающие лишь часть информации и *разрывные*, позволяющие однозначно восстановить объект по его образу. Как правило, разрывные методы заключаются в одновременной генерации всех сечений и расположении их в 3-мерном пространстве.

Возможности визуализации позволяют оперировать не с чисто геометрическими объектами, а наделять их цветом и текстурой. Поэтому при использовании геометрических методов можно передавать информацию с помощью свойств поверхности. В особенности это касается методов типа проекций - на проекцию множества с помощью цвета и текстуры передаётся информация о прообразе данной точки проекции [2,5] (см. рис. 3). Таким образом геометрические методы классифицируются по тому, сколько компонент  $n$ -мерных векторов они передают *формой*, а сколько *свойствами поверхности*.

Отметим, что геометрические подходы целесообразно использовать на начальном этапе анализа объекта, чтобы понять его строение и найти интересующие свойства, которые затем следует представить "структурно".

### 2.3. Системы визуализации

Мощность визуализации значительно возрастает, если различные методы визуализации объединены в единый комплекс - систему визуализации. Основное качество таких систем - наличие обратной связи с пользователем, что даёт возможность в прямом смысле слова осуществлять визуальный анализ объекта.

Полноценная система обычно обладает следующими качествами:

*Наличие нескольких взаимосвязанных видов отображения.* Часто одно представление является главным и на нём изображена 2- или 3-мерная проекция множества. Другие виды отображений специализированные. Часть из них показывает структурные описания множества, а другая часть предназначена для локального исследования окрестностей точек, выбираемых пользователем на главной проекции.

*Динамика представлений.* У пользователя имеется возможность изменять параметры представлений: двигать и вращать плоскости сечения и 3-мерные тела на экране,

менять используемые цвета и текстуры, размеры изображаемых объектов.

*Сохранение однородности данных.* Представления должны настраиваться так, чтобы компоненты векторов, несущие похожую смысловую нагрузку, представлялись одинаково. К примеру, если три компоненты вектора из  $M$  отвечают за координаты  $p=(x, y, z)$  точки в трёхмерном пространстве, а две другие компоненты описывают массу и скорость  $q=(m, |v|)$ , то каждый из наборов координат  $p$  и  $q$  желательно отображать либо формой, либо свойствами поверхности. Это связано с тем, что человек не может точно сравнить числовую координату и цвет точки.

Другой пример. График функции комплексного переменного можно трактовать как подмножество 4-мерного евклидова пространства. Тогда, для выполнения требования однородности, представление этого множества должно передавать аргумент функции - координатами точки на плоскости, а значение функции - цветом этой точки, см. рис. 4.

Поясним на нескольких примерах конкретных систем изложенные идеи. Для иллюстраций используется искусственно построенное четырехмерное множество  $M = \{(x, y, z, w)\}$ , напоминающее по устройству многомерный кластер. Его проекция на  $(x, y)$  есть квадрат с закруглёнными краями. Прообраз точки  $(x, y)$  в  $(z, w)$  состоит из нескольких точек, случайным образом расположенных в прямоугольнике  $[-x, x] \times [-y, y]$ .

*Система общего назначения.* Основное представление - таблица двумерных проекций множества на все пары координатных осей. Пользователь выделяет на одной из проекций интересующую его область и анализирует её как отдельное множество, причём дополнительные представления также являются таблицами проекций. На рис. 5 показана таблица проекций  $M$ , выделенная область на  $(y, z)$ -проекции, и таблица для подмножества  $M$ , соответствующего выделенным точкам. В каждой проекции цвет точки передаёт число точек прообраза. Система работает с множествами произвольной размерности, и мы применяем её для первоначального изучения множеств. Она даёт непрерывные однородные представления, основанные на передаче формы. Цветом передаются не компоненты векторов, а структурная информация.

