

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

УДК 004.43

ФОРМАЛЬНЫЙ ЯЗЫК ОПИСАНИЯ ГЕНЕРАТОРА ТРЕХМЕРНОГО ЛАНДШАФТА НА ОСНОВЕ КОМПОЗИЦИЙ ГАРМОНИЧЕСКИХ ФУНКЦИЙ

канд. техн. наук, доц. Д.О. ГЛУХОВ, канд. техн. наук, доц. Р.П. БОГУШ, Т.М. ГЛУХОВА
(Полоцкий государственный университет)

Вопросы генерирования ландшафта являются актуальными в теории компьютерного моделирования. Особую актуальность данное направление получило при решении задач трехмерной компьютерной графики. Предложен механизм формирования ландшафта с использованием композиций гармонических функций. Для достижения этой цели использован формальный язык описания генератора сложного рельефа, а также разработано соответствующее программное обеспечение.

Введение. Если попытаться классифицировать существующие методы генерирования ландшафтов, то, исходя из практической составляющей, мы предлагаем придерживаться следующего разделения:

1) **алгоритмы, генерирующие случайные ландшафты.** Такие алгоритмы не в состоянии воспроизвести один и тот же ландшафт повторно, поскольку используют генераторы псевдослучайных последовательностей. Для таких алгоритмов применяются случайные фракталы [1], диаграмма Воронова для случайного набора точек [2], для оптимизации случайных наборов используется метод релаксации Ллойда [3];

2) **алгоритмы, генерирующие неслучайные ландшафты ограниченных размеров.** Такие алгоритмы чаще всего применяются для целей визуализации фотореалистических изображений ландшафтов, а также для игр с ограниченным размером карт. Данный класс алгоритмов предлагает широкий ряд математических методов (фрактальные алгоритмы, алгоритмы на основе готовой карты высот) [4];

3) **алгоритмы, генерирующие неслучайные ландшафты неограниченных размеров.** Под неограниченностью размеров генерируемого пространства мы будем иметь в виду такие размеры, при которых невозможно хранение в памяти всего сгенерированного ландшафта и возникает потребность генерирования в режиме реального времени. Данный класс алгоритмов представляется нам наиболее перспективным, и именно на нем мы и сфокусируем наше внимание.

Основная часть. Рассмотрим возможную параметризацию задачи генерирования ландшафта. Нами предлагаются следующие генерируемые параметры.

Непрерывные:

- 1) высота блока;
- 2) цвет блока;
- 3) прозрачность блока.

Дискретные (опция):

- 4) тип блока;
- 5) тип растения;
- 6) высота растения;
- 7) размер растения по высоте;
- 8) деревья.

Основные понятия и определения

Области генерации – произвольные подпространства с различной математикой, на которые функционально разбивается все пространство с целью снижения вычислительной нагрузки при обеспечении высокого многообразия ландшафтов. Для выполнения быстрого разбиения пространства на подпространства с различной математикой предлагается задать границы между ними через плавный переход относительно некоторого базового пространства, заполняющего зоны между пространствами разбиения.

Пространства с различной математикой обладают следующими свойствами:

- 1) не пересекаются, но примыкают друг к другу, образуя разбиение покрытия;
- 2) существует ограниченное количество типов пространств с различной математикой;
- 3) имеют неправильные формы;
- 4) для любой области покрытия однозначно определяется ее тип.

Введем необходимые для построения генератора определения.

Периодическая функция – это функция, представляющая собой суперпозицию нескольких (от 3-х и более) синусоидальных волн, распространяющихся в двумерном пространстве в различных направлениях. Амплитуда волны в каждой точке двумерного пространства будет нами интерпретироваться в контексте генератора как часть генерируемого параметра (вклад в высоту ландшафта, цвет, тип и т.д.).

Каждая периодическая функция задается количеством волн, их направлениями, начальными фазами, периодами и амплитудами (рис. 1).

