

Программные системы и вычислительные методы

Правильная ссылка на статью:

Вяткин С.И., Долговесов Б.С. — Прямой рендеринг трехмерных объектов на основе функций возмущения с использованием графических процессоров // Программные системы и вычислительные методы. – 2023. – № 1.
DOI: 10.7256/2454-0714.2023.1.38263 EDN: IWRNCU URL: https://nbpublish.com/library_read_article.php?id=38263

Прямой рендеринг трехмерных объектов на основе функций возмущения с использованием графических процессоров

Вяткин Сергей Иванович

кандидат технических наук

старший научный сотрудник, Институт автоматизации и электрометрии СО РАН

630090, Россия, г. Новосибирск, ул. Коптюга, 1

✉ sivser@mail.ru



Долговесов Борис Степанович

ORCID: 0000-0002-6255-9315

кандидат технических наук

Заведующий Лабораторией, Институт автоматизации и электрометрии СО РАН

630090, Россия, г. Новосибирск, ул. Ак. Коптюга, 1, каб. 318

✉ bsd@iae.nsk.su



[Статья из рубрики "Автоматизация проектирования и технологической подготовки производства"](#)

DOI:

10.7256/2454-0714.2023.1.38263

EDN:

IWRNCU

Дата направления статьи в редакцию:

13-06-2022

Аннотация: Объектом исследования является метод прямого рендеринга сложных трехмерных объектов на основе функций возмущения с применением графических процессоров, с использованием множества потоковых мультипроцессоров. Прямой рендеринг означает, что визуализация функционально заданных моделей происходит без их предварительного преобразования в другие форматы, например, в сетки треугольников. Метод исследования базируется на аналитической геометрии в пространстве, дифференциальной геометрии, теории интерполяции и теории матриц, опирается на математическое моделирование и теорию вычислительных систем. Основными выводами проведенного исследования являются: возможность прямого

рендеринга функционально заданных объектов, при рендеринге важно, чтобы вычислительные процессоры не простаивали. Первая проблема, которая была решена, заключалась в том, что у разных графических процессоров разное количество потоковых мультипроцессоров. Поэтому необходимо было выбрать во время исполнения оптимальную стадию, с которой начиналась работа. Таким образом, можно частично избавиться от проблемы с неиспользуемыми вычислительными ресурсами. Вторая проблема - задача балансировки, была решена с помощью использования большого количества вычислительных процессоров. Для реализации была использована модель параллельного программирования CUDA, которая вместе с набором программных средств позволяет реализовывать программы на языке C для исполнения на графическом процессоре. Полученная в результате система визуализирует сложные функционально заданные объекты с высоким разрешением в интерактивном режиме. Исследована зависимость производительности от вычислительной мощности графических процессоров.

Ключевые слова:

функционально заданный объект, функции возмущения, конструктивная твердотельная геометрия, прямой рендеринг, графический процессор, потоковые мультипроцессоры, параллельные вычисления, модель параллельного программирования, иерархия групп потоков, ускорение вычислений

1. Введение

Функционально заданные поверхности - это представление замкнутых многообразий для моделирования и визуализации. Функциональные представления являются способом представления твердотельных моделей [\[1, 2\]](#). По сравнению с треугольными сетками и NURBS, они обеспечивают простые операции с CSG. В полигональные модели сложно внести атрибутивную информацию, а алгоритмы визуализации выполнения топологических операций достаточно трудоемки. Полигональные модели позволяют визуализировать только внешний слой объекта, в то время, как для многих приложений необходима внутреннее строение. Функциональные представления обеспечивают однозначную проверку внутри и снаружи объектов, и произвольное разрешение [\[3, 4\]](#). Основная задача твердотельного моделирования состоит в том, чтобы знать, находится ли точка внутри или снаружи объекта [\[5-8\]](#). Также важно знать, находится ли точка в пределах допуска к границе.

Для моделей с представлениями границ (B-гер) ответ на этот вопрос означает проверку точки по каждой части топологии в модели [\[9-12\]](#). Общий подход заключается в создании луча, который начинается с рассматриваемой точки и движется в любом направлении. Если число пересечений нечетное, точка находится внутри объекта, в противном случае она находится снаружи. Такое вычисление является дорогостоящим и подверженным ошибкам, так как грани могут быть пропущены или пересчитаны дважды, если они нанесены вблизи их краев или вблизи касательной к краям или граням. Для моделей B-гер и сетки – это одна из основных проблем.