*Система для анализа 4-мерных кластеров.*

Используется одно представление - так называемая скользящая проекция. Это однородное представление с одним управляющим параметром. Сначала четыре компоненты векторов разбиваются на пары  $(x, y)$  и  $(z, w)$ . Затем плоский квадрат делится на  $m \times m$  малых квадратов. Каждая точка  $a = (x, y, z, w)$  из  $M$  располагается в квадрате так:  $(x, y)$  отвечают за выбор малого квадрата, а  $(z, w)$  отвечает за расположение точки в малом квадрате. Пользователь меняет параметр  $m$ . Когда  $m=1$ , получаемое представление есть  $(z, w)$ -проекция  $M$ , а когда  $m$  таково, что размер малого квадрата равен 1 пикселу, то это  $(x, y)$ -проекция  $M$ . Промежуточные значения  $m$  показывают плавный переход между двумя проекциями. Система полезна при изучении информативности признаков при кластерном анализе, а также для выяснения строения множеств. Дело в том, что при промежуточных  $m$

представление напоминает разрывный метод сечений с возможностью настройки степени разрывности. Это позволяет довольно точно оценить строение множества  $M$ . На рис. 6 две правых картинка показывают, что множество состоит из случайных точек, причём точки в сечении  $(x, y)$  лежат в прямоугольниках, задаваемых значениями  $x, y$ .

*Система для анализа одного класса 4-мерных множеств* [2]. Эта система разрабатывалась для изучения множеств, возникающих при моделировании химических реакций. Множество строится на четырехмерной сетке, являющейся произведением двух двумерных сеток коэффициентов. Поэтому оказалось удачным сделать упор на разделении пар координат.

Для лучшей интерпретации данных вводится специальный объект, так называемый росток точки, связанный с представлением двумерных сечений множества, где секущая гиперплоскость задается с помощью выбора точек из двумерной проекции. Пользователь в интерактивном режиме выбирает одну точку двумерной проекции и наблюдает строение ее ростка, тем самым получая возможность визуально оценить его конфигурацию. Перемещая выделенную точку, исследователь в динамике видит изменение строения ростка при изменении координат. Для преодоления ограничений двумерных представлений используется трехмерные проекции. Вводится понятие трехмерного слоя, то есть набора ростков точек проекции, оказавшихся на данной прямой. Изобразив слой как трехмерное тело, можно получить визуальную оценку совокупности ростков. Представления свойств ростков на проекции осуществляется как за счет использования цвета, так и за счет передачи параметра ростка с помощью трехмерного графика над проекцией множества. Высота графика над данной точкой задается значением изучаемого параметра.

Таким образом, основной вид отображения - двумерная проекция. Вспомогательными видами отображений для изучения локального строения являются представления ростка и набора ростков, а структурные виды отображения - все упомянутые представления свойств ростков на проекции.

Система оказалась мощным средством для работы с информационными множествами. Общий вид системы показан на рис. 7.

Мы указали приёмы многомерной визуализации и примеры их использования в реальных системах. Но при создании системы визуализации важнейшим условием успеха является взаимодействие разработчика и конечного пользователя. Возникающим при этом трудностям и способам их преодоления посвящён следующий параграф.

### 3. ПРАКТИЧЕСКИЕ РЕКОМЕНДАЦИИ

Многомерная визуализация пока остаётся "экзотическим" направлением научной визуализации. Задачи, решаемые в ней, нетипичны ни для математиков, ни для программистов. Опыт создания СМВ в мире как правило является "точечным" - проблема визуализации возникает перед исследователем, воспринимается им как часть его исследований, каким-либо образом решается (или не решается) самостоятельно и, не представляя для него дальнейшего интереса, забывается.

Накопление и обобщение опыта решения таких задач является необходимым для более эффективного решения вновь возникающих проблем.

Подчеркнём, что создание метода неразрывно связано с его реализацией. С одной стороны, разработка системы визуализации данного класса множеств включает в себя разработку (выбор) методов визуализации. С другой, создание метода подразумевает его *удачную* реализацию. Действительно, мощная система бесполезна, если заложенные в ней идеи не позволяют исследователю интерпретировать интересующий его объект. Эффективный метод не найдёт своего применения, если его реализация будет некачественной. Поэтому построение метода и его реализация представляют собой единый процесс.

### 3.1. Типичные проблемы

Рассмотрим некоторые специфические проблемы, возникающие при разработке систем многомерной визуализации.