Ограниченная периодическая функция – периодическая функция, существующая при превышении ее амплитуды заданного порогового значения (рис. 2).

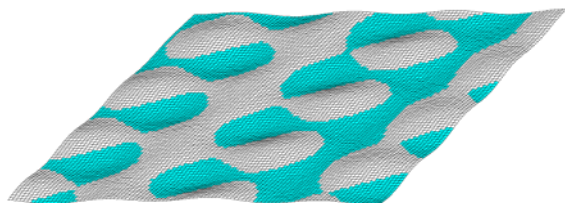


Рисунок 1 – Пример гармонической функции

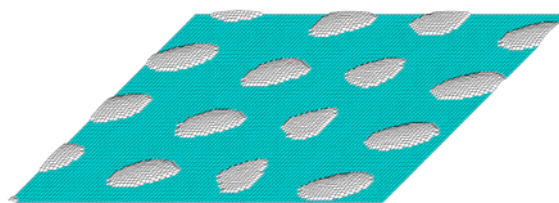
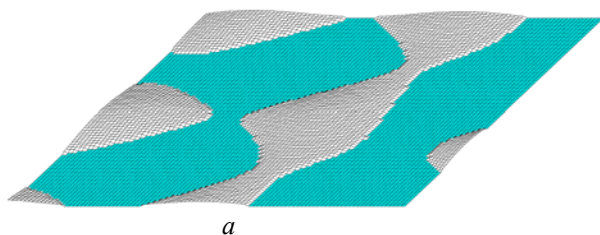
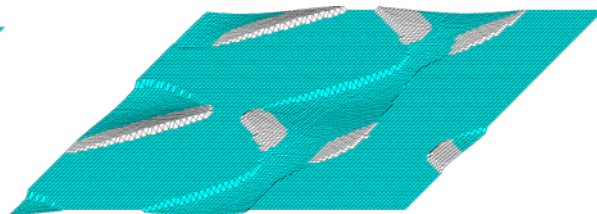


Рисунок 2 – Пример ограниченной гармонической функции

Периодическая функция ограниченная другой периодической функцией – это периодическая функция, существующая только в тех точках, в которых ограничивающая периодическая функция по амплитуде превышает заданный порог (рис. 3).



а



б

Рисунок 3 – Пример ограниченной волновой функции: а – ограничивающая, б – ограниченная

Композиция периодических функций – суперпозиция периодических функций, получаемая с целью усложнение результата интерференции и получения более разнообразных поверхностей (рис. 4).

Ограниченная композиция периодических функций – это суперпозиция периодических функций, существующая при превышении амплитуды композиции заданного порогового значения (рис. 5).

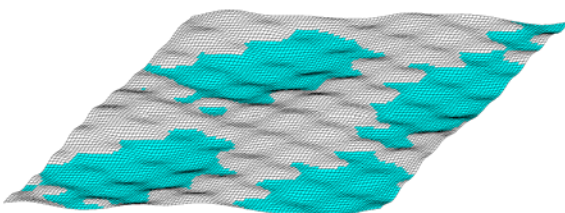


Рисунок 4 – Пример композиции периодических функций

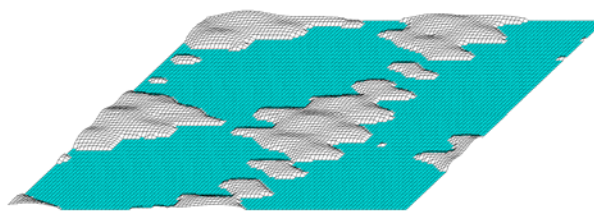


Рисунок 5 – Пример ограниченной композиции периодических функций

Инверсная периодическая функция или композиция – это периодическая функция или композиция периодических функций, отраженная относительно плоскости распространения волны (рис. 6).

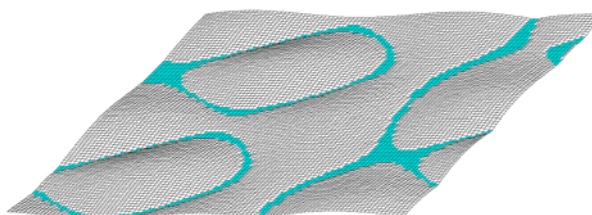


Рисунок 6 – Пример композиции (периодическая функция и ее инверсия), ограниченной ненулевым порогом

Маска – это периодическая функция или композиция периодических функций, возвращающая для всех ненулевых значений значение 1.

Маска является прекрасным механизмом разбиения пространства на подпространства, которые образуют покрытие исходного пространства. Композиция из N масок позволяет разбить пространство на $N + 1$ (если учитывать остающиеся подпространства с нулевым значением) покрывающих его подпространств (рис. 7). Причем чем выше амплитуда результирующей композиции, тем более уникальным является подпространство. Это позволяет контролировать редкость определенных генераций.

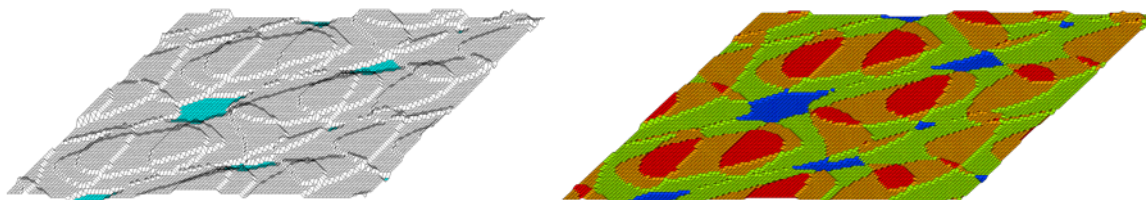


Рисунок 7 – Пример композиции нескольких масок

Разработка языка управления генератором

Базовые определения стали основой разработанного языка управления периодическими функциями и композициями периодических функций. Рассмотрим основные операторы предлагаемого языка управления генератором.

Оператор периодической функции, представляющей собой суперпозицию трех синусоид

$$\mathbf{W3} \text{ name } x_0 \ y_0 \ dx_1 \ dy_1 \ p_1 \ a_1 \ dx_2 \ dy_2 \ p_2 \ a_2 \ dx_3 \ dy_3 \ p_3 \ a_3,$$

где x_0, y_0 – нулевая точка; dx_i, dy_i – направление распространения гармоника; p_i и a_i – период и амплитуда гармоника; *name* – имя периодической функции.

Оператор композиции периодических функций

$$\mathbf{MIX} \text{ name } name_1 \ name_2 \ \dots,$$

где *name* – имя композиции; *name₁, name₂, ...* – имена периодических функций, входящих в композицию.

Оператор максимума композиции

$$\mathbf{MAX} \text{ name } name_1 \ name_2 \ \dots,$$

где *name* – имя композиции; *name₁, name₂, ...* – имена композиций-операндов.

Оператор минимума композиции

$$\mathbf{MIN} \text{ name } name_1 \ name_2 \ \dots,$$

где *name* – имя композиции; *name₁, name₂, ...* – имена композиций-операндов.

Оператор ограничения композиции по не превышению порога или периодической функции

$$\mathbf{RD} \text{ name } d,$$

где *name* – имя композиции; *d* – пороговое значение.

Оператор ограничения композиции по превышению порога или периодической функции

$$\mathbf{RDU} \text{ name } d,$$

где *name* – имя композиции; *d* – пороговое значение.

Оператор ограничения в ноль композиции или периодической функции

$$\mathbf{RDZ} \text{ name } d,$$

где *name* – имя композиции; *d* – пороговое значение.

Оператор ограничения композиции или периодической функции другой композицией или периодической функцией

$$\mathbf{RW} \text{ name}_1 \ name_2 \ d,$$

где *name₁* – имя ограничиваемой композиции; *name₂* – имя ограничивающей композиции; *d* – пороговое значение.

Оператор маски

MASK name,

где name – имя композиции, по которой строится маска.

Рассмотрим пример 1 получения композиции (рис. 8):

W3w0 0 0 -0.9 0.13 0.9 0.3 0.5 0.75 0.7 0.3 0.4 -0.87 0.5 0.3

W3w1 0 0 1 0.2 10 2 -0.5 0.7 10 2 -0.5 -0.7 10 2

W3w2 0 0 0.4 0.5 3 1 0.7 -0.5 3.2 1.5 -0.3 -0.8 2.5 1

MIXw3 w2 w0 w1

RWw3 w3 0.3

MIXELw3

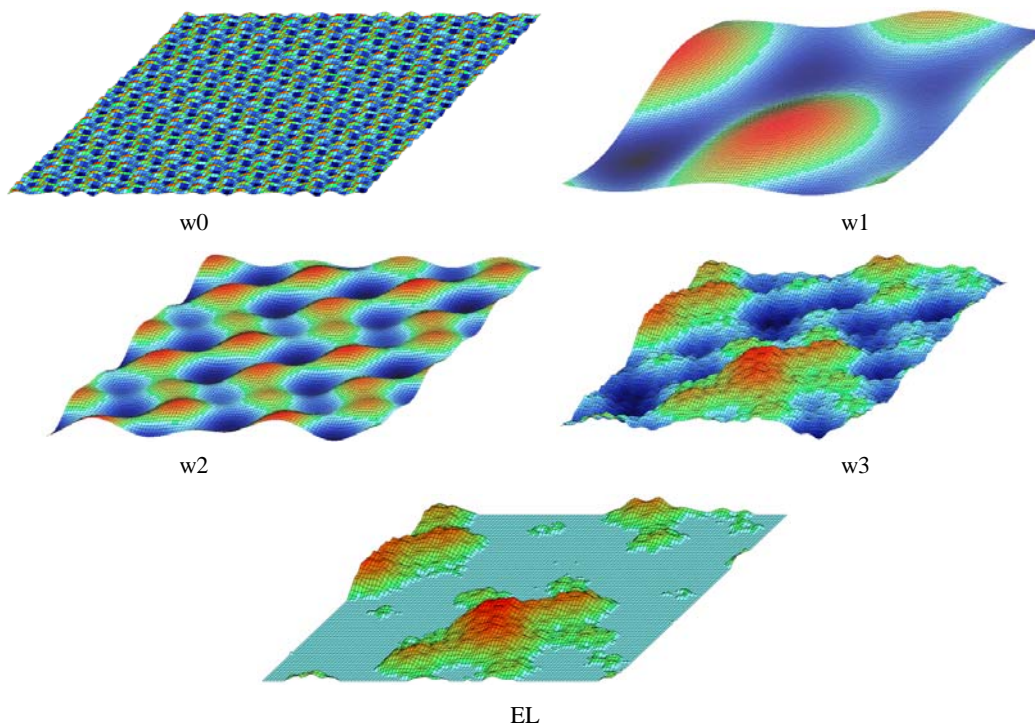


Рисунок 8 – Периодические функции и композиции из примера 1

Оператор сложения периодической функции с числом

ADD name d,

где name – имя композиции; d – число.

Оператор масштабирования композиции или периодической функции

SCALE name d,

где name – имя композиции или периодической функции; d – масштаб.

Оператор перемножения композиций или периодических функций

MUL name₁ name₂,

где name₁ – имя умножаемой композиции; name₂ – имя композиции множителя.

Оператор умножения композиций или периодической функции на число

MUL name₁ d,

где name₁ – имя умножаемой композиции; d – множитель.

Оператор перемножения композиций

MULT name₁ name₂ name₃,

где name₁ – имя композиции результата; name₂ и name₃ – имена перемножаемых композиций.

