

# Создание сложных поверхностей на базе функций возмущения из полигональных сеток

С.И. Вяткин, Б.С. Долговесов, А.В. Рухлинский  
Институт автоматики и электрометрии СО РАН,  
Новосибирск, Россия  
[sivser@mail.ru](mailto:sivser@mail.ru), [bsd@iae.nsk.su](mailto:bsd@iae.nsk.su), [ares@iae.nsk.su](mailto:ares@iae.nsk.su)

## Аннотация

Предлагается метод создания поверхностей на базе функций возмущения из сложных полигональных моделей без предварительной вокселизации последних. Данные функционально заданные поверхности строятся из патчей произвольных форм, благодаря чему достигается высокий коэффициент геометрического сжатия трехмерных объектов. Этот метод был применен ко многим полигональным моделям и был вычислен график зависимости искажения формы объектов от коэффициента сжатия.

**Ключевые слова:** Геометрическое моделирование, патчи произвольных форм, функции возмущения.

## 1. ВВЕДЕНИЕ

В статье [1] был описан метод триангуляции функциональных форм. В данной работе решается прямо противоположная задача, а именно, преобразование полигональных моделей в функциональное представление. Следует отметить, что решение этой задачи на порядок сложнее задачи триангуляции.

Известна работа [2], в которой решалась подобная проблема. В методе предложенном Gary Yngve и Grey Turk полигональные модели сначала преобразовывались в воксельное описание, и уже после вокселизации конвертировались в неявные поверхности итерационно. Такой метод требует огромного количества вычислений (на компьютере 195MHz R10k до 24 часов времени для моделей с приемлемым разрешением и качеством), а также хранения и обработки большого числа данных- $N^3$ .

В работе [3] были представлены патчи произвольных форм на базе аналитических функций возмущения. С какой целью были разработаны такие патчи? Сплайны низких порядков оправдывают свое предназначение только для гладких криволинейных поверхностей, а для сложных нерегулярных поверхностей их применение теряет всякий смысл, поскольку их количество становится соизмеримым с числом треугольников, то есть компактность описания сложных моделей отсутствует, в то время, как вычисления усложняются многократно, в сравнении с обработкой треугольников.

В настоящее время для генерации изображений в реальном времени используются алгоритмы растривания на плоскости и воксельно - базируемые алгоритмы. Основным достоинством первого подхода является низкая вычислительная сложность алгоритмов рендеринга. Для генерации изображения не требуется высоких вычислительных мощностей. К недостаткам можно отнести,

в первую очередь, отсутствие информации о луче, проходящем сквозь объект. Во-вторых, низкую степень детализации отображаемых объектов. Что, в конечном счете, сильно сказывается на качестве получаемых изображений. Рассмотрим подробнее, в чем заключаются эти проблемы. Отсутствие информации о луче делает сложной реализацию 3D эффектов типа облаков, дыма, огня. В настоящий момент все эти эффекты моделируются за счет специфических приемов, типа environment map – bump map или elevation map, недостатками которых являются сильные ограничения на условия их применения. Например, в случае elevation map реализм достигается только при углах зрения близких к прямому. Повышение степени детализации ведет к сильной загрузке транспортных магистралей, и хотя вычислительных мощностей современных процессоров вполне хватает для обработки такого количества полигонов, пропускной способности шины не всегда хватает для передачи данных. Применение текстуры решает проблему реалистичности лишь отчасти, так как вблизи объект все так же представляется как плоскость с наклеенной картинкой. Эффекты типа bump mapping (рельефное текстурирование) немного улучшают картину, но и они обладают существенными ограничениями. В самой простой форме, рельефное текстурирование добавляет реализм к текстурам и объектам, создавая иллюзию рельефности, другими словами, изменений в поверхностной глубине, на плоской поверхности. Таким образом, получаются ячейки на мяче для гольфа, грубая поверхность камня или изрытая кора дерева. В преобразовании выдавливания, для убедительности, глаз должен чувствовать изменения в поверхностной глубине даже притом, что поверхность действительно плоская. На визуальное восприятие глубины влияет количество света, отраженного рассматриваемой поверхностью, а количество света, отраженного в любом данном направлении, зависит от отражающей поверхности. Иными словами, гладкая поверхность всегда отражает больше чем рельефная. В трехмерной графике иллюзия рельефа основана исключительно на цвете освещения/затенения. Эффект освещения/затенения легко достигается модификацией цвета каждого из элементов текстуры, или текстурируемых пикселей, при выводе изображения на экран. Рельефное текстурирование осуществляется следующим образом. Имеется карта рельефа поверхности, как правило, это графический файл, и после вычисления нормали к грани, нормаль отклоняется на некий “возмущающий” вектор, значение которого вычисляется по карте рельефа. Очевидные недостатки такой методики. Хотя поверхность трехмерной модели и кажется рельефной, на краях рельеф отсутствует. Так же, при рельефном текстурировании не учитывается Z координата объекта. В итоге, неровности на объекте, расположенном вблизи, и неровности на удаленном объекте

