Описание программного комплекса для моделирования кубантов.

Предисловие.

Для того, чтобы моделировать кубанты или кубические комплексы (комплексы кубантов) было создано специальное программное обеспечение. Оно позволяет, с одной стороны производить вычисления и получать некоторые результаты над кубантами, и, с другой стороны — визуализировать полученные результаты.

Для того, чтобы эта многофункциональность было доступна, была разработана архитектура, состоящая из трёх практически независимых частей:

1.«Ядро»
2.«Интерпретатор»
3.«Визуализатор»

Опишу каждую из этих частей:

Ядро.

Ядро — это основная часть программы, которая предоставляет возможность создавать кубанты, производить с ними различные операции, получать результат. Ядро системы написано на языку программирования C++, для того чтобы обеспечить кроссплатформенность. За счёт этого, все программы, написанные с использованием библиотеки-ядра могут работать как и на обычных компьютерах, так и на суперкомпьютерах.

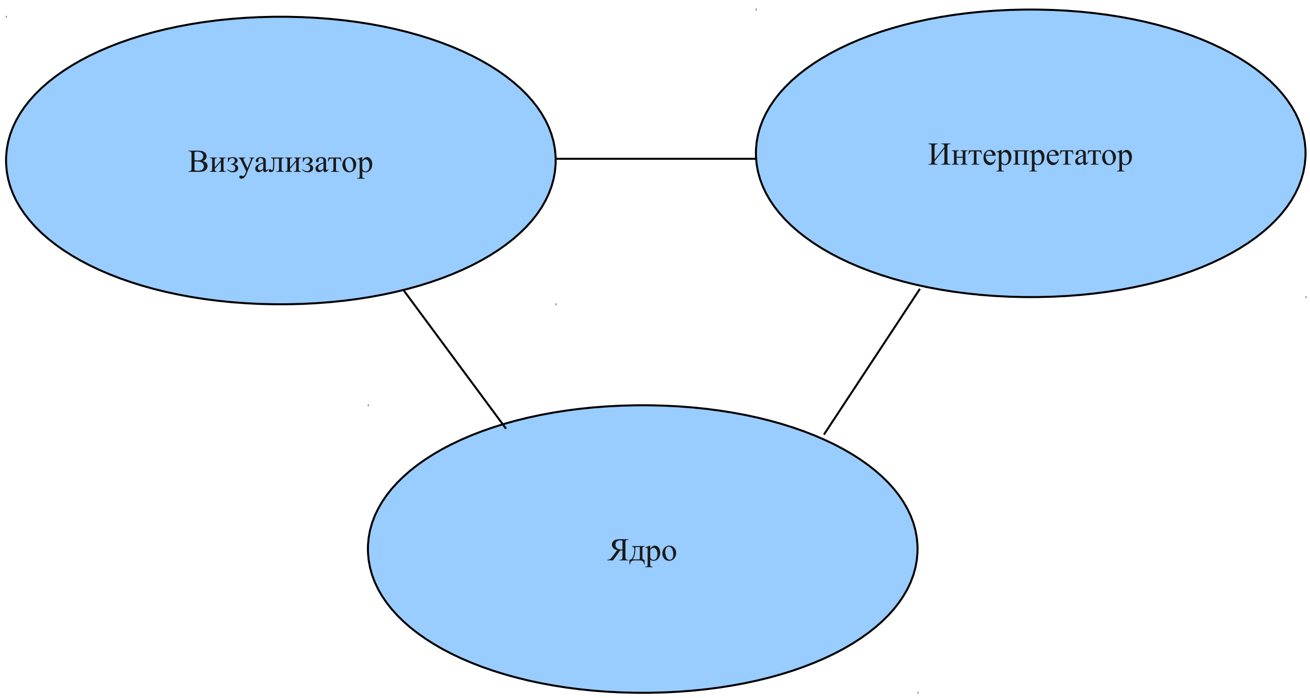
Интерпретатор.

Для того, чтобы произвести простой расчёт или просто произвести вычисления, не требующие больших вычислительных мощностей, можно использовать интерпретатор. Его использование позволяет быстро написать программу, не компилируя её. В качестве интерпретируемого языка в системе моделирования кубантов используется JavaScript. Интерпретатор может использоваться для предварительных расчётов, для поиска вариантов решения. После того, как подходящий алгоритм найден, мы можем реализовать его, используя непосредственно ядро, и прогонять его как на обычном настольном компьютере, так и на кластере.

Визуализатор.

Чтобы более полно представлять картину происходящего, в систему был добавлен визуализатор. Он используется для непосредственного отображения кубантов. С помощью визуализатора мы можем отобразить репер, кубант или несколько кубантов, задать различные цвет для кадого кубанта.

Все три компонента этой системы являются легко заменяемыми на другие, с аналогичным интерфейсом. За счёт этого достигнута гибкость — интерпретатор не обязательно должен быть интерпретатором языка Javascript, а может быть интерпретатором любого скриптового языка, например Lua или ActionScript. Практически это свойство заменяемости использовалось для построения двух различных визуализаторов: один 2D, а другой 3D, между которыми мы можем свободно переключаться. Так, рисуя кубант или кубический комплекс, мы получаем два (в текущей реализации) варианта его представления. Первый — это двухмерный вариант с плоским репером (для плоского варианта в текущей реализации репер мы можем задавать самостоятельно) и трёхмерный вариант, который отображает кубант, используя конусообразный репер.