Оператор сдвига композиции или периодической функции

SHIFT name d_x d_y,

где name – имя сдвигаемой композиции; d_x d_y – величина сдвига в плоскости распространения волны.

Оператор инверсии композиции или периодической функции

REVERS name,

где name – имя композиции, подвергаемой инверсии.

Оператор селекции композиции или периодической функции

CASE name₁ name₂ name₃ d,

где name₁ – имя результирующей композиции; name₂ – имя композиции оригинала; name₃ – имя композиции условия; d – уровень условия.

Оператор выбора является мощным средством сокращения вычислительных операций, он позволяет создать карту условий для подпространств с разной математикой, где каждое подпространство локализовано на своем уровне высоты и обеспечивает плавность сопряжения подпространств.

Рассмотрим пример 2 применения селектора (оператора выбора) (рис. 9):

```

W3w0      0 0      -0.9 0.13 0.9 0.3      0.5 0.75 0.7 0.3      0.4 -0.87 0.5 0.3
W3mega1   1230 3450 0.3 0.7 130949 0.3      0.6 -0.4 134212 0.4      -0.6 -0.71 142534 0.3
W3mega2   30 13450 -0.9 0.12 230949 0.3      0.6 -0.4 334212 0.4      0.6 0.71 242534 0.3
W3mega3   0 0      0.5 0.7 109049 0.3      -0.6 -0.45 103412 0.3      -0.6 0.71 104534 0.4
MIXm1     mega1 mega2 mega3
CASEc1    w0m1 2
MIXELc1   m1
  