имеют одинаковый размер. Технология Elevation maps, предложенная компанией Nvidia, по сути, является комбинацией рельефного текстурирования и геометрического моделирования рельефа, учитывающим Z- глубину. Суть метода заключается в рендеринге объекта, представленного как набор параллельных друг другу слоев треугольников, и использовании альфа канала текстуры как высоты данного тексела относительно некоей базовой поверхности, и канала RGB для вычисления нормали к поверхности (здесь и проявляется общность elevation map с bump mapping). Соответственно, если высота пиксела меньше высоты соответствующего пиксела в карте высот, то по значениям RGB канала данного тексела вычисляется нормаль, и затем цвет пиксела. Так как при таком подходе мы имеем текстуру “слитую” с геометрией объекта в одном примитиве то они согласованы по глубине (Z координате) и соответственно лишены вышеописанных недостатков bump mapping. Недостатки же и этого подхода - очевидны. Использование технологии elevation map имеет сильное ограничение на угол зрения. При угле зрения близком к нулевому, будет наблюдаться не объект, а только набор параллельных друг другу плоскостей. Следовательно, отобразить с высокой степенью детализации можно только те поверхности, на которые мы практически всегда будем смотреть под углами, близкими к прямому, например, траву на земле. Реалистичное изображение эффектов типа волн на поверхности воды, облаков тумана, можно получить, используя технологию environment map - bump mapping, являющуюся еще одной модификацией bump mapping. В этом случае, помимо базовой текстуры объекта, применяются еще две текстуры. Текстура, являющаяся отрендеренным вариантом трехмерной сцены вокруг объекта (*environment map*), и текстура - карта рельефа (*bump map*). Самостоятельно и совместно с процедурными текстурами, данная технология позволяет получить такие натуральные эффекты как отражение, отражение в кривом зеркале, дрожание поверхностей, искажение изображения вызываемое водой и теплым воздухом, трансформация искажений по шумовым алгоритмам, имитация туч на небе и другие.

Таким образом, чтобы получить полноценный трехмерный объект со сложной поверхностью, необходимо иметь огромное число треугольников (сотни миллионов, если визуализировать *multiresolution geometric models*), что по числу данных приближается к воксельным моделям. С целью геометрического сжатия таких сложных объектов и были разработаны патчи произвольных форм [3].

Данная работа посвящена автоматическому преобразованию сложных полигональных моделей в функциональное описание с применением патчей произвольных форм, напрямую, без предварительной вокселизации, как это делалось в методе [2].

## 2. ПАТЧИ ПРОИЗВОЛЬНЫХ ФОРМ

В работе [3] были предложены патчи произвольных форм на базе аналитических функций возмущения. Патчи произвольных форм обладают достоинством сплайнового представления поверхностей - высокой степенью гладкости, но главной их отличительной чертой является произвольность формы при небольшом количестве функций возмущения.

Патч произвольной формы есть композиция базового треугольника (рис. 1, слева) и возмущений  $F'(x,y,z) = F(x,y,z) + R(x,y,z)$ , где функция возмущения  $R(x,y,z)$  находится следующим образом:

$$R(x,y,z) = \begin{cases} Q^2(x,y,z), \text{ при } Q(x,y,z) > 0 \\ 0, \text{ при } Q(x,y,z) \leq 0, \end{cases} \quad (1)$$

где  $Q(x,y,z)$  – возмущающая квадратика.

