



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н. Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н. Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

К КУРСОВОЙ РАБОТЕ

НА ТЕМУ:
«Генератор трехмерного ландшафта»

Студент ИУ7-53Б
(Группа)

(Подпись, дата)

Лысцев Н. Д.
(И. О. Фамилия)

Руководитель курсовой работы

(Подпись, дата)

Филлипов М. В.
(И. О. Фамилия)

2023 г.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 Аналитический раздел	4
1.1 Формализация объектов синтезируемой сцены	4
1.2 Анализ и выбор формы задания трехмерных моделей	4
1.3 Анализ способов представления данных о ландшафте	5
1.3.1 Карта высот	5
1.3.2 Иррегулярная сетка	7
1.3.3 Посегментная карта высот	8
1.4 Анализ алгоритмов процедурной генерации ландшафта	10
1.4.1 Алгоритм Diamond-Square	10
1.4.2 Холмовой алгоритм	11
2 Конструкторский раздел	13
3 Технологический раздел	14
4 Исследовательский раздел	15
ЗАКЛЮЧЕНИЕ	16

ВВЕДЕНИЕ

В настоящее время технологии трехмерной графики стремительно развиваются. Одним из направлений применения этих технологий является создание видеоигр. Наибольшую популярность набирают игры с так называемым открытым миром.

Основой для открытого мира служит трехмерный ландшафт значительных размеров, который для реалистичности должен быть разнообразным и детализированным. Традиционные методы ручного моделирования ландшафта являются крайне громоздкими, трудоемкими и ограниченными в своей вариативности.

Возникает потребность в создании программного обеспечения, которое бы позволяло автоматизировать процессы создания реалистичного ландшафта, чтобы ускорить разработку игр и обеспечить большую степень креативной свободы для разработчиков. Целью работы является разработка программного обеспечения для генерации и визуализации трехмерного ландшафта.

Для достижения желаемых результатов необходимо решить следующие задачи:

- 1) Выполнить формализацию объектов синтезируемой сцены;
- 2) Провести анализ существующих алгоритмов создания ландшафта и визуализации сцены;
- 3) Выбрать подходящие алгоритмы для решения поставленной задачи.
- 4) Реализовать выбранные алгоритмы.

1 Аналитический раздел

В данном разделе будет дано описание алгоритмов визуализации ... (дописать)

1.1 Формализация объектов синтезируемой сцены

Сцена состоит из следующих объектов:

- Ландшафт – трехмерная модель, представляющая собой полигональную сетку, состоящую из связанных между собой плоских многоугольников;
- Источник света – материальная точка пространства, испускающая лучи света.

1.2 Анализ и выбор формы задания трехмерных моделей

В компьютерной графике для описания трехмерных объектов существует три вида моделей: каркасная, поверхностная и твердотельная. Использование моделей позволяет правильно отображать форму и размеры объектов сцены.

- Каркасная модель – в трехмерной графике описывает совокупность вершин и ребер, которая показывает форму многогранного объекта. Это моделирование самого низкого уровня и имеет ряд серьезных ограничений, большинство из которых возникает из-за недостатка информации о гранях, которые заключены между линиями, и невозможности выделить внутреннюю и внешнюю область изображения твердого объемного тела. Однако каркасная модель требует меньше памяти и вполне пригодна для решения задач, относящихся к простым. Основным недостатком каркасной модели является то, что модель не всегда однозначно передает информацию о форме объекта;
- Поверхностная модель. Поверхностное моделирование определяется в терминах точек, линий и поверхностей. При построении поверхностной модели

предполагается, что технические объекты ограничены поверхностями, которые отделяют их от окружающей среды. Недостатком поверхностной модели является отсутствие информации о том, с какой стороны находится поверхности материал;

- Твёрдотельная модель. Отличие данной формы задания модели от поверхностной формы состоит в том, что в объёмных моделях к информации о поверхностях добавляется информация о том, где расположен материал путём указания направления внутренней нормали.

Для решения моей задачи использование каркасной модели не подойдет, поскольку та дает не однозначного представления о форме ландшафта, также использование твердотельной модели не дает каких-либо преимуществ, поскольку для моей задачи нет необходимости знать физические свойства ландшафта. Поэтому наилучшим способом описания трехмерной модели ландшафта будет поверхностная модель.

1.3 Анализ способов представления данных о ландшафте

Существует несколько основных принципов представления данных для хранения информации о ландшафтах:

- Использование регулярной сетки высот (карта высот);
- Использование иррегулярной сетки вершин и связей, их соединяющих (хранение простой триангулированной карты);
- Хранение карты ландшафта, но в данном случае хранятся не конкретные высоты, а информация об использованном блоке. В этом случае создается некоторое количество заранее построенных сегментов, а на карте указываются только индексы этих сегментов.

