

На правах рукописи

ДМИТРИЕВ Кирилл Александрович

Разработка и модернизация методов генерации физически аккуратных изображений на ЭВМ

Специальность 05.13.11 — математическое и программное обеспечение
вычислительных машин, комплексов и компьютерных сетей

А в т о р е ф е р а т
диссертации на соискание ученой степени
кандидата физико-математических наук

Москва — 2002 год

Работа выполнена в Институте прикладной математики
им. М.В. Келдыша РАН.

Научный руководитель – кандидат физико-математических наук, ст.науч.сотр.
Юрий Матвеевич Баяковский.

Научный консультант – кандидат физико-математических наук, ст.науч.сотр.
Владимир Александрович Галактионов.

Официальные оппоненты:

- доктор физико-математических наук, профессор
Евгений Викторович Шикин,
- кандидат физико-математических наук, ст.науч.сотр.
Михаил Михайлович Комаров.

Ведущая организация:

Институт вычислительной математики и математической геофизики СО РАН.

Защита состоится «____» _____ 2002 г. в ____ час. на заседании Дис-
сертационного совета Д 002.024.01 в Институте прикладной математики им.
М.В. Келдыша РАН по адресу: 125047, Москва, Миусская пл., 4.

С диссертацией можно ознакомиться в библиотеке Института прикладной
математики им. М.В.Келдыша РАН.

Автореферат разослан «____» _____ 2002 г.

Ученый секретарь диссертационного совета
доктор физико-математических наук

Т.А.Полилова

Общая характеристика работы

Актуальность работы

Задача разработки программных средств автоматической генерации изображений по заданным исходным данным является весьма актуальной в настоящее время. Человеко-машинные интерфейсы систем автоматического проектирования, баз данных и знаний, игровых приложений и т.д., часто реализуются посредством визуализации некоторых объектов (природа которых зависит от конкретного приложения). Хотя еще совсем недавно мощности ЭВМ хватало лишь для схематичной визуализации объектов, бурное развитие компьютерной техники сделало возможным генерацию высокореалистичных изображений. Это не только позволило улучшить существующие приложения, но и открыло ряд новых применений ЭВМ.

Для реалистичной визуализации необходимо моделировать взаимодействие света с объектами, что является вычислительно сложной задачей. Даже мощности современных ЭВМ часто не достаточно для решения ее с приемлемой скоростью. Поэтому иногда для повышения эффективности прибегают к различным эвристическим подходам, не претендующим на физическую аккуратность. Хотя это позволяет значительно повысить скорость, для достижения реалистичности обычно необходима трудоемкая настройка параметров алгоритма для конкретной сцены или коррекция полученного изображения вручную.

Более перспективным направлением представляется повышение эффективности за счет использования более совершенных методик построения физически аккуратных изображений. Это направление является одним из приоритетных в мировой индустрии компьютерной графики. Программные модули физически аккуратного моделирования включены в последние версии таких известных программ визуализации как 3D Studio MAX, 3D Studio VIZ, RenderMan.

Одним из перспективных способов повышения скорости генерации изображений является использование метода квази-Монте-Карло вместо традиