

On the figures we can see, that two controls compare with each other. And when we consider, the active element out control system is stronger than competitor's control system.

Conclusion. The problem of active system modeling was researched. There are different approaches to model such systems. We used identification theory approach for solving the problem. According this approach it's necessary to construct control system and reach the desired value of output variable. The problem is difficult because of competitor. The competitor is another control system at the point identification theory. If we use ordinary way to construct control system the desired value isn't reached. The new scheme including the active element for constructing control system is proposed. It helps to predict behavior of competitor. The proposed approach works that desired value is reached.

References

1. IBM 100 Icons of Progress. Deep Blue. 2011.
2. Mazrooei P., Archibald C., and Bowling M. Automating Collusion Detection in Sequential Games. Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence. July, 2013.
3. Blincoe R. \$8.9m poker prize up for grabs – humans only please. Why can't a bot play at this level? The Register. 5th November, 2010.
4. Jackson P. *Vvedeniye v ekspertnyye sistemy* (Introduction in the expert systems). 3rd edition. Moscow, Published House Williams, 2001, 624 p. [in Russian]
5. Ruchkin V., Fulin V. *Universal'nyy iskusstvennyy intellekt i ekspertnyye sistemy* (Univarsal artificial intelligent and expert systems). S. Peterburg, BHV-Peterburg Publ., 2009, 240 p. [in Russian].
6. Jeffery M. *Data-Driven Marketing. The 15 Metrics Everyone in Marketing Should Know*. John Wiley and Sons, 2010.
7. Leung L. *Identifikatsiya sistem. Teoriya pol'zovatelya* (System identification: theory for user). Moscow, Nauka Publ., 1991, 432 p. [in Russian].
8. Eykhoff P. *Osnovy identifikatsii sistem upravleniya. Otsenivaniye parametrov sostoyaniya* (Foundations of control systems identification). Moscow, Mir Publ., 1975, 688 p. [in Russian].
9. Medvedev A. [Theory of the nonparametric systems. Processes]. *Vestnik SibGAU*, 2010, no. 3 (29), p. 4–9.
10. Watson G. *Smooth regression analysis*. Sankhya. Ser. A. 1964.

Библиографические ссылки

1. IBM 100 Icons of Progress. Deep Blue. 2011.
2. Mazrooei P., Archibald C., and Bowling M. Automating Collusion Detection in Sequential Games // Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence. July, 2013.
3. Blincoe R. \$8.9m poker prize up for grabs – humans only please. Why can't a bot play at this level? The Register. 5th November, 2010.
4. Джексон П. Введение в экспертные системы. 3-е изд. М. : Вильямс, 2001. 624 с.
5. Ручкин В. Н., Фулин В. А. Универсальный искусственный интеллект и экспертные системы. СПб. : БХВ-Петербург, 2009. 240 с.
6. Jeffery M. *Data-Driven Marketing. The 15 Metrics Everyone in Marketing Should Know*. John Wiley and Sons, 2010.
7. Льюинг Л. Идентификация систем. Теория пользователя. М. : Наука, 1991. 432 с.
8. Эйкхофф П. Основы идентификации систем управления. Оценивание параметров состояния / под ред. Н. С. Райбмана М. : Мир, 1975. 688 с.
9. Медведев А. В. Теория непараметрических систем. Процессы // Вестник СибГАУ. 2010. Вып. 3 (29). С. 4–9.
10. Watson G. *Smooth regression analysis*. Sankhya. Ser. A. 1964.

© Стрельников А. В., 2014

УДК 004.942

ПРИМЕНЕНИЕ АЛГОРИТМА SPACE COLONIZATION ПРИ ТРЕХМЕРНОМ МОДЕЛИРОВАНИИ СЛОЖНЫХ ПРИРОДНЫХ ОБЪЕКТОВ

А. А. Ткачева

Сибирский государственный аэрокосмический университет имени академика М. Ф. Решетнева
Российская Федерация, 660014, г. Красноярск, просп. им. газ. «Красноярский рабочий», 31
E-mail: tkacheva@mail.sibsau.ru

Для создания моделей лесной растительности используются особые методы и алгоритмы. Изучение процесса роста растений требует более мощных инструментов моделирования для того, чтобы описать и смоделировать различные процессы функционирования растений. В последнее десятилетие наиболее часто для решения подобных задач используется формализм L-систем. Основная идея L-систем (Линденмайер системы) – постоянная перезапись элементов строки, L-система оперирует со строкой символов по специальным правилам, начиная с первоначальной простой аксиомы, позволяя генерировать различные объекты с фрактальной структурой. Рассматривается адаптация L-систем и Space Colonization алгоритма на языке Python, популяр-

ного и мощного открытого лицензионного динамического языка. Использование данного языка программирования позволяет строить сложные модели, кроме того, приведены полученные результаты для различных исходных данных, получены трехмерные модели, произведена визуальная оценка реалистичности полученных моделей.

Ключевые слова: трехмерное моделирование растительности, алгоритм Space Colonization, L-системы, имитационное моделирование.

SPACE COLONIZATION ALGORITHM APPLICATION IN THREE-DIMENSIONAL MODELING OF COMPLEX NATURAL OBJECTS

A. A. Tkacheva

Siberian State Aerospace University named after academician M. F. Reshetnev
31, Krasnoyarsky Rabochy Av., Krasnoyarsk, 660014, Russian Federation
E-mail: tkacheva@mail.sibsau.ru

Special methods and algorithms are used to create models of forest vegetation. The study of plant development requires increasingly powerful modeling tools to help understand and simulate the growth and functioning of plants. In the last decade, the formalism of L-systems has emerged as a major paradigm for modeling plant development language. Lindenmayer systems (L-systems) are formalism for parallel rewriting, used for generating fractal. In this work, we present an adaptation of L-systems and Space Colonization algorithm to the Python language, a popular and powerful open-license dynamic language. We show that the use of dynamic language properties makes it possible to build complex models, moreover the results obtained for a variety of input data, a three-dimensional model, made a visual assessment of the realism of the obtained models are given in the article.

Keywords: three dimensional plant modeling, Space colonization algorithm, L-systems, simulating modeling.

Процесс моделирования ландшафтных сцен разбивается на две части: непосредственная генерация самого ландшафта и наполнение исходного ландшафта объектами сцены (растительность, источники воды, света). Моделирование антропогенных объектов (здания, заборы, транспортные средства) не вызывает значительных трудностей, так как геометрия данных объектов состоит из таких примитивов, как куб, шар, цилиндр, и описывается математическими системами однозначно. Необходимость в ландшафтных сценах возникает в различных пакетах архитектурного моделирования, симуляторов виртуальной реальности, геоинформационных системах [1]. Реалистичность ландшафта напрямую зависит от точности объектов, наполняющих эту сцену. Растительность является основным элементом естественного ландшафта, соответственно, от реалистичности полученных моделей деревьев и кустарников будет зависеть сложность и естественность полученной сцены. В свою очередь, моделирование растительности является нетривиальной задачей. Сложность данной задачи заключается в том, что форма объектов деревьев и кустарников далека от геометрических примитивов и их сочетания, обладает ветвящейся структурой, примитивно которую можно описать фракталами. Модели с такой структурой проектируются в CAD-системах неоднозначно, а также требуют значительных временных затрат.

Существуют два основных подхода в ландшафтном моделировании: моделирование всей сцены в специальных пакетах трехмерного моделирования (Vue9.5 xStream, 3dsMAX, XSI, Maya, DAZ 3D Bryce)

и моделирование на основе данных наземного лазерного сканирования и аэрофотоснимков. Для каждого из этих подходов можно выделить общие принципы моделирования растительности: математическое моделирование, основанное на правилах и грамматиках; интерактивное моделирование, основанное на шаблонах моделей, и моделирование, основанное на ряде изображений исходного объекта. Альтернативой является моделирование роста дерева на основе L-систем, данный подход изобрел датский математик и биолог Линденмайер [2]. L-система – это грамматика некоторого языка программирования:

$$G = \{V, \omega, P\}, \quad (1)$$

где V – алфавит; $\omega \in V$ – аксиомы; $P \in V \times V$ – правила.

В программном виде данное определение L-систем выглядит следующим образом:

```

module Apex(age), Internode(length, radius)
MAX_AGE, dr = 10, 0.02 # константы
Axiom: Apex(0) # определение аксиом
production: # описание правил ветвления
Internode(l,r) --> Internode(l,r + dr)
Apex(age):
if age < MAX_AGE:
produce Internode(1,0.05)/(137.5)
[+(40)Apex(age+1)]Apex(age+1)
    
```

Алфавит L-систем состоит из набора стандартных символов. Для каждого типа L-систем расширяется стандартный алфавит, например, для скобочных

L-систем вводятся символы «[» и «]». Линденмайер описал несколько типов L-систем. Представим классификацию от простых L-систем к более сложным: скобочные (bracketed), параметрические (parametric), контекстно-зависимые (context-sensitive), стохастические (stochastic) [3]. Основное отличие стохастических L-систем заключается в том, для каждого символа алфавита можно задать не одно правило, а несколько, снабдив каждое из них вероятностью его применения. В этом случае развитие системы уже не будет жестко предопределенным – при подстановках будет происходить случайный выбор одного правила из нескольких в соответствии с заданными вероятностями.

Дадим определение стохастическим L-системам: пусть существует только одно x из множества V^* для каждого a из множества V ; если x может быть выбрано с определенной степенью свободы, система называется стохастической L-системой, обозначаемой в виде четверки $G = \{V, \omega, P, \pi\}$. Функция π отображает соответствующего предшественника (a) на вероятностях предшественника (x). Общая сумма всех вероятностей x , присваиваемых конкретному a , должна быть равна 1. Пример программной реализации стохастической системы:

```
from random import * # использование генератора
случайных чисел
p0 = 0.5 # исходная вероятность
Axiom: Apex(0)
production:
Apex(age): # описание процесса ветвления
  nproduce Internode(1,0.1)/(180)
  if uniform() < p0: nproduce[+(20)Apex(age+1)]
  produce Apex(age+1)
```

Данный подход позволяет описывать развитие кроны дерева и кустарника, а также учитывать практически все параметры дерева, измеряемые в природе; сложность заключается в естественном имитировании различных типов ветвления: моноподиальном, симподиальном и ложнодихотимическом (тернарном). Для каждого из типов ветвления разработаны собственные L-системы (рис. 1).

При моноподиальном типе ветвления один из углов ветвления равен нулю и ясно выражена главная

ось. Правила ветвления выглядят следующим образом:

```
p1: A(l,w) : *→
      !(w)F(l)[&(a0)B(l*r2,w*wr)]/(d)A(l*r1,w*wr)
p2: B(l,w) : *→
      !(w)F(l)[-a2]C(l*r2,w*wr)]C(l*r1,w*wr)
p3: C(l,w) : *→
      !(w)F(l)[+a2]B(l*r2,w*wr)]B(l*r1,w*wr)
```

Другие L-системы охватывают симподиальные структуры, в которых оба дочерних сегмента формируют ненулевой угол с материнским. В этом случае активность главной вершины снижена из-за формирования ствола. Тернарный тип ветвления является по сути симподиальным типом и отличается лишь тем, что отсутствует жесткое предопределение длин ветвей, их развитие описывается вместе с процессом роста растительности. Приведем псевдокод описания тернарного типа ветвления:

```
d1= 94.74 # угол ветвления первой боковой оси
d2= 132.63 # угол ветвления второй боковой оси
a=18.95 # первоначальный угол ветвления
lr=1.109 # коэффициент пропорциональности
длины ветви
vr=1.732 # коэффициент пропорциональности
ширины ветви

ω : !(1)F(200)/(45)A
p1 : A : *→ !(vr)F(50)[&(a)F(50)A]/(d1)
      [&(a)F(50)A]/(d2)[&(a)F(50)A]
p2 : F(l) : *→ F(l*lr)
p3 : !(w) : *→ !(w*vr)
```

Для описания растительности в глобальной системе окружающей среды используется особый тип L-систем, зависимый от окружения в модельном пространстве (environmentally sensitive); в данных системах используются различные запросы получения позиции вектора и его ориентации в пространстве:

- ? P(x, y, z) – получение текущей позиции;
- ? H(x, y, z) – вектор направления;
- ? U(x, y, z) – отклонение вверх вектора направления;
- ? L(x, y, z) – отклонение влево вектора направления.