#### Проблема 1: Пользователь не может сформулировать своих требований

Часто пользователь не может ясно объяснить, какие именно свойства объектов его интересуют. Такие ситуации типичны для пользователей, которые ранее не сталкивались с проблемой многомерной визуализации. Они просто хотят "видеть" множество - и уже после "видения" понимают, что конкретно их интересует. Причиной здесь является то, что исследователь не может представить себе, как должен выглядеть его объект и какие средства визуализации можно задействовать для получения нужных представлений.

#### Проблема 2: Разработчик не знает, с чего начать разработку системы

Постановка задачи как правило даётся в общем виде - "получить информативное представление для данного класса множеств". Выбор и создание методов визуализации - ответственность разработчика. Если разработчик не обладает достаточным опытом, то создание визуальной системы может оказаться для него на первый взгляд невыполнимой задачей. Важно понимать, что такая работа - процесс творческий, и изначально отсутствие очевидного решения не означает, что такого решения не существует.

#### Проблема 3: Реализация системы не удовлетворяет пользователя

Одобрённый и реализованный метод может не устраивать пользователя, в то время как на его реализацию было потрачено много ресурсов. Причины такой ситуации заключаются в следующем:

- выбран неподходящий метод: особенно высока вероятность такой ошибки на ранних стадиях разработки;
- метод не полностью соответствует исследовательской задаче и нуждается в модификации;
- получаемые визуальные представления невыразительны;
- интерфейс системы не проработан.

Отметим, что неопытный пользователь (то есть не пользовавшийся ранее возможностями многомерной визуализации) может формально и принять систему, но фактически не будет её использовать. Отказ от дальнейших

попыток поиска подходящего метода зачастую мотивируется априорным отношением к задачам многомерной визуализации как к практически нерешаемым.

Описанные проблемы, особенно проблема 1, часто встречаются на практике и способны привести разработку системы в тупиковое состояние. Особенно сложной является ситуация, когда и пользователь не может четко выразить своих требований, и разработчик не знает, как следует действовать и как вести линию взаимодействия с пользователем. Однако практика показывает, что все указанные проблемы – решаемы.

Далее описаны условия и рекомендации, обобщающие концепцию разработки систем многомерной визуализации и призванные сделать путь их разработки более эффективным и ясным.

## 3.2. Необходимые условия

Рассмотрим условия, необходимые для успешной разработки методов и систем многомерной визуализации.

### 1. Понимание сути задачи пользователя.

Необходимо знать, откуда появляются множества, требующие визуализации, какой смысл несут величины и отношения, описанные в этих множествах. Так разработчик сможет взглянуть на проблему глазами пользователя и лучше понять цели, стоящие перед ним.

### 2. Понимание структуры и специфики множеств.

Информация об устройстве множества и о том, как оно породилось, является ключевой - в основном она и определит структуру искомого метода.

### 3. Понимание того, какие свойства множеств интересуют пользователя.

Разрабатываемый метод должен ориентироваться на отражение тех и только тех свойств исходных объектов, которые важны для пользователя. К сожалению, иногда разработчики упускают из вида главные свойства объектов, представляющие для исследователя основной интерес, но концентрируются на второстепенных и часто ненужных деталях.

### 4. Учёт требований пользователя к визуализации и его ожиданий.

Качество графики и интерфейса создаваемой системы должно соответствовать ожиданиям пользователя. Однако и введение в систему дополнительных возможностей, эффективных по мнению разработчика, может обернуться лишь тратой ресурсов.

### 5. Постоянное взаимодействие с пользователем на всех этапах разработки методов и системы.

Между разработчиком и пользователем должен установиться плотный информационный контакт. С самого начала необходимо узнать как можно больше о задаче пользователя, структуре требующего визуализации объекта и способе его порождения. С другой стороны, необходимо ознакомить пользователя с возможностями многомерной визуализации с помощью демонстрационных примеров и действующих систем. Пользователь должен знать о различных средствах, методах и подходах визуализации для того, чтобы адекватно строить свои ожидания и самому непосредственно участвовать в разработке.

Особо важен оперативный обмен информацией и идеями в процессе работы – уточнение существенности тех или иных свойств объектов, определение важности не упоминавшихся ранее свойств, выяснение целесообразности применения различных подходов.

Отметим, что когда испытателем (тестером) системы является её будущий пользователь, эффективность тестирования и планирования значительно повышается.

## 3.3. Дополнительные рекомендации

Следующие рекомендации позволяют существенно уменьшить время разработки и повысить качество создаваемых систем.

**1. Параллельная разработка нескольких эскизных методов.** На ранних этапах разработки не может быть однозначного ответа на вопрос о структуре искомого метода. Зачастую полезным оказывается вести поиск сразу в нескольких направлениях. Из набора различных методов пользователь сможет выбрать наиболее перспективные и лучше понять, какие подходы его устраивают.

**2. Повышенная эргономичность пользовательского интерфейса.** Действительно, работа в системах многомерной визуализации требует большой концентрации. Нельзя допустить, чтобы пользователь тратил значительную часть своего внимания на управление системой - напротив, управление должно осуществляться на уровне автоматизма. Подчеркнём, что от его удобства во многом зависит успех системы в целом. Основные принципы создания интерфейсов можно найти в [7] и [8].

**3. Максимальное использование возможностей человеческого восприятия.** Часто искомые представления оказываются информационно перегруженными. Поэтому целесообразно строить системы с привлечением самых разнообразных подходов визуализации – от цветопередачи и анимации до средств виртуальной реальности и нетрадиционных решений в области человеко-машинного взаимодействия.

**4. Привлечение новых метафор.** Многомерная визуализация служит для порождения когнитивных представлений. Получаемые представления зачастую несут уникальную смысловую нагрузку, которая задаётся системой понятий научной области, породившей изучаемый объект, и способом его трансформации при визуализации. Для обеспечения лучшего понимания отображаемых сущностей полезным и эффективным оказывается привлечение новых метафор.

## 4. ЗАКЛЮЧЕНИЕ

Многомерная визуализация является благодатной областью для исследований и практических разработок. Она обеспечивает потребности многих отраслей современной науки. Задач, связанных с визуализацией, становится всё больше. Растёт и число методов, решающих такие задачи. Можно надеяться, что накопление большого арсенала методов позволит создать программные системы, решающие широкий класс задач визуализации, аналогично тому, как системы, подобные MatLab, успешно решают задачи вычислительной математики.

Авторы выражают благодарность В.Л. Авербуху за всестороннюю поддержку их работы, а также А.Г. Иванову, С. В. Скорбу и другим сотрудникам ИММ УрО РАН за множество ценных предложений и проявленное терпение.

Мы надеемся, что приведенная концепция будет полезной многим разработчикам и теоретикам визуализации.

## 5. ПРИЛОЖЕНИЕ

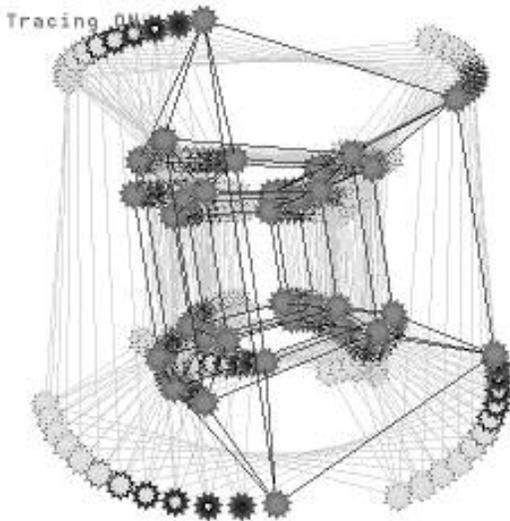


Рис. 1: Метод проекций и анимация. Показан след вращения 5-куба.