При моделировании функционально заданных поверхностей используют информацию о знаках для обозначения внутреннего и внешнего региона, поэтому вычисления являются простыми [\[13\]](#). Значения, близкие к нулю, находятся ближе к границе, поэтому можно

установить, находится ли точка в пределах допуска к границе. Как с точки зрения точности, так и с точки зрения производительности, использование функционально заданных поверхностей лучше подходит для такой задачи.

Визуализировать функционально заданные не простая задача. В работе [\[14\]](#) описан метод визуализации поверхностей, заданных алгебраически полиномами высокой степени. При этом моделировать объекты с помощью полиномов сложно, не гарантируется то, насколько точно начальная функция будет приближена к кривой Безье. Кроме этого перевод объекта в другую систему координат – не простая задача, поэтому создание динамических сцен проблематично. Метод визуализации аналитически заданных объектов [\[15\]](#), основанный на не постоянном размере шага, вычисляет сферу с центром в текущей точке на луче, однако нахождение подходящего радиуса сферы не простая задача. Кроме того, в алгоритме использована предварительная обработка, поэтому, как и в предыдущем методе, визуализация объектов, которые изменяют свою форму и положение во времени требует больших вычислений.

Для быстрой визуализации обычно применяется преобразование функционально заданных поверхностей в сетки треугольников.

Однако сделать надежными такие преобразования очень сложно. Например, для топологических пространств, локально сходных с евклидовым, а для обнаружения тонких объектов требуется высокое разрешение выборки. Поэтому разработка метода прямого рендеринга функционально заданных объектов является актуальной задачей.

В нашей работе одной из главных задач является эффективное нахождение первого пересечения луча с поверхностью. Задание объектов с использованием функций возмущения позволяет эффективно осуществлять поиск точек поверхности. Для вычисления пересечения лучей с поверхностями трехмерных объектов предлагается метод, в котором отсутствуют недостатки, характерные для известных подходов. Уменьшение времени на визуализацию достигается за счёт эффективного использования вычислительных ресурсов графического процессора с архитектурой CUDA.

Целью данной работы является разработка метода прямого рендеринга 3D объектов на основе функций возмущения с применением графических процессоров.

2. Прямой рендеринг моделей на основе функций возмущения

Задание моделей с использованием функций возмущения описано в работе [\[13\]](#).

Для реализации алгоритма рендеринга на графическом процессоре (GPU), учитывалась его архитектура. Например, условные переходы осуществлять в такой архитектуре очень дорого, как с точки зрения времени выполнения операций, так и эффективности вычислений. Для графического процессора эффективно сразу "зарядить" его данными, чтобы он начал выполнять параллельные вычисления, иначе параллельные вычисления не эффективны и требуется синхронизация. Кроме этого графические процессоры имеют разное количество потоковых мультипроцессоров, поэтому необходимо учитывать оптимальную стадию с которой надо начинать работать. Благодаря таким мерам можно лишь частично решить проблему неиспользуемых вычислительных ресурсов, поскольку существует еще задача балансировки.

Рассмотрим сцену, находящуюся в единичном трёхмерном кубе. Наблюдатель смотрит на куб вдоль оси Z. С точки зрения наблюдателя исходят лучи такие, что каждый луч соответствует пикселю на изображении. Каждый из лучей делится вдоль оси Z. Таким

образом, вычисляем функцию плотности вдоль луча, которая зависит от одной переменной. Задача состоит в нахождении первой точки, в которой функция обращается в ноль. Поскольку мы имеем аналитически заданную функцию плотности, то это позволяет эффективно проводить поиск точки пересечения луча с поверхностью. После вычисления точки пересечения с поверхностью каждого луча, определяем глубину кадра. Затем в каждом пикселе вычисляем нормаль. Используя данные глубины и нормали в каждом пикселе, используем модель локального освещения. В результате получаем изображение гладкого объекта с учётом освещения.

Для реализации была использована архитектура CUDA). Это позволило использовать большое количество вычислительных процессоров, которые одновременно вычисляют несколько лучей. Также учитывалось влияние скорости работы с памятью. Для ускорения вычислений максимально использованы регистры, так как это самый быстрый вид доступной памяти. Следующая по скорости - это совместно используемая память. Во всех остальных случаях использовалась общая память графического процессора.