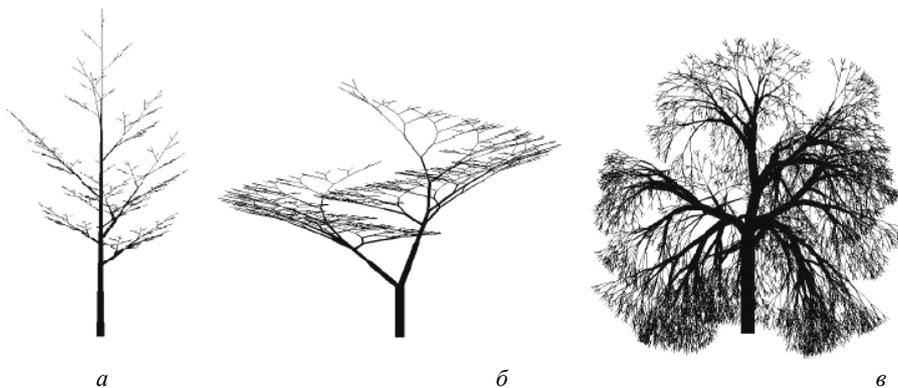


Рис. 1. Модели деревьев L-систем: а – моноподиальный тип ветвления; б – симподиальный тип ветвления; в – тернарный тип ветвления

Кроме приведенной классификации, существуют и «оригинальные» типы ветвления (ветвления типа пагоды и терминалия), которые также заслуживают внимания, хотя в природе не так часто встречаются, однако именно такие объекты придают ландшафту естественность и разнообразие.

Тем не менее, несмотря на успешную работу в создании реалистичных моделей деревьев, по-прежнему остается невозможным точно и естественно моделировать процесс ветвления, физические движения деревьев под воздействием ветра, взаимовлияние растительности и соседствующих объектов [4]. Ветровые модели в основном применяются к стволу и ветвям, листья же в общем случае представляются в виде текстуры и являются неподвижными, что сказывается на их трехмерном представлении. Также при взаимодействии растительности друг с другом и другими антропогенными объектами необходимо учитывать развитие кроны дерева в ограниченном пространстве.

Взаимодействие растительности с окружающей средой можно рассматривать с глобальной и локальной точки зрения. Глобальные переменные взаимодействуют с растительностью в целом: температурный режим, газовый состав воздуха, воздействие солнечных лучей на рост (фотосинтез) и пр. К локальным переменным относятся те, которые изменяют лишь отдельные параметры растительности, такие как позиция и ориентация объектов в пространстве [5]. Будем рассматривать лишь локальные переменные окружающей среды, а именно, взаимное влияние соседствующей растительности друг на друга, воздействие ветра, ограниченность пространства растущих деревьев: стены, дома и другие архитектурные комплексы, обрезка деревьев.

Рассмотрим иной подход к моделированию растительности – использование алгоритма Space Colonization. Несмотря на то, что изначально данный алгоритм применялся для предсказания движения пешеходов в толпе [6], его можно применить и для моделирования растительности, так как нетрудно заметить, что для определения возможной траектории движения пешехода строится некая ветвящаяся структура, подобная растениям.

Основная идея алгоритма Space Colonization заключается в итеративном добавлении новых элементов (ветвей) к существующей геометрической структуре объекта (дерева), сформированного на предыдущих шагах [7].

Данный алгоритм является адаптивным, т. е. процесс роста зависит от следующих параметров: близлежащее присутствие объектов окружающего мира, соседство с другими деревьями. Изначально задается облако из N точек (листьев), их суммарное количество обычно составляет сотни или тысячи, соответственно, на этом этапе можно задать некие препятствия роста дерева, причем точки, выходящие за заданные границы, будут отбрасываться. Также на этом этапе задается начальная точка ствола, с которой начинается итеративное построение «скелета» дерева, на каждой итерации воспроизводится ветвь в направлении ближайшей точки кроны дерева. Пошаговая работа алгоритма представлена на рис. 2.