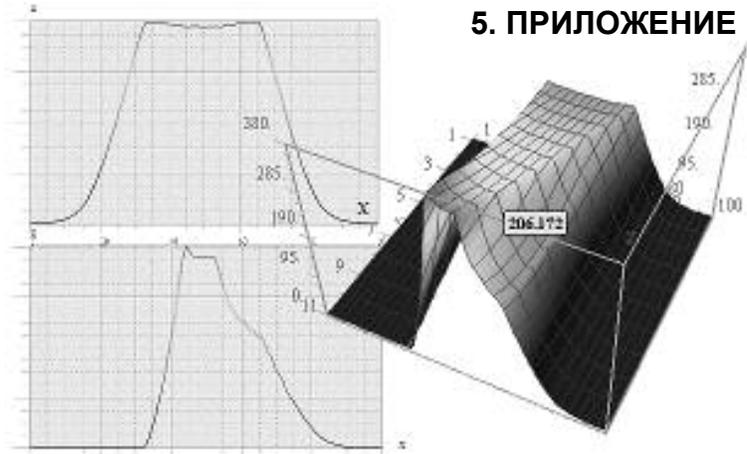


Рис. 2: Метод скалярных гистограмм. Слева сверху – гистограмма куба, внизу – пирамиды. Справа изображена гистограмма четырехмерного полиэдра «пирамида-куб».



Рис. 3: Представление 4-множества, полученное с помощью раскраски. Цвет задаётся четвёртой компонентой точек множества.

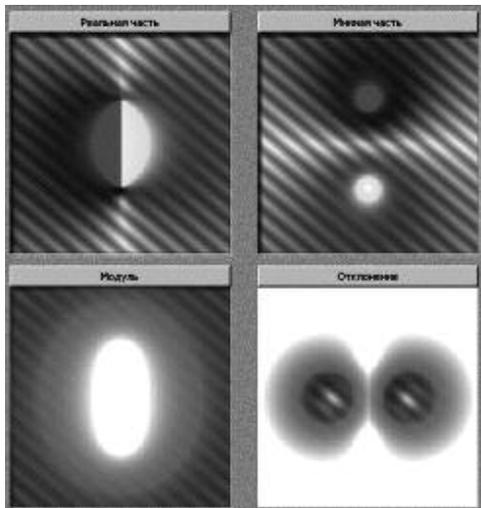
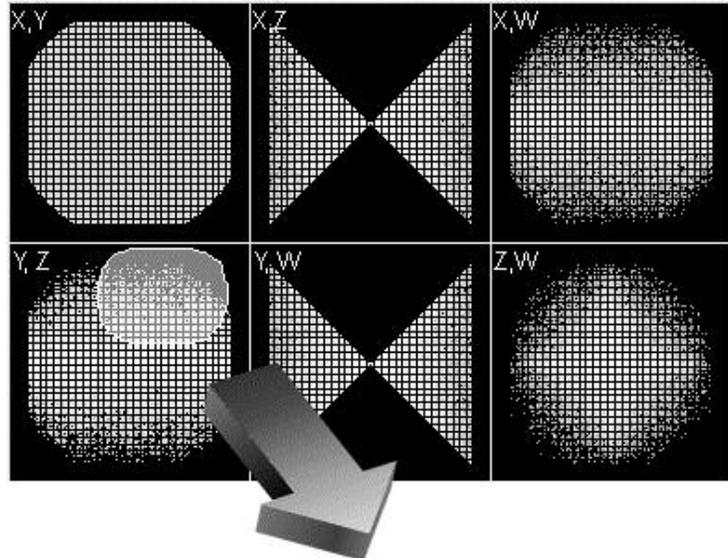


Рис. 4: Представление функций комплексного переменного. Изображена функция  $\text{arctg}(z)$ . Белый цвет соответствует положительным значениям, тёмно-серый – отрицательным. Близкие к нулю значения обозначены штрихом.