1.3.1 Карта высот

Данные представлены в виде двумерного массива. Для каждой вершины с индексами $[i][j]$ в двумерном массиве определяется соответствующее значение

высоты h_{ij} .

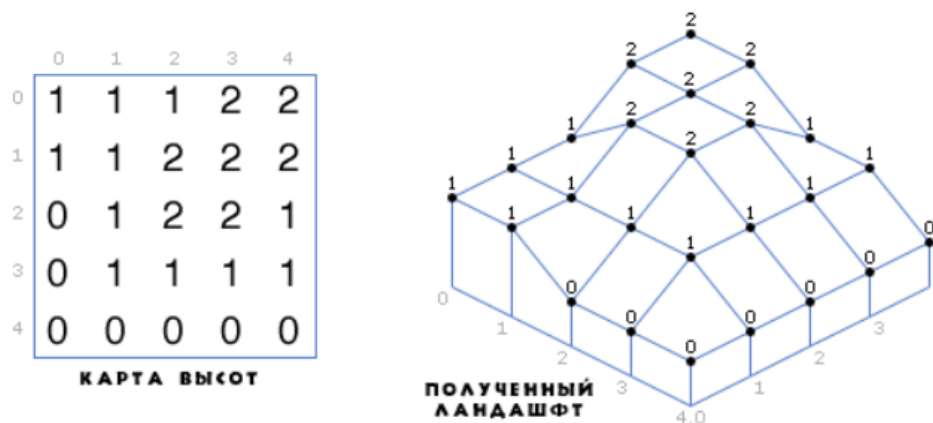


Рисунок 1.1 – Представление ландшафта с помощью регулярной сетки высот

Преимущества:

- наглядность, простота изменения данных;
- легкость нахождения координат и высоты на карте;
- из-за близкого друг к другу расположения вершинных точек можно более точно производить динамическое освещение.

Недостатки:

- слишком много описаний для точек;
- избыточность данных, например, при задании плоскости.

1.3.2 Иррегулярная сетка

Это метод представления ландшафта, который не использует равномерно распределенные узлы или точки, как в случае регулярной сетки. Вместо этого, иррегулярная сетка позволяет размещать точки или узлы в произвольных местах в зависимости от необходимости и особенностей ландшафта.

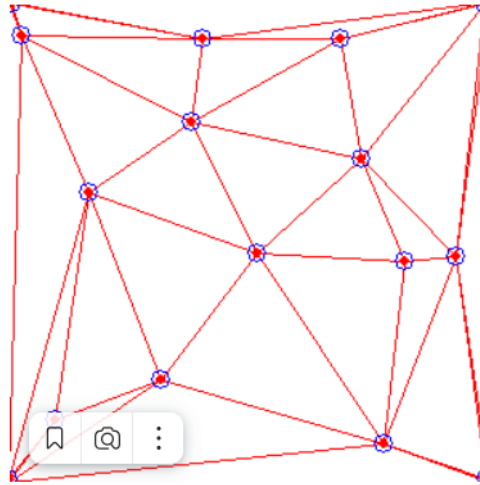


Рисунок 1.2 – Представление ландшафта с помощью иррегулярной сетки высот

Преимущества:

- используется меньше информации для построения ландшафта. Необходимо хранить только значения высот каждой вершины и связи, эти вершины соединяющие.

Недостатки:

- многие алгоритмы построения ландшафтом предназначены для регулярных сеток высот, поэтому оптимизация этих алгоритмов под иррегулярную сетку потребует значительных усилий и времени;
- из-за неравномерного расположения вершинных точек друг к другу возникает сложность при создании динамического освещения;
- сложности при хранении, модификации и просмотре такого ландшафта.

1.3.3 Посегментная карта высот

В данном способе также используются карты высот. Только вместо высот в ней хранятся индексы ландшафтных сегментов. Как эти сегменты представлены, в принципе, роли не играет. Они могут быть и регулярными, и иррегулярными (причем можно использовать и те и другие одновременно).

Преимущества:

- возможность представления больших открытых пространств;
- кроме самих ландшафтов в таких блоках можно хранить и информацию о зданиях, строениях, растениях, специфических ландшафтных решениях (например, пещеры или скалы, нависающие друг над другом);
- возможность создания нескольких вариантов одного и того же сегмента, но при разной степени детализации. В зависимости от скорости или загрузки компьютера можно выбирать более или менее детализованные варианты.

Недостатки:

- проблема стыковки разных сегментов;
- неочевидность данных. Взглянув на картинку, вы не сможете моментально представить, как это должно будет выглядеть в игре;
- проблема модификации. Проанализировав все три способа представления данных о ландшафте, выбор пал на регулярную сетку высот, поскольку:
- Первый способ дает более наглядное и понятное представление данных, позволяет легко их модифицировать. Также для этого способа существует множество алгоритмов построения ландшафта;
- При использовании иррегулярной сетки возникают сложности с хранением и модификацией данных. Также оптимизация существующих алгоритмов построения ландшафта под использование этого способа отнимет немало времени и усилий;
- Третий способ также влечет за собой проблемы с хранением и модификацией данных.

1.4 Анализ алгоритмов процедурной генерации ландшафта

В данном разделе рассмотрены различные методы и алгоритмы процедурной генерации ландшафта. Также рассмотрены преимущества и недостатки каждого метода. Основным критерием выбора алгоритма будет качество получаемого ландшафта, поскольку для решения задачи необходимо создавать правдоподобный рельеф.

1.4.1 Алгоритм Diamond-Square

Данный алгоритм является расширением одномерного алгоритма *midpoint displacement* на двумерную плоскость. Алгоритм *Diamond – Square* начинает работу с двумерного массива размера $2^n + 1$. В четырёх угловых точках массива устанавливаются начальные значения высот. Шаги *diamond* и *square* выполняются поочередно до тех пор, пока все значения массива не будут установлены.

- 1) Шаг *diamond*. Для каждого квадрата в массиве, устанавливается срединная точка, которой присваивается среднее арифметическое из четырёх угловых точек плюс случайное значение.
- 2) Шаг *square*. Берутся средние точки граней тех же квадратов, в которые устанавливается среднее значение от четырёх соседних с ними по осям точек плюс случайное значение.

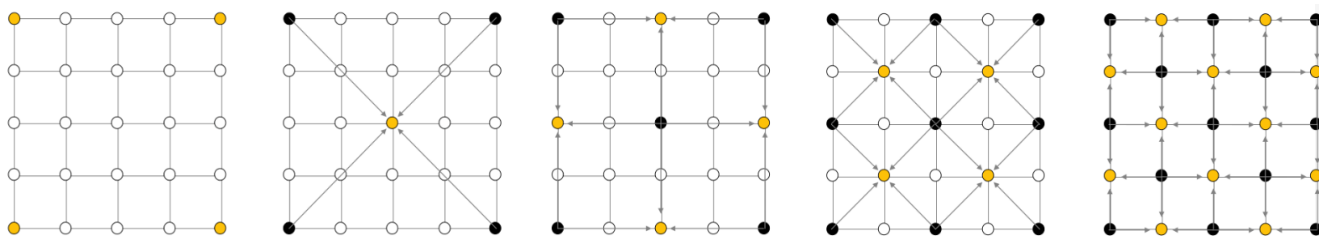


Рисунок 1.3 – Шаги, проходимые алгоритмом Diamond-Square на примере массива 5x5

Преимущества:

- алгоритм Diamond-Square достаточно прост в реализации, особенно по сравнению с некоторыми другими алгоритмами процедурной генерации ландшафта. Он может быть реализован относительно легко и быстро на различных платформах;
- алгоритм Diamond-Square позволяет быстро генерировать реалистичные и природные ландшафты с разнообразной текстурой и детализацией.

Недостатки:

- алгоритм создает заметные вертикальные и горизонтальные "складки" на краях карты из-за наиболее значительного возмущения, происходящего в прямоугольной сетке;
- при использовании этого алгоритма возникают сложности с контролем получаемого ландшафта.

1.4.2 Холмовой алгоритм

Это простой итерационный алгоритм, основанный на нескольких входных параметрах. Алгоритм изложен в следующих шагах:

- 1) создаем двумерный массив и инициализируем его нулевым уровнем (заполняем все ячейки полями);
- 2) берем случайную точку на ландшафте или около его границ (за границами), а также берем случайный радиус в заранее заданных пределах. Выбор этих пределов влияет на вид ландшафта – либо он будет пологим, либо скалистым;
- 3) В выбранной точке "поднимаем" холм заданного радиуса;
- 4) Возвращаемся ко второму шагу и так далее до выбранного количества шагов. От него потом будет зависеть внешний вид нашего ландшафта;

- 5) Проводим нормализацию ландшафта;
- 6) Проводим "долинизацию" ландшафта. Делаем его склоны более пологими.

Холм – половина шара, похоже на перевернутую параболу. Выбранный радиус определяет высоту холма и радиус его основания. Уравнение холма выглядит так:

2 Конструкторский раздел

3 Технологический раздел

4 Исследовательский раздел

ЗАКЛЮЧЕНИЕ