Математическая постановка задачи заключается в следующем: пусть имеется некоторая «корневая» точка v , из этой точки может выходить несколько ветвей, т. е. в окрестности данной точки имеется множество точек $S(v)$, находящихся на расстоянии меньшем, чем d_i – параметр, задаваемый пользователем (параметр управления). Если множество $S(v)$ непустое, то новая точка v' присоединяется к общей структуре дерева путем построения сегмента (v, v') . Причем новая точка v' находится на расстоянии D от исходной точки v в направлении средненормированного вектора n по отношению ко всем источникам:

$$D = \frac{v' - v}{\hat{n}}, \quad (2)$$

$$\hat{n} = \frac{\bar{n}}{\|\bar{n}\|}, \quad (3)$$

$$\sum_{s \in S(v)} \frac{s - v}{\|s - v\|}. \quad (4)$$

Данный процесс заканчивается при прохождении всех точек кроны дерева (N) либо при выполнении заданного пользователем количества итераций. Некоторые узловые точки удаляются, т. е. при построении нового сегмента происходит проверка: все точки, находящиеся на расстоянии dk (dk – *killdistance*, входной параметр) от точки v' , удаляются. Далее для более реалистичной визуализации задается сглаживание ветвей – скругление изгибов-переломов на узловых точках, т. е. аппроксимация по точкам (рис. 2, *e*). На рис. 2, *ж* показана объемная визуализация полученного «скелета» с помощью цилиндров разных радиусов. При соединении дочерних ветвей в узловой точке разных радиусов r_1 и r_2 , новый радиус r вычисляется по формуле 5:

$$r^n = r_1^n + r_2^n, \quad (5)$$

где $n \in (2, 3)$.

На конечном этапе происходит наложение листвы на существующую структуру ветвей (рис. 2, *з*). Данный алгоритм позволяет получать реалистичные модели деревьев и кустарников различной формы.

Приведем примеры различных исходных облаков точек. Для детального построения геометрии дерева необходимо задавать облака точек, содержащих сотни элементов, также немаловажным является и распределение точек по всей области облака (рис. 3).

Полученные результаты, приближенные к реальности, представлены на рис. 4.

Одна из наиболее важных особенностей данного алгоритма заключается в возможности моделирования деревьев и кустарников различной формы, связанной с воздействием окружающей среды: освещенность, расположенность моделируемых объектов вдоль стены или иных преград, соседство рядом с другой растительностью (рис. 5).

Все представленные выше результаты выполнены с помощью динамического языка программирования Python, все исходные облака точек задавались с помощью генератора случайных чисел, дальнейшим направлением работы является использование данных лазерного сканирования (рис. 6).

Визуализация написанного скрипта выполнена средствами графического редактора Blender (открытое программное обеспечение). Основная геометрия дерева – процесс ветвления – осуществлялся по алгоритму Space Colonization, стохастические L-системы

использовались на конечных итерациях (дорисовка мелких ветвей) перед наложением текстур на объект. Построенные в соответствии с изложенным методом 3D-ландшафтные сцены являются реалистичными и не требуют больших вычислительных затрат.