Текущая схема Ядро — Интерпретатор — Визуализатор является реализацией известного паттерна проектирования MVC — Model — Controller — View , где модель — это ядро, контроллер — это интепретатор, а View — это визуализатор.

Поддерживаемые операции.

Так как система моделирования кубантов является хорошо расширяемой системой, то количество операций в Ядре постоянно увеличивается. Поэтому я опишу некоторые из них, которые уже давно реализованы и стабильны.

Операция «Оболочка кубантов из гиперграней»

Данная операция по одному кубанту размерности больше 0 (т. е. не точке) создаёт множество кубантов, являющимися его гипергранями. То есть, для кубанта /2,0,1/ результат выполнения этой операции будет множество кубантов (/0,0,1/, /1,0,1/). С точки зрения языка Javascript (компонент интерпретатор) эта операция выглядит так:

```
var cubant=createCubant("/2,0,1/");
System.print(cubantFacet(cubant));
```

В первой строке создаётся кубант /2,0,1/, и сохраняется в переменной cubant. Во второй строке к переменной cubant применяется операция «Оболочка кубантов из гиперграней», и и сразу после выполнения этой операции выводится результат. Полученный результат — это Javascript-массив:

```
[ "/0,0,1/" ,"/1,0,1/"]
```

Таким образом с помощью интерпретатора мы можем достаточно быстро необходимую нам информацию.

Операция «Хэммингово расстояние между кубантами»

В связи с тем, что мы рассматриваем решетчатые структуры, то и подсчёт расстояния между объектами в них может быть выполнен по-разному. Сейчас мы рассмотрим расстояние, вычисляемое на основе манхетенновской метрики. Мы будем считать расстояние между кубантами (только кубантами, а не кубическими комплексами) по рёбрам решетки. То есть, фактически хэммингово расстояние в этом случае — это минимальное расстояние между какими -либо частями кубантов.

Предположим, что у нас имеется 2 кубанта в 9-мерном пространстве. Пусть они будут заданы кодами /1,2,1,0,0,1,2,2,1/ и /0,1,1,1,1,2,2,2,2/ . Выпишем их в виде таблицы и посчитаем хэмминогово расстояние.

1	2	1	0	0	1	2	2	1
0	1	1	1	1	2	2	2	2
∅	1	1	∅	∅	1	2	2	1

В первых двух строках данной таблицы описаны исходные кубанты, а в третьей строке — результат выполнения операции «хэммингово расстояние». Как видно, оно вычисляется с помощью простых поразрядных операций. (Это свойство можно использовать для представления кубанта в виде некоторого кода в памяти вычислительной машины, для увеличения быстродействия и экономии оперативной памяти.)

В результате операция нахождения хэммингова расстояния зависит линейно (O(n)) от размерности пространства. Разумеется, здесь говориться об нахождении расстояния в рамках единичного куба.

Заметим, что в получившемся результате нахождения расстояния (мы будем называть эту операцию операцией умножения) информации больше, чем просто число — символов ∅. В результате этой операции мы получаем кубант (или псевдокубант, если количество в его записи есть хотя бы один символ ∅).

Если символов ∅ в записи кубанта — результата операции умножение нет, то получившийся кубант — это просто пересечение данных кубантов.

Таким образом, операция умножения носит комбинаторный характер.

В реализации интерпретатора эта операция представлена методом `intersect()`.

Пример её применения:

```
var cubant1=createCubant("/2,0,1/");
var cubant2=createCubant("/2,1,1/");
var result=intersect(cubant1, cubant2);
System.print(cubant);
```

На выходе их консоли мы получим результат — кубант со значением `"/2,Z,1/",` что эквивалентно /2,∅,1/. Таким образом, получается, что хэммингово расстояние расстояние между двумя данными кубантами — 1.

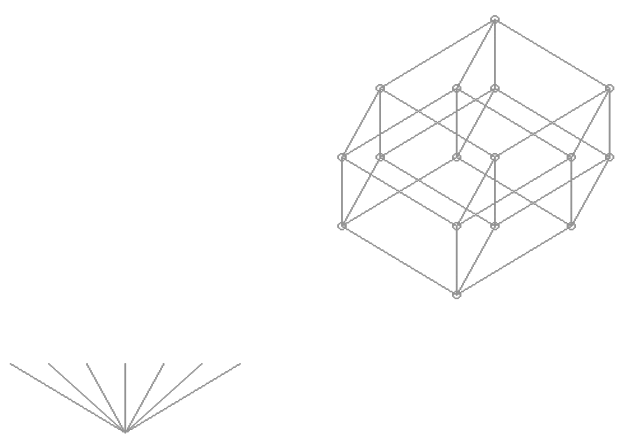
Однако, рассматривая внутри единичного N-мерного куба не только простые кубанты, но и кубические комплексы понятно, что они могут быть и не выпуклыми. Наиболее естественной метрикой в данном случае будет являться хаусдорфова метрика

Предположим у нас имеется два кубических комплекса в пространстве R3: комплекс A={/0,0,0/,/2,0,0/}, и комплекс B={/1,1,1/,/0,0,2/}. Найдём расстояние (в в смысле расстояния хаусдорфа между двумя этими комплексами)

// Здесь описание алгоритма.

Графическое представление.

Для представления кубантов была разработана специльная утилита.



4-мерный кубант в 7-мерном пространстве.

// Еще пара рисунков.

Использование интерпретируемого языка Javascript.