```

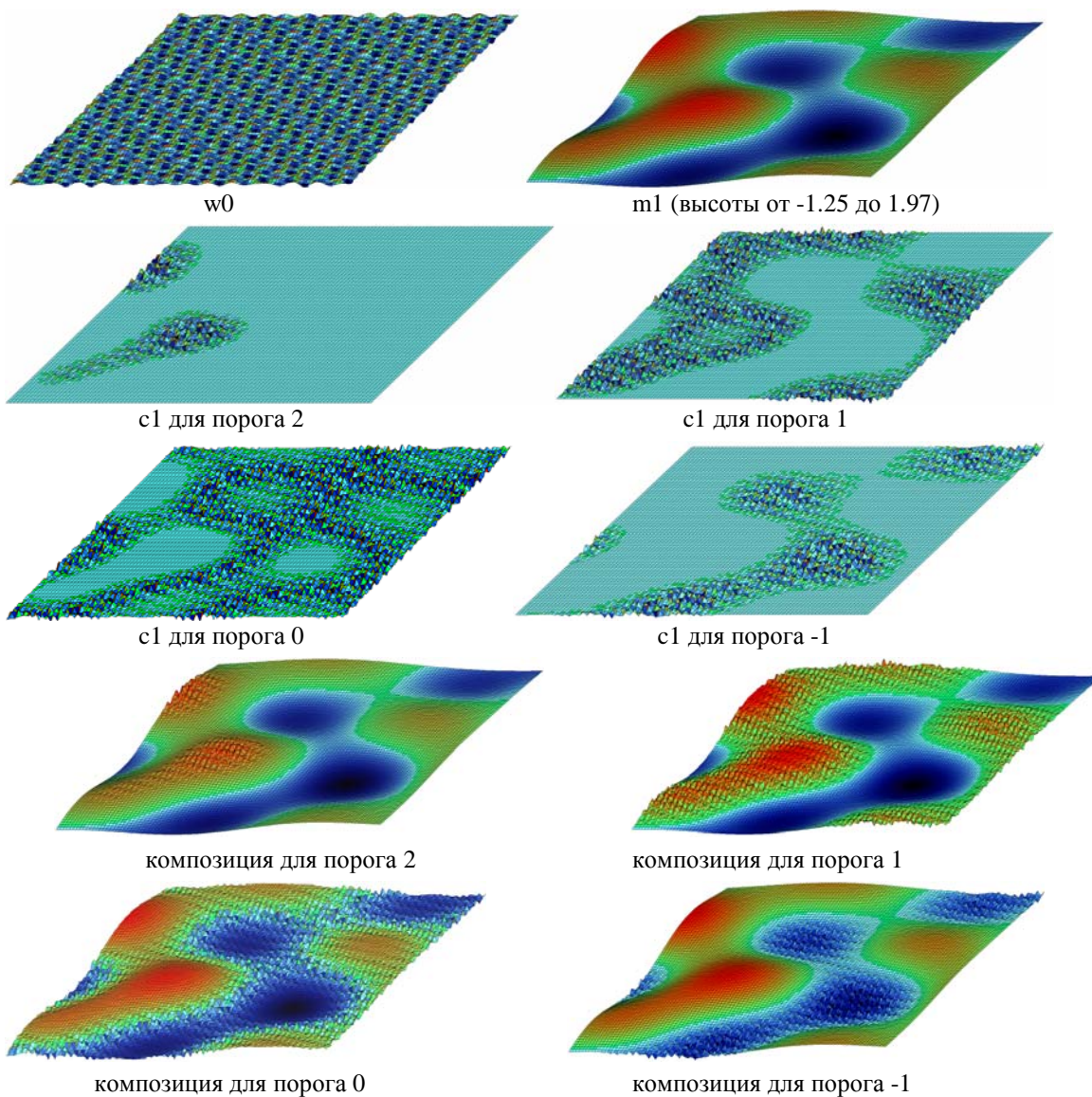


Рисунок 9 – Периодические функции и композиции из примера 2

Из примера 2 видно, как можно выборочно применять композиции для областей, определяемых уровнями представления разбиения пространства.

Оператор копирования композиций или периодических функций

COPY name₁ name₂,

где name₁ – имя копии композиции; name₂ – имя копируемой композиции.

Рассмотрим еще несколько интересных операций над периодическими функциями. Данные операции неприменимы к композициям, поскольку работают с фазами волн.

Оператор фазового сдвига

FASA name₁ name₂,

где name₁ – имя периодической функции, у которой будет происходить фазовый сдвиг на величину, задаваемую значением периодической функции второго операнда, name₂ – имя периодической функции, значения которой интерпретируются как величина фазового сдвига.

Рассмотрим пример 3 (рис. 10):

```

W3 mega1 1230 3450 0.3 0.7 530949 0.3 0.6 -0.4 634212 0.4 -0.6 -0.71 742534 0.3
W3pl1 0 0 1 0 23 0.3 1 0 28 0.5 1 0 45 0.87
W3pl2 0 0 0.6 0.3 33 0.7 0.6 0.3 38 0.5 0.6 0.3 35 0.87
W3pl3 0 0 0.3 0.6 63 0.3 0.3 0.6 48 0.53 0.3 0.6 15 0.7
W3pl4 0 0 0 1 73 0.67 0 1 126 0.75 0 1 68 0.37
MIX plx pl1 pl2 pl3 pl4
MUL plx 10
COPY super1 mega1
FASA super1 plx
    
```

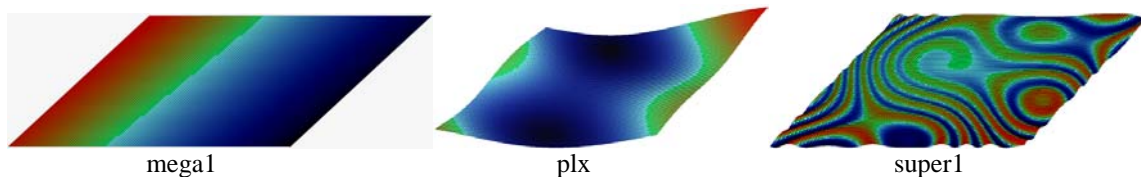


Рисунок 10 – Периодические функции и композиции из примера 3

Применение данной функции позволяет получить интересные профили поверхности, например, при генерировании пещер (рис. 11).