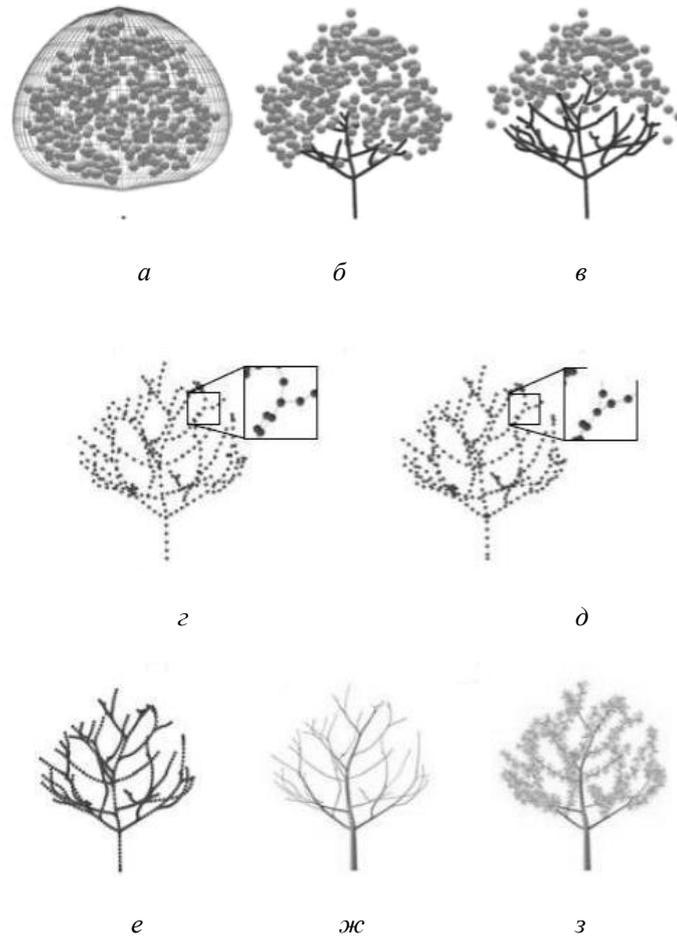


Рис. 2. Последовательность шагов алгоритма SpaceColonization:
a – задание облака точек кроны дерева; *б* – прорисовка общего предварительного «скелета» ветвей;
в – инициализация почек; *г, д* – прореживание точек ветвей; *е* – скругление ветвей;
ж – объемное (цилиндрическое) моделирование ветвей; *з* – наложение лиственной массы

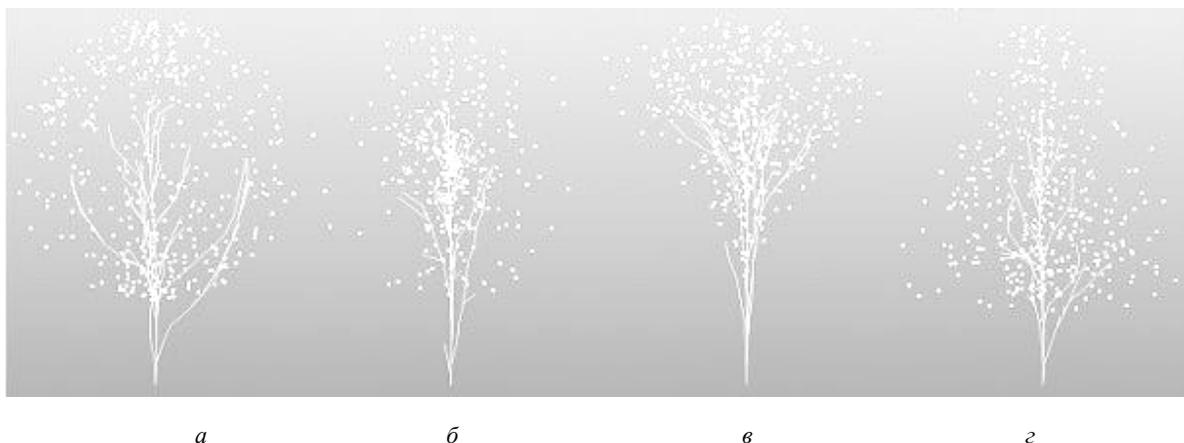


Рис. 3. Результаты первых этапов работы алгоритма SpaceColonization для различных облаков точек: *a*, *б* – облака точек с элементами, сосредоточенными ближе к границе / центральной оси; *в, г* – концентрация элементов облака в верхних и нижних частях соответственно



Рис. 4. Модели дерева и кустарника, полученные с помощью алгоритма Space Colonization