Получающаяся поверхность патча будет высокого порядка (рис. 1, справа), и потребуются небольшое количество функций возмущения для создания сложных форм патчей.

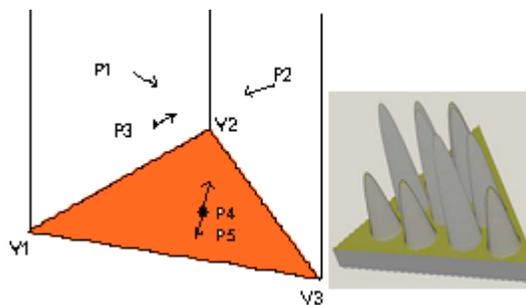


Рис. 1. Слева: клипирующие плоскости треугольника (p1, p2, p3) и базовая плоскость (p4, p5); справа: патч произвольной формы

## 3. АВТОМАТИЧЕСКАЯ КОНВЕРСИЯ

Данное преобразование должно удовлетворять следующим требованиям: автоматической (оптимальной) сегментации, связности, компактности и гладкости. Автоматическая конверсия состоит из следующих основных задач: сегментации (упрощение исходной модели) и вычисления базовых треугольников, построения специального графа (для эффективного анализа вершин треугольных сеток), вычисления функций возмущения, генерации патчей произвольных форм и сопряжения смежных патчей.

### 3.1 Сегментация

Данная задача решается с помощью алгоритма упрощения (симплификации) полигональных моделей (рис. 2 и рис. 3) подобного алгоритму из 3D studio Max. Проблемам симплификации посвящено огромное число работ, авторы одной из наиболее известных работ Paul S. Heckbert и Michael Garland [4].



Рис. 2. Исходная полигональная модель (69451 треугольников)



Рис. 3. Упрощенная модель (1000 базовых треугольников).

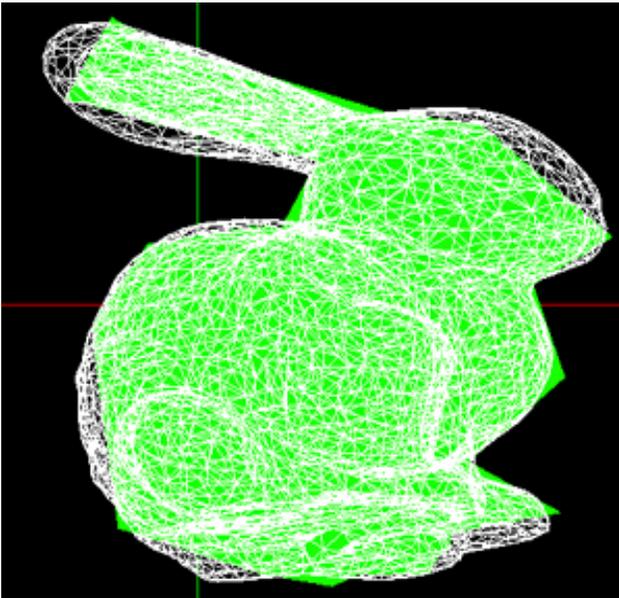


Рис. 4. Исходная и упрощенная полигональные модели

После симплификации мы получаем упрощенную модель, окруженную исходной (самой детальной) полигональной сеткой (рис. 4). Треугольники упрощенной модели далее

используются в качестве базовых для вычисления патчей произвольных форм.

### 3.2 Построение графа

Далее базовые треугольники упрощенной модели переводятся в новый формат как показано на рисунке 1. Каждому базовому треугольнику составляется список вершин исходной детальной сетки, для этого эти вершины проецируются на плоскость базового треугольника, и те вершины, которые к нему относятся остаются в списке, остальные отбраковываются. Базовые треугольники и принадлежащие им вершины показаны на рисунке 5. Вершины в таком списке делятся (рис. 6) на внутренние (проекции которых лежат внутри базового треугольника) и граничные (проекции которых находятся на границах базового треугольника-его ребрах, это необходимо для генерации общих функций возмущения для дальнейшего сопряжения смежных патчей).