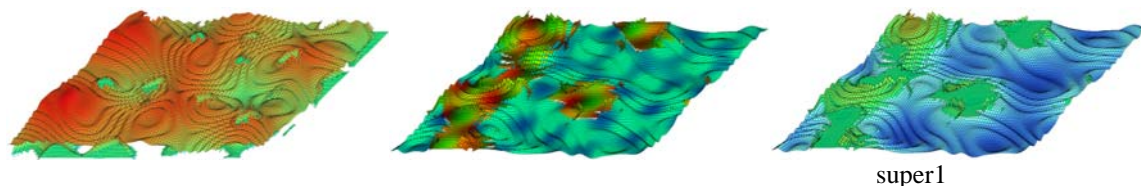


Рисунок 11 – Пещеры и композиции из примера 3

Рассмотрим операторы быстрого выбора областей.

Оператор выбора композиции

SELECT name₁ op name₂

d₁ name₃

d₂ name₄

...

END

где name₁ – имя результирующей композиции; name₂ – имя композиции шаблона выбора; d₁, d₂, ... – значения порогов; name₃, name₄ – имена композиций-источников.

Оператор выбора ближайшей композиции

SELECTNEAREST name₁ op name₂

d₁ name₃

d₂ name₄

...

END

где name₁ – имя результирующей композиции; name₂ – имя композиции шаблона выбора; d₁, d₂, ... – значения порогов; name₃, name₄ – имена композиций-источников.

Поясним работу операторов выбора на примере 4 (рис. 12):

// пусть у нас есть две периодические функции, колеблющиеся в пределах ± 1 на уровнях 12 и 6:

```
W3 whi 0 0 1 0 300 0.25 0.5 0.7 300 0.3 -0.5 0.7 300 0.35
```

```
ADD whi 12
```

```
COPY hiwhi
```

```
ADD hi 6
```

```
// Создадим холм высотой 9
```

```
W3 circle 0 0 1 0 1500 0.25 0.5 0.7 1500 0.3 -0.5 0.7 1500 0.35
```

```
MUL circle 8
```

```
ADD circle 2
```

```
/*
```

Холм выступает шаблоном. Там, где значение высоты на холме равно 8, в итоговую композицию будет проецироваться функция *hi*. С удалением от уровня 8 ее влияние будет уменьшаться, по достижению отклонения значения 1 влияние функции будет равно 0. Между уровнями 8 и 9 будет наблюдаться плавный переход от функции *whi* к функции *hi*.

```
*/
```

```
SELECTWoncircle
```

```
9 hi
```

```
8 whi
```

```
END
```

```
/*
```

В отличие от предыдущего выбора оператор **SELECTNEAREST** обеспечивает скачкообразный переход к функции, чей порог оказывается ближе.

```
*/
```

```
SELECTNEARESTWNoncircle
```

```
9 hi
```

```
8 whi
```

```
END
```

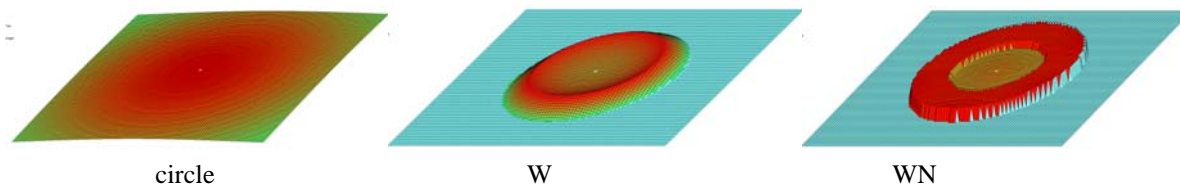


Рисунок 12 – Визуализации к примеру 4

Макрокоманды

Макрокомандами (макросами) называются сложные операторы, которые построены на основе комбинации более простых команд (базовых команд).