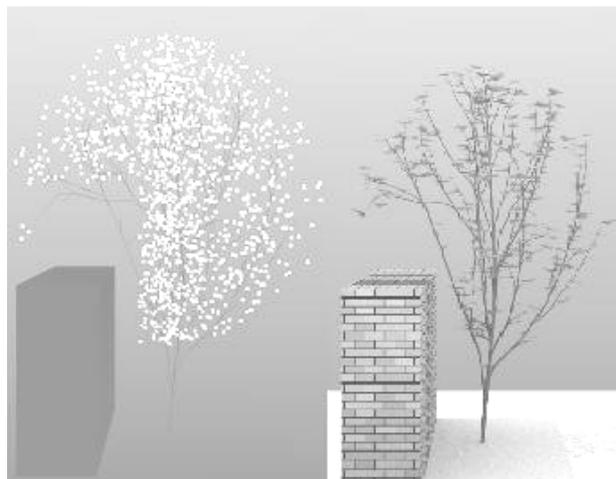


Рис. 5. Моделирование объекта в условиях ограниченного роста

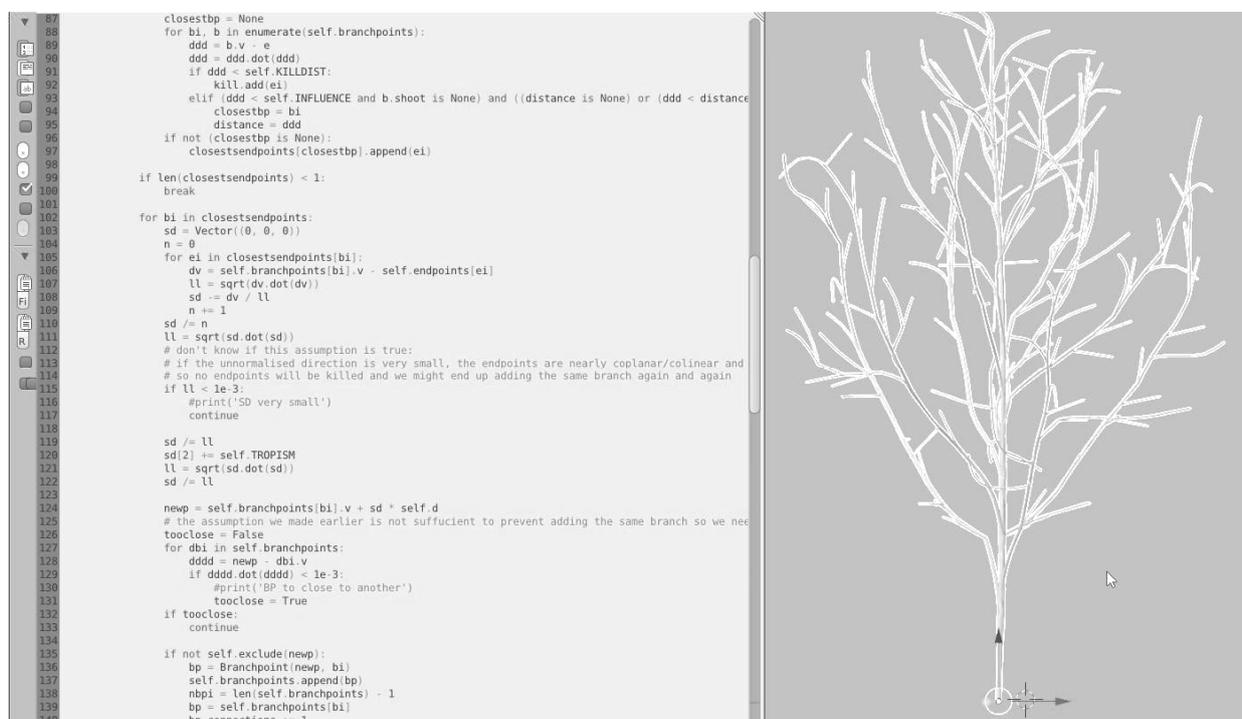


Рис. 6. Моделирование дерева в Blender с использованием Python API

Библиографические ссылки

1. Якубайлик О. Э. Проблемы формирования информационно-вычислительного обеспечения систем экологического мониторинга // Вестник СибГАУ. 2012. Вып. 3 (43). С. 96–102.
2. Prusinkiewicz P., Lindenmayer A. The algorithmic beauty of plants. New York : Springer-Verlag, 1990. 256 p.
3. Longay S., Runions A., Boudon F., Prusinkiewicz P. TreeSketch: Interactive procedural modeling of trees on a

4. Favorskaya M., Tkacheva A. Rendering of wind effects in 3D landscape scenes // Procedia Computer Sciences. 2013, vol. 22, p. 1229–1238.
5. Boudon F. [et al.]. L-Py: an L-System simulation framework for modeling plant development based on a dynamic language // Frontiers in technical advances in plant science. 2012, p. 1–20.

6. De Lima Bicho A. [et al.] Simulating crowds based on a space colonization algorithm // *Computers and Graphics*. 2012, № 36, p. 70–79.

7. Runions A., Lane P. Modeling trees with a space colonization algorithm // *Eurographics Workshop on Natural Phenomena*. 2007, p. 63–70.

References

1. Yakubailik O. *Vestnik SibGAU*. 2012, no. 3 (43), p. 96–102.

2. Prusinkiewicz P., Lindenmayer A. NewYork, *Springer-Verlag*, 1990. 256 p.

3. Longay S., Runions A., Boudon F., Prusinkiewicz P. *Proceedings of the Eurographics Symposium on Sketch-Based Interfaces and Modeling*. 2012, p. 107–120.

4. Favorskaya M., Tkacheva A. *Procedia Computer Sciences*. 2013, vol. 22, p. 1229–1238.

5. Boudon F., Pradal C., Cokelaer T., Prusinkiewicz P., Godin C. *Frontiers in technical advances in plant science*. 2012, p. 1–20.

6. de Lima Bicho A., Rodrigues R., Musse S., Jung C., Paravisi M. *Computers and Graphics*. 2012, № 36, p. 70–79.

7. Runions A., Lane P. *Eurographics Workshop on Natural Phenomena*. 2007, p. 63–70.

© Ткачева А. А., 2014

УДК 539.3+539.4

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ ПРЕДЕЛЬНЫХ ДЕФОРМАЦИЙ ПЛОСКИХ КОНСТРУКЦИЙ, АРМИРОВАННЫХ ВДОЛЬ КРИВОЛИНЕЙНЫХ ТРАЕКТОРИЙ

Н. А. Федорова

Сибирский федеральный университет,
Российская Федерация, 660074, г. Красноярск, ул. Киренского, 26
E-mail: ran@akadem.ru

На основе структурной модели в рамках плоской неоднородной линейной задачи упругости решена задача рационального армирования криволинейными волокнами. Рассмотрено растяжение трехслойного армированного диска под действием центробежной силы в полярной системе координат. Получена разрешающая система обыкновенных дифференциальных уравнений относительно радиальной и окружной компонент перемещений в осесимметрической постановке задачи. Изучено влияние структурных параметров на предельное нагружение конструкции. Показано, что за счет выбора структуры армирования возможно увеличение предельной скорости вращения диска почти в два раза.

Ключевые слова: армирование, структурная модель, криволинейные траектории.

MATHEMATICAL MODELLING FOR EXTREME DEFORMATIONS OF PLANAR CONSTRUCTIONS REINFORCED WITH CURVILINEAR TRAJECTORIES

N. A. Feodorova

Siberian Federal University
26, Kirenskogo Av., Krasnoyarsk, 660074, Russian Federation
E-mail: ran@akadem.ru

The problem of curvilinear fibers rational reinforcement is solved by reference to the structural model within the heterogeneous liner elasticity problem. Traction of a three-layered reinforced disc under the centrifugal force action in the polar coordinate system is considered. The resolving system of ordinary differential equations in radial and peripheral movement components for the axisymmetric problem definition is obtained. The effect of structural parameters for a construction limit stressing is studied. It is approved that by reinforcement structure choosing the extreme speed of a disc rotation can be increases twofold.

Keywords: reinforcement, structural model, curvilinear trajectories.

В современной промышленности широко используются тонкостенные элементы из волоконистых композитных материалов. Волокнистое армирование устанавливает анизотропию свойств материала [1] и позволяет применять новые принципы проектирования и изготов-

ления изделий, основанные на том, что материал и изделие создаются одновременно в рамках единого технологического процесса. В результате получается изделие с новыми уникальными эксплуатационными качествами. До недавнего времени армирование осуществлялось