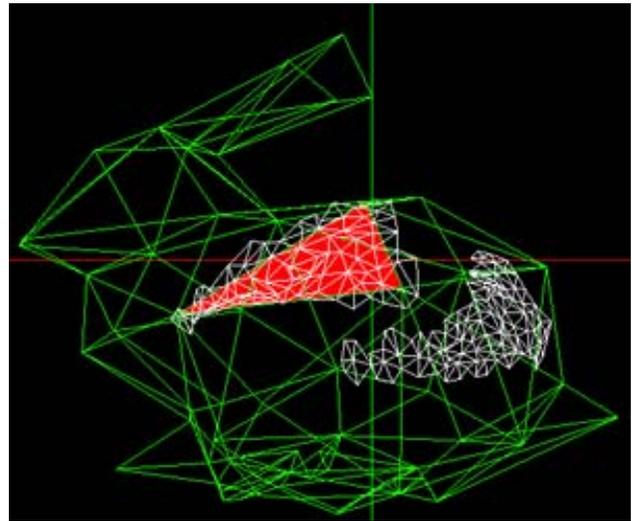


Рис. 5. Детальные треугольные сетки над базовыми треугольниками

### 3.3 Вычисление функций возмущения

Для каждого базового треугольника анализируются вершины полигональной сетки, находят локальные экстремумы.

Действие возмущения  $R$  (1) на неявную функцию  $M(x,y,z)$ , вне зависимости от суммарной степени её многочлена (формируемого, в нашем случае, возмущениями и операциями, гладкости и т.д.), описывается следующим образом:

$$M' = M + fR^2 \quad (2)$$

где  $f$ - коэффициент действия возмущения («фактор»). Имея десять точек полигональной модели  $P_1[x_1, y_1, z_1] \div P_{10}[x_{10}, y_{10}, z_{10}]$ , в которых значение функции  $M$  имеет один и тот же знак (т.е. все десять точек лежат либо внутри, либо снаружи от фигуры, описываемой функцией  $M$ ) ставится задача, прохождения фигуры, описываемой  $M'$ , через эти точки.

Математически это требование описывается как  $M'(P_1[x_1, y_1, z_1] \div P_{10}[x_{10}, y_{10}, z_{10}]) = 0$ . Тогда (2) можно переписать в следующем виде:

$$0 = M' = M + fR^2 \quad (3)$$

$$0 = M + fR^2 \quad (4)$$

$$fR^2 = -M \quad (5)$$

Т.к.  $R^2$  величина положительная, то возьмём

$$f = -\text{sign}(M'(P_1[x_1, y_1, z_1] \div P_{10}[x_{10}, y_{10}, z_{10}])) \quad (6)$$

Следовательно:

$$R^2 = |M|$$

$$R = \sqrt{|M|}$$

(7)

Записываем (7) для  $P_1[x_1, y_1, z_1] \div P_{10}[x_{10}, y_{10}, z_{10}]$ :

$$\left\{ \begin{array}{l} R(P_1) = \sqrt{|M(P_1)|} \\ R(P_2) = \sqrt{|M(P_2)|} \\ \dots \\ R(P_i) = \sqrt{|M(P_i)|} \\ \dots \\ R(P_9) = \sqrt{|M(P_9)|} \end{array} \right\} \quad (8)$$

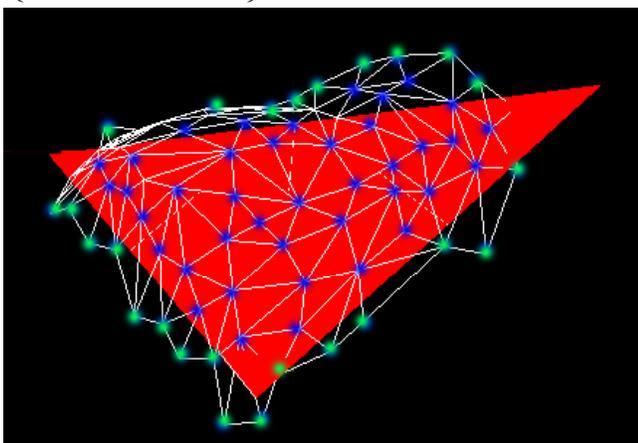


Рис. 6. Граф: детальная треугольная сетка над базовым треугольником

Значение функции 2-го порядка заданной в произвольной точке  $P[x, y, z]$  будет:

$$Q(x, y, z) = q_{xx}x^2 + q_{yy}y^2 + q_{zz}z^2 + q_{xy}xy + q_{xz}xz + q_{yz}yz + q_x x + q_y y + q_z z + q \quad (9)$$