В результате было реализовано приложение, с помощью которого можно осуществлять прямой рендеринг функционально заданных моделей. С помощью графического процессора вычисляются глубина кадра, нормали и освещение. В функции центрального процессора входят геометрические преобразования. Для отображения изображения использовалась DirectX. Тестирование производилось на процессоре Intel Core2 CPU E8400 3.0 GHz, GPU 9800 GT и 470 GTX. В таблице 1 приведены время вычислений, количество операций в секунду и ускорение.

Табл. 1. Время вычислений, количество кадров в секунду и ускорение

		0	1	2	3	4	6	7
Время	9800 GT	0,1164	0,39797	0,68046	0,3689	0,17703	0,07047	0,09109
	470 GTX	0,0178	0,06266	0,07235	0,04251	0,02109	0,01203	0,01125
	E8400	0,9015	3,17578	3,27562	2,0336	0,80265	0,59422	0,42344
Кол-во кадров в сек.	9800 GT	42,955	12,5637	7,34797	13,5538	28,2438	70,9521	54,8907
	470 GTX	280,58	79,7957	69,1085	117,619	237,079	415,627	444,444
	E8400	5,5459	1,57441	1,52642	2,45869	6,22936	8,41439	11,8080
Ускорение	9800 GT	7,7453	7,97994	4,81383	5,51260	4,53397	8,43224	4,64858
	470 GTX	50,592	50,6827	45,27464	47,83816	38,05832	49,39485	37,6391

На рисунке 1 показана диаграмма: по оси абсцисс отображен номер теста, по оси ординат – количество кадров в секунду для разных тестов. На рисунке 2 показано среднее время на кадр для разных тестов. На рисунке 3 – ускорение относительно E8400.

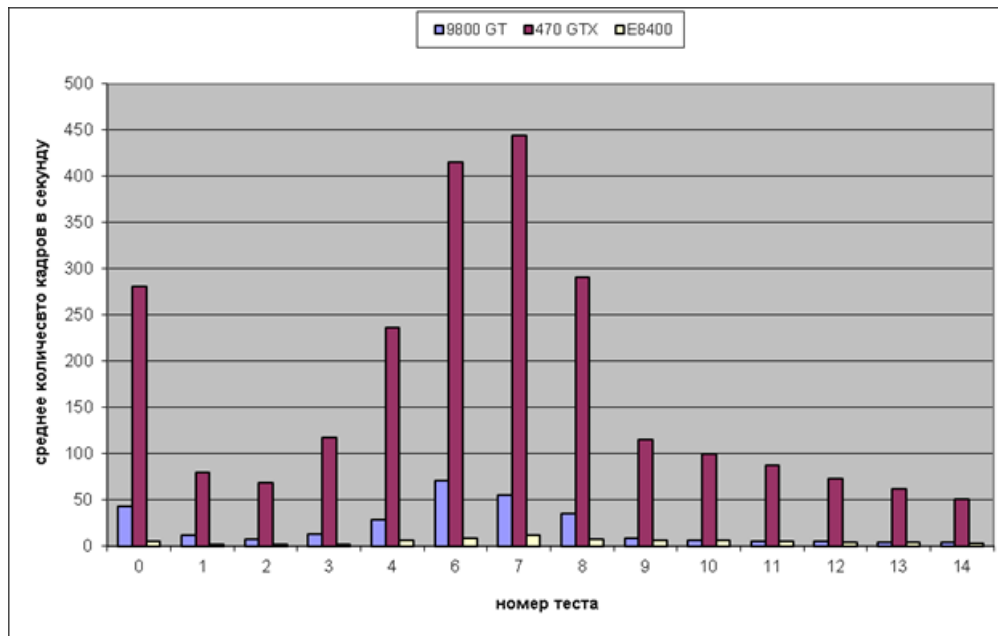


Рис. 1. Количество кадров в секунду для разных тестов

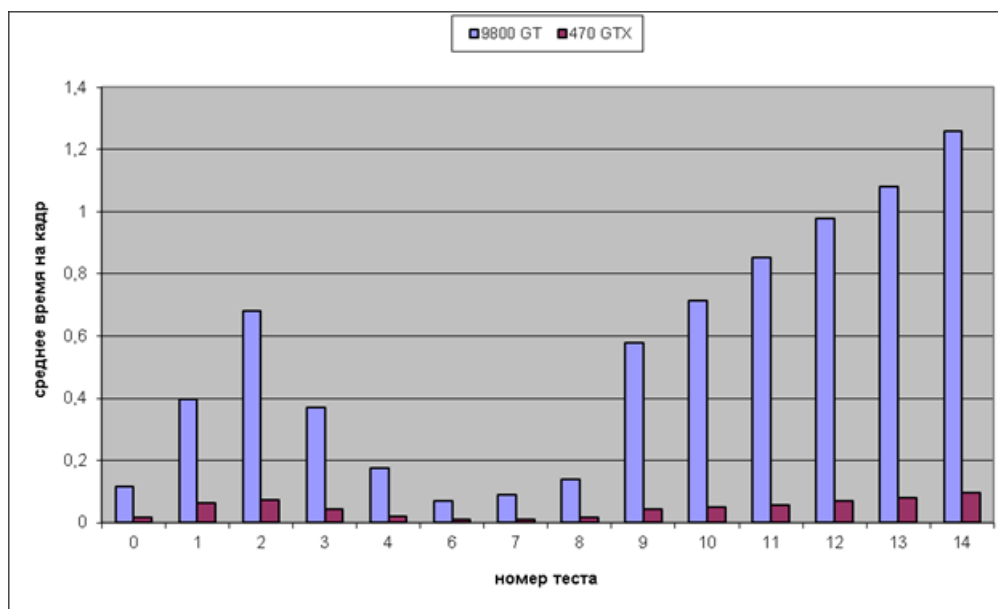


Рис. 2. Среднее время кадра для разных тестов

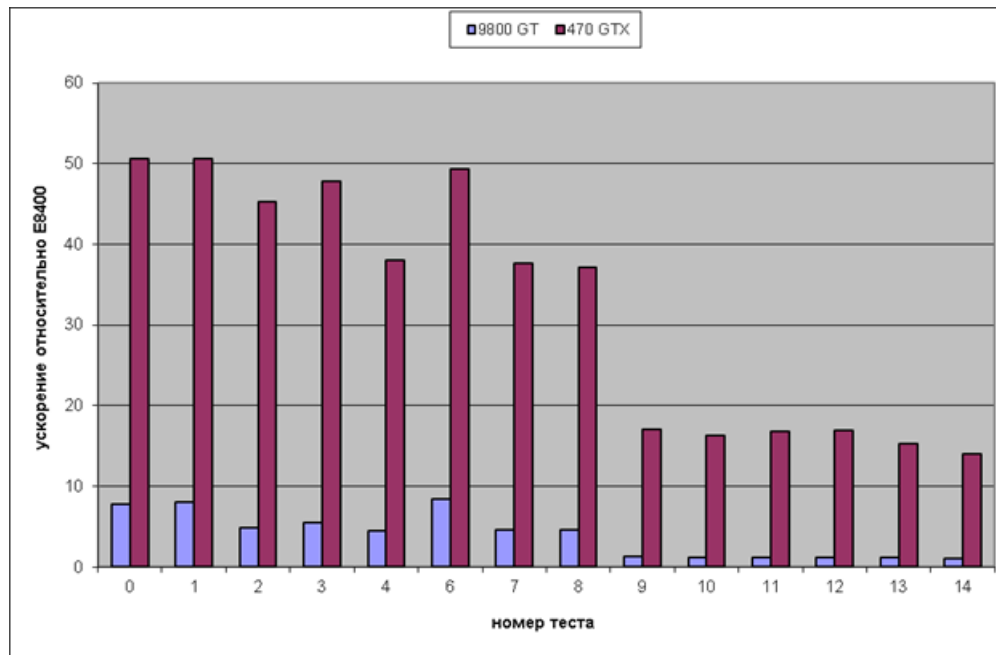


Рис. 3. Ускорение относительно центрального процессора E8400 для разных тестов

Мы сравнили предлагаемый метод прямого рендеринга с известными подходами.

Так в работе [14] описан метод рендеринга алгебраически заданных поверхностей. Однако разновидность поверхностей, которые можно описать с помощью функций возмущения намного шире, чем алгебраически заданными поверхностями. Кроме этого метод [14] не гарантирует то, насколько точно начальная функция приближена к кривой Безье. Также недостатком этого метода является то, что не просто преобразовать объект в другую систему координат. Поэтому создание динамических сцен проблематично.

В работе [15] описан метод с переменным размером шага на луче. Однако недостатком является нахождение подходящего радиуса, так как он эффективен только для статичных сцен. Для этого используются предварительные вычисления. Таким образом этот метод можно применять только тогда, когда объекты статичны. Поэтому, как и в предыдущем методе, рендеринг объектов, которые изменяют свою форму и положение во времени, не эффективно.

Заключение

Предложенные способ задания функциональных объектов и метод прямого рендеринга имеют преимущества перед известными подходами.