Синтаксис описания макроса:

```
PROCname
```

```
// тело макроса в виде последовательности операторов языка управления генератором.
```

```
// в операндах операторов могут использоваться ключи вставки реальных параметров,
```

```
// передаваемых при вызове макроса. Для вставки параметров применяется
```

```
// следующий синтаксис ключа: #0#, #1#, #2#, ... с номером операнда.
```

```
END
```

где *name* – имя макроса.

Оператор вызова макроса

```
CALL name1 name2 p1 p2 p3 ...
```

где *name₁* – имя макроса; *name₂* – имя результирующей композиции; *p₁, p₂, p₃, ...* – операнды (имена композиций, числа).

Рассмотрим пример 5 описания макроса и результат его вызова:

```

PROCgo
/*
создадим две периодические функции: первую с именем go1#0# – основную, и вторую с именем
go2#0# – для фазового сдвига.
*/
W3 go1#0# #1# #2# 1 0.2 #3# 0.33 0.5 0.7 #3# 0.33 -0.5 0.7 #3# 0.33
W3 go2#0# #1# #2# 1 0.2 100 1 0.5 0.7 100 1 -0.5 0.7 100 1
FASA go1#0# go2#0#
MIX #0# go1#0#
END
CALL go S1 0 0 200
CALL go S2 100 0 200
CALL go S3 200 0 200
CALL go S4 300 0 200
CALL go S5 400 0 200
MIXGS1 S2 S3 S4S5

```

Результат выполнения вызовов макроса представлен на рисунке 13.

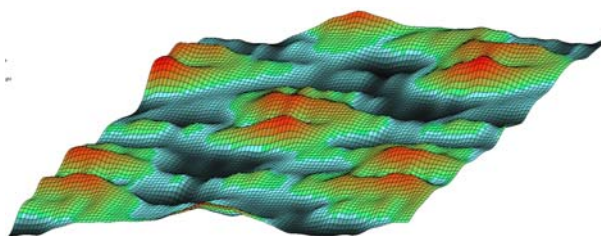


Рисунок 13 – Визуализации к примеру 5

Заключение. Разработанный в рамках данной работы метод генерирования ландшафта на основе композиции гармонических функций и предложенный формальный язык описания генератора ландшафта обладают выразительностью и позволяют формировать оптимизированные по производительности генераторы неограниченного воспроизводимого ландшафта. Данный инструмент может быть применен в области компьютерной графики и 3D-моделирования.

ЛИТЕРАТУРА

1. Paul Martz Generating Random Fractal Terrain [Электронный ресурс]. – Режим доступа: <http://www.gameprogrammer.com/fractal.html>.
2. Guibas, L. Primitives for the manipulation of general subdivisions and the computation of Voronoi / Lenoidas Guibas, Jorge Stolfi // ACM. – 1985. – P. 107. – (Retrieved 2009-08-01).
3. Lloyd, S. P. Least squares quantization in PCM / Stuart P. Lloyd // IEEE Transactions on Information Theory. – 1982. – Vol. 28 (2). – P. 129–137. – DOI : 10.1109/TIT.1982.1056489.
4. Advances in multimedia modelling : 13th International Multimedia Modeling Conference / ed. Tat-Jen Cham. – 2007. – ISBN 3-540-69428-5.

Поступила 18.09.2015

FORMAL LANGUAGE OF GENERATOR DESCRIPTION OF THE THREE-DIMENSIONAL LANDSCAPE ON THE BASIS OF THE COMPOSITIONS OF HARMONIC FUNCTIONS

D. GLUKHOV, R. BOGUSH, T. GLUKHOVA

Generating landscape are relevant to the theory of computer simulation. Of particular relevance, this trend has been in solving the three-dimensional computer graphics. In this paper an attempt is made to offer a mechanism of formation of the landscape using the compositions of harmonic functions. To achieve this, there is provided a formal description language generator difficult terrain, as well as to develop appropriate software.