Используя (9) получаем систему уравнений. Решением системы уравнений со свободной членом при заданных  $P_1[x_1, y_1, z_1] \div P_{10}[x_{10}, y_{10}, z_{10}]$  будут десять искомых

коэффициентов функции возмущения  $q_{xx}, q_{yy}, q_{zz}, q_{xy}, q_{xz}, q_{yz}, q_x, q_y, q_z, q$ .

### 3.4 Генерация патчей произвольных форм

Значение функции, описывающей поверхность в виде плоскостей базовых треугольников и возмущений, будет равно нулю на границе, а по мере удаления от границы абсолютное значение функции возрастает. Таким образом, задача подгонки под поверхность сводится к минимизации суммарной функции на наборе точек, представляющих полигональную модель (рис. 6).

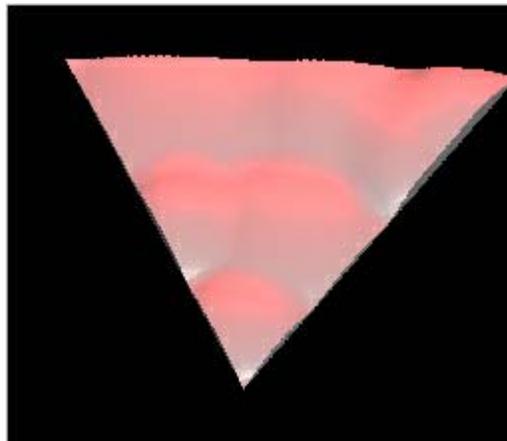


Рис. 7. Патч произвольной формы на базе функций возмущения

Наиболее быструю сходимость минимизации неявной функции обеспечивает метод градиентного спуска, но для рассматриваемого случая необходимо использовать несколько модифицированный метод, т.к. неявная функция, описывающая поверхность, может быть достаточно высоких порядков, а оптимизационная задача сходится к локальному минимуму и на высоких порядках движение по градиенту может перебросить в область другого минимума. Поэтому движение вдоль направления градиента в предложенном методе более плавное, в отличие от оригинального.

Шаг алгоритма:

- I. Добавляем новую квадрику/возмущение.
- II. Оптимизируем функцию, и найдя локальный минимум переходим к шагу I., пока не достигнем требуемого качества.

Используя оптимизационный метод на каждой следующей итерации, имеем значение функционала меньше значения на предыдущем шаге. Оптимизационная задача варьирует шестнадцать коэффициентами квадрики, – т.е. мы спускаемся к минимуму по 16-мерной производной в 16-мерном пространстве. Таким образом, задача генерации патча произвольной формы сводится к задаче деформации базового треугольника (рис. 7).

Замечания:

- 1) базовые треугольники могут быть сферическими для криволинейных поверхностей [3],
- 2) чтобы не было разрывов поверхности на границах смежных патчей генерируются одинаковые функции возмущения.

#### 4. РЕЗУЛЬТАТЫ

На рисунке 8 показана модель Bunny построенная из 5000 патчей произвольных форм (исходных - 69451 треугольников) с коэффициентом сжатия 14, время конверсии - 1 минута на ПК (Intel Pentium 4 Processor, 2800 MHz). Для модели Cow (не показана), состоящей из 1000 патчей произвольных форм (исходных - 34824 треугольников), коэффициент сжатия - 30, время конверсии - 26 секунд.



Рис. 8. Модель Bunny построенная из 5000 патчей произвольных форм



Рис. 9. Пример данных буфера глубины функциональной модели Bunny

Для визуализации был использован алгоритм многоуровневого отслеживания лучей [5, 6, 7].