К основным достоинствам предлагаемых способа задания объектов на основе функций возмущения и метода прямого рендеринга относятся: простота и эффективность вычисления точек пересечения поверхности с лучами; компактное описание криволинейных объектов (задание объектов функционально заданными поверхностями сокращает в 100 и более раз описание базы данных по сравнению с описанием их полигонами); простота анимации и деформации поверхностей.

Функциональное задание объектов особенно актуально в ряде задач компьютерной графики, включая моделирование мягких или органических объектов, трехмерного морфинга, определения столкновений объектов и конструктивной твердотельной геометрии. Области применения функционально заданных объектов: молекулярная биология, интерактивные графические системы визуализации, CAD-системы, системы 3D-

моделирования, 3D веб-визуализация, аддитивное производство и т. д.

Библиография

1. Sigg, C. Representation and rendering of implicit surfaces. Diss. ETH No. 16664, Dipl. Rechn. Wiss. ETH Zurich, Switzerland, 2006, pages 162. DOI:10.3929/ETHZ-A-005267203 Corpus ID: 124349594
2. Dekkers, D., Overveld, K.V. Golsteijn, R. Combining CSG modeling with soft blending using Lipschitz-based implicit surfaces. *Visual Computer*, vol. 20, no. 6, 2004. pp. 380–391.
3. Vyatkin, S. An Interactive System for Modeling, Animating and Rendering of Functionally Defined Objects. *American Journal of Computer Science and Engineering Survey*, 2014, vol. 2, no. 3. pp. 102-108.
4. Vyatkin, S. Perturbation functions for compact database. *Review of computer engineering research*. 2017. vol. 4., no. 1. pp. 30–37. doi: 10.18488/journal.76.2017.41.30.37
5. Farin, G., Hoschek, J., Kim, M.S. *Handbook of Computer Aided Geometric Design*. [electronic resource]. 2002 Elsevier. ISBN 978-0-444-51104-1.
6. Pottmann, H., Brell-Cokcan, S., Wallner, J. Discrete Surfaces for Architectural Design. Archived 2009-08-12 at the Wayback Machine, pp. 213–234 in *Curve and Surface Design*, Patrick Chenin, Tom Lyche and Larry L. Schumaker (eds.), 2007. Nashboro Press, ISBN 978-0-9728482-7-5.
7. Farin, G. *Curves and Surfaces for CAGD. A Practical Guide*, Morgan-Kaufmann, 2002 ISBN 1-55860-737-4.
8. Sturm, T. An Algebraic Approach to Offsetting and Blending of Solids in Computer Algebra in Scientific Computing. CASC 2000, V.G. Ganzha, E.W. Mayr and E.V. Vorozhtsov (Eds.), Springer-Verlag, Berlin (2000), pp. 367-381.
9. Stroud, I. *Boundary Representation Modelling Techniques*. Publisher: Springer, January 2006, ISBN: 978-1-84628-312-3 DOI:10.1007/978-1-84628-616-2
10. Kainz, W., Neufeld, E., Bolch, W.E., Graff, C.G. Advances in Computational Human Phantoms and Their Applications in Biomedical Engineering-A Topical Review. *IEEE Transactions on Radiation and Plasma Medical Sciences* PP(99):1-1. December 2018 DOI:10.1109/TRPMS.2018.2883437
11. Arioli, C., Shamanskiy, A., Klinkel, S., Simeon, B. Scaled boundary parametrizations in isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering* 349, March 2019, DOI:10.1016/j.cma.2019.02.022
12. Leidinger, L.F, Breitenberger, M., Bauer, A.M., Hartmann, S. Explicit dynamic isogeometric B-Rep Analysis of penalty-coupled trimmed NURBS shells. *Computer Methods in Applied Mechanics and Engineering* 351 April 2019, DOI:10.1016/j.cma.2019.04.016
13. Vyatkin, S.I. Complex Surface Modeling Using Perturbation Functions. *Optoelectronics, Instrumentation and Data Processing*, vol. 43, no. 3, 2007. pp. 40-47.
14. Reimers, M., Seland, J. Ray Casting Algebraic Surfaces using the Frustum Form. *Eurographics*, vol. 27 (2008), no. 2, pp. 361-370.
15. Liktov, G. Ray Tracing Implicit Surfaces on the GPU. *Computer Graphics and Geometry*, vol. 10, no. 3, 2008, pp. 36-53. <http://www.cgj-journal.com/2008-3/04.htm>

Результаты процедуры рецензирования статьи

В связи с политикой двойного слепого рецензирования личность рецензента не раскрывается.

Со списком рецензентов издательства можно ознакомиться [здесь](#).

Предмет исследования – разработка метода прямого рендеринга 3D-объектов на основе функций возмущения с применением графических процессоров.

Методология исследования основана на сочетании теоретического и эмпирического подходов с применением методов анализа, моделирования, численного эксперимента, обобщения, сравнения, синтеза.

Актуальность исследования определяется широким распространением технологий компьютерной графики, необходимостью разработки соответствующих программных систем и вычислительных методов, включая рендеринг трёхмерных объектов с использованием графических процессоров.

Научная новизна связана с предложенным автором способом задания функциональных объектов и метод прямого рендеринга, который имеет ряд преимуществ перед известными аналогами (простота и эффективность вычисления точек пересечения поверхности с лучами, компактное описание криволинейных объектов, простота анимации и деформации поверхностей).

Статья написана русским литературным языком. Стиль изложения научный.

Структура рукописи включает следующие разделы: 1. Введение (функционально заданные поверхности, представление замкнутых многообразий для моделирования и визуализации, твердотельные модели, треугольные сетки и NURBS, простые операции с CSG, полигональные модели, модели с представлениями границ (B-рег), задача визуализации, метод визуализации поверхностей, заданных алгебраически полиномами высокой степени, метод визуализации, основанный на не постоянном размере шага, преобразование функционально заданных поверхностей в сетки треугольников, разработка метода прямого рендеринга функционально заданных объектов, эффективное нахождение первого пересечения луча с поверхностью), 2. Прямой рендеринг моделей на основе функций возмущения (задание моделей с использованием функций возмущения, реализация алгоритма рендеринга на графическом процессоре (GPU), сцена в единичном трёхмерном кубе, получение изображения гладкого объекта с учётом освещения, архитектура CUDA, приложение для прямого рендеринга функционально заданных моделей (вычисляются глубина кадра, нормали и освещение), отображение изображения DirectX, тестирование на процессоре Intel Core2 CPU E8400 3.0 GHz, GPU 9800 GT и 470 GTX, количество кадров в секунду, среднее время кадра и ускорение (относительно центрального процессора E8400) для разных тестов, сравнение предлагаемого метода прямого рендеринга с известными подходами – метод рендеринга алгебраически заданных поверхностей, метод с переменным размером шага на луче), Заключение (выводы), Библиография.

Разделы «Введение», «Заключение» следует единообразно пронумеровать либо оставить без нумерации. Дублирование данных в таблице и на рисунках не целесообразно, что следует исправить.

Текст включает одну таблицу, три рисунка. Дублирование данных в таблице и на рисунках не целесообразно, что следует исправить. Для представленных величин нужно указать размерность.

Содержание в целом соответствует названию. Встречающиеся в тексте аббревиатуры (CSG, CUDA и др.) при первом упоминании следует привести полностью.

Библиография включает 15 источников отечественных и зарубежных авторов – монографии, научные статьи, Интернет-ресурсы и пр. Библиографические описания некоторых источников требуют корректировки в соответствии с ГОСТ и требованиями редакции, например:

1. Sigg C. Representation and rendering of implicit surfaces. Zurich, Switzerland : Dipl. Rechn. Wiss. ETH, 2006. 162 p.
2. Dekkers D., Overveld K. V., Golsteijn R. Combining CSG modeling with soft blending using Lipschitz-based implicit surfaces // Visual Computer. 2004. Vol. 20. № 6. P. 380–391.
5. Farin G., Hoschek J., Kim M.S. Handbook of Computer Aided Geometric Design. Место издания ??? : Elsevier, 2002. ??? p.

Возможно излишнее самоцитирование (Vyatkin S.I. с соавторами).

Апелляция к оппонентам (Sigg C., Dekkers D., Overveld K. V., Golsteijn R., Farin G., Hoschek J., Kim M. S., Pottmann H., Brell-Cokcan S., Wallner J., Farin G., Sturm T., Stroud I., Kainz W., Neufeld E., Bolch W.E., Graff C. G., Arioli C., Shamanskiy A., Klinkel S., Simeon B., Leidinger L. F, Breitenberger M., Bauer A. M., Hartmann S., Reimers M., Seland J., Liktov G. и др.) имеет место.

В целом материал представляет интерес для читательской аудитории и после доработки может быть опубликован в журнале «Программные системы и вычислительные методы».