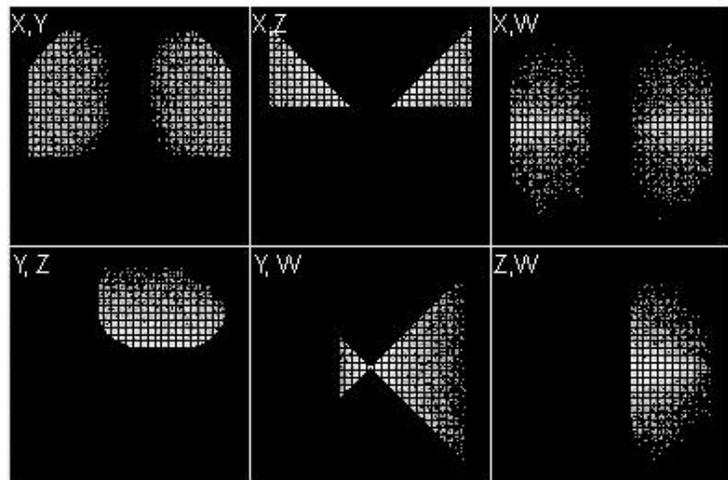


Рис. 5: Метод проекций для 4-множеств. Сверху показаны все проекции множества. Выделена область на (Y,Z)-проекции; внизу изображены все проекции её прообраза.

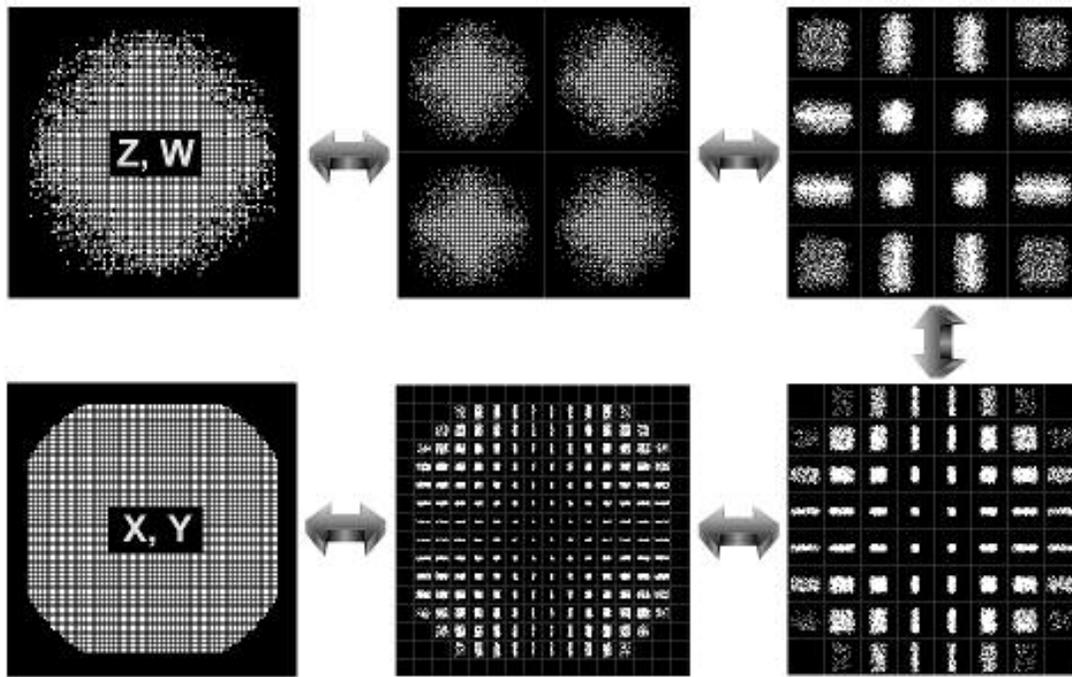


Рис. 6: Метод скользящей проекции.

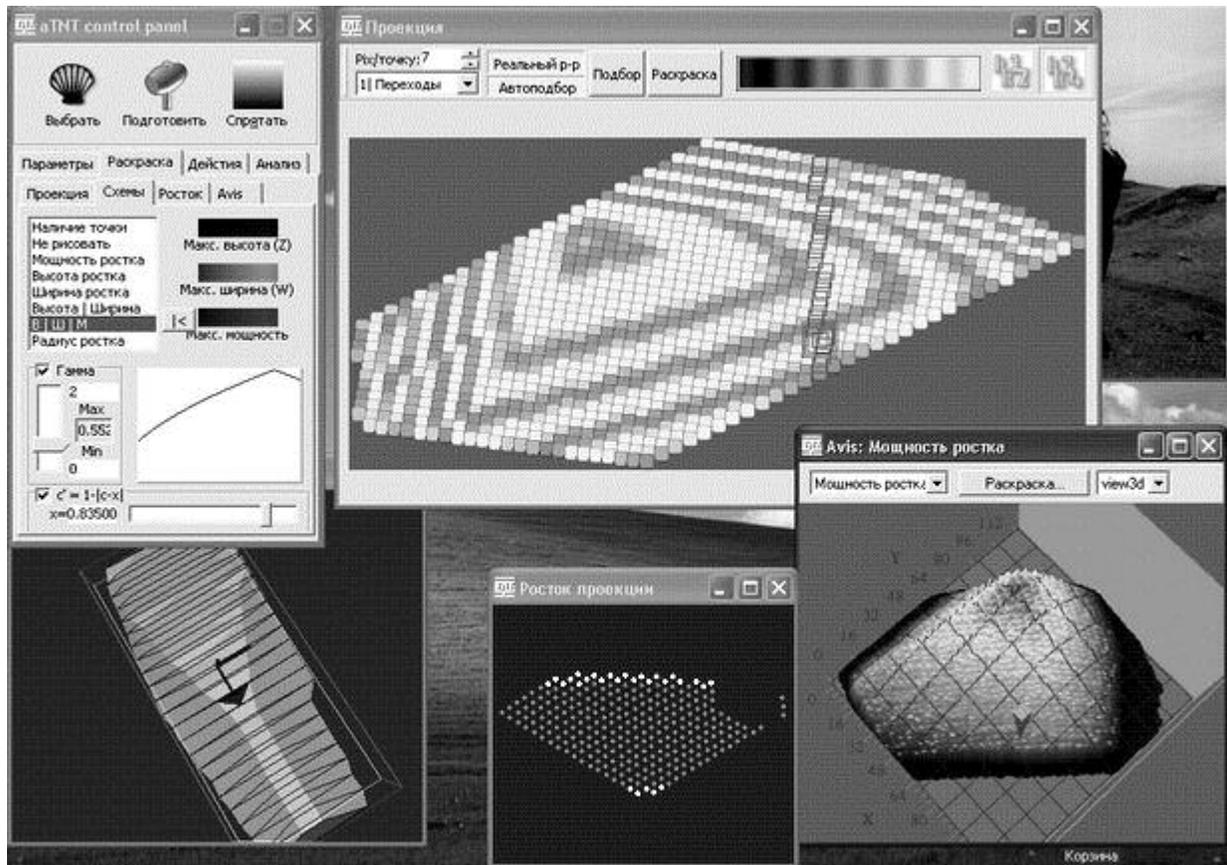


Рис. 7: Общий вид системы tnt.

## 6. ЛИТЕРАТУРА

1. Васёв П.А., Перевалов Д.С., Некоторые методы многомерной визуализации, в сб. "Проблемы теоретической и прикладной математики", Екатеринбург: ИММ УрО РАН, 2001г, с.275-279
2. Васёв П.А., Перевалов Д.С., Один метод визуализации 4-мерных множеств, в сб. "Суперкомпьютерные вычислительно-информационные технологии в физических и химических исследованиях", Черногловка: ИПХФ РАН, 2001г., с.32-37.
3. Авербух В.Л., Зенков А.И., Исмагилов Т.Р. и др.. Разработка специализированных систем научной визуализации, в сб. "Алгоритмы и средства параллельных вычислений", вып. 4, Екатеринбург: ИММ УрО РАН, 2000 г., с.3-23.
4. Зиновьев А.Ю., Визуализация многомерных данных, 2000 г. <http://www.ihes.fr/~zinovyev/papers/book/>
5. Healey C.G., Enns J.T. Building Perceptual Textures to Visualize Multidimensional Datasets, The University of British Columbia, IEEE, 1998.
6. Дюран Б., Оделл П., Кластерный анализ // Пер. с франц. М.: Статистика, 1977.
7. Macintosh Human Interface Guidelines, Apple Computer, Inc, 1996, <http://developer.apple.com/techpubs/mac/HIGuidelines/HIGuidelines-2.html>
8. Bruce Tognazzini, First Principles, <http://www.asktog.com/basics/firstPrinciples.html>