Для анализа отклонений (отклонение вершин и нормалей) был разработан анализатор критериев отклонений. Анализ степени приближения полученной сетки к функционально заданной поверхности осуществляется по двум метрикам: отклонение вершин треугольной сетки от поверхности функционального объекта; отклонение нормалей к вершинам треугольной сетки от значений нормалей к вершинам на функциональной поверхности.

Для этого мы сначала вычисляем данные буфера глубины двух полигональной и функциональной моделей (рис. 9) в виде двумерного массива данных. Каждый элемент такого

массива представляет собой расстояние от камеры до соответствующей точки на поверхности модели. Затем мы сравниваем все точки этих буферов для вычисления средней разницы. Таким образом мы вычисляем среднее отклонение для соответствующей модели.

Если полигональная и функциональная модели полностью совпадают, то отклонение, соответственно, будет равно нулю. Эти результаты вычислялись в кубе с разрешением 1.0 x 1.0 x 1.0. На рисунках 10 и 11 показаны отклонения функциональных моделей Bunny и Cow от полигональных с разными коэффициентами сжатия (термальная шкала).

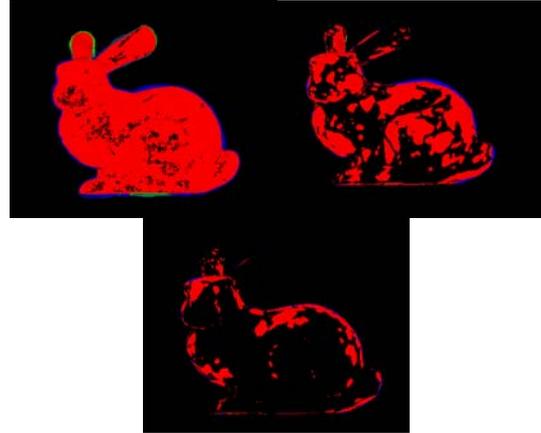


Рис. 10. Отклонения для модели Bunny с разными коэффициентами сжатия (термальная шкала)



Рис. 11. Отклонения для модели Cow с разными коэффициентами сжатия (термальная шкала)

На основании этих данных был построен график зависимости искажений формы объектов от коэффициентов сжатия (рис. 12). По оси X отложено среднее отклонение, по оси Y - коэффициент сжатия. Верхние кривые построены для симплификации тестируемых моделей, из графика хорошо видно резкое увеличение отклонений от увеличения коэффициентов сжатия. Нижние кривые характеризуют автоматическую конверсию полигональных моделей в функциональные на базе патчей произвольных форм. Эти кривые более пологие, это означает то, что отклонения увеличиваются не так сильно при увеличении коэффициента сжатия.

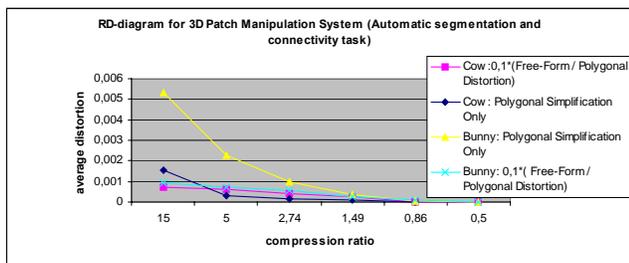


Рис. 12. График зависимости искажений формы объектов от коэффициентов сжатия

## 5. ЗАКЛЮЧЕНИЕ

В данной работе представлен метод автоматического преобразования полигональных моделей в функциональные на базе патчей произвольных форм без предварительной вокселизации. Во-первых, в предлагаемом методе меньше вычислений в отличие от метода [2]. Во-вторых, для хранения и обработки требуется меньшее число данных. В-третьих, благодаря заданию объектов патчами свободных форм достигается высокий коэффициент геометрического сжатия.

В дальнейшем планируется разработать новый подход (обратный обход дерева деления объектного пространства) на базе квадрик для автоматической конверсии полигональных и воксельных моделей в функциональные с применением функций возмущения, с целью сжатия без потерь и с увеличением коэффициента сжатия при меньших искажениях.

увеличением коэффициента сжатия при меньших искажениях.

## 6. БИБЛИОГРАФИЯ

- [1] Шевелев С.В., Вяткин С.И., Долговесов Б.С. Триангуляция функциональных форм// Труды 15-й Междунар. конф. "Графикон-2005". Новосибирск, 2005. <http://www.graphicon.ru/proceedings2005/>
- [2] Gary Yngve, Grey Turk. "Robust Creation of Implicit Surfaces from Polygonal Meshes" // IEEE Transactions on Visualization and Computer Graphics. Vol.8., No.4., October-December 2002.
- [3] Вяткин С.И., Долговесов Б.С. Поверхности и патчи свободных форм на основе скалярных и аналитических функций возмущения// Труды 12-й Междунар. конф. "Графикон-2002". Нижний Новгород, 2002. <http://www.graphicon.ru/2002/papers.htm>
- [4] Paul S. Heckbert and Michail Garland. Optimal Triangulation and Quadric-Based Surface Simplification. Journal of Computational Geometry: Theory and Applications, October 25, 1999.
- [5] S.I. Vyatkin, B.S. Dolgovesov, A.V. Yesin et al. "Geometric Modeling and Visualization of Functionally Defined Objects" // Optoelectronics, Instrumentation and Data Processing. Allerton Press, Inc. USA. - 1999.-N6.
- [6] S.I. Vyatkin, B.S. Dolgovesov, A.V. Yesin, et al. Voxel Volumes volume-oriented visualization system, International

Conference on Shape Modeling and Applications (March 1-4, 1999, Aizu-Wakamatsu, Japan) IEEE Computer Society, Los Alamitos, California, 1999, pp. 234-241.

[7] S.I.Vyatkin, B.S.Dolgovesov, A.V.Yesin, A.A.Zhigach, S.E.Chizhik, and R.A.Shcherbakov. "Geometric Modeling and Visualization of Functionally Defined Objects" Computer Graphics and Geometry Internet-Journal Vol. 3., N. 2. <http://www.cgg.ru/>

## Об авторах

Сергей Иванович Вяткин – к.т.н, ст.н.с. Лаборатории синтезирующих систем визуализации Института автоматизации и электротехники СО РАН.

Адрес: Новосибирск, 630090, пр-т Коптюга, 1, ИАиЭ СО РАН.

Телефон: (383) 333-36-30

E-mail: [sivser@mail.ru](mailto:sivser@mail.ru)

Борис Степанович Долговесов – к.т.н, зав. Лабораторией синтезирующих систем визуализации Института автоматизации и электротехники СО РАН.

Адрес: Новосибирск, 630090, пр-т Коптюга, 1, ИАиЭ СО РАН.

Телефон: (383) 333-36-30

E-mail: [bsd@iae.nsk.su](mailto:bsd@iae.nsk.su)

Алексей Владимирович Рухлинский – аспирант физического факультета Новосибирского Государственного Университета

Адрес: Новосибирск, 630090, пр-т Коптюга, 1, ИАиЭ СО РАН.

Телефон: (383) 333-36-30

E-mail: [ares@iae.nsk.su](mailto:ares@iae.nsk.su)

## Creating Complex Surfaces Based on Perturbation Functions from Polygonal Meshes

### Abstract

We introduce a practical method for creating surfaces based on perturbation functions from polygonal models that produces high-quality results for complex models. Whereas much previous work has been done with primitives such as deformed quadrics, we use surfaces based on the patches of arbitrary forms. We have applied this method to a variety of polygonal meshes.

**Keywords:** Geometric modeling, patches of arbitrary forms, perturbation functions.

### About the author(s)

Sergei I. Vyatkin (Ph.D.) is a senior scientific researcher of Synthesizing Visualization Systems Laboratory at Institute of Automation and Electrometry SB RAS.

His contact email is [sivser@mail.ru](mailto:sivser@mail.ru).

Boris S. Dolgovesov (Ph.D.) is a head of Synthesizing Visualization Systems Laboratory at Institute of Automation and Electrometry SB RAS.

His contact email is [bsd@iae.nsk.su](mailto:bsd@iae.nsk.su)

Aleksey V. Ruchlinskiy is a post-graduate student at Novosibirsk State University, Department of Physics. His contact email is [ares@iae.nsk.su](mailto:ares@iae.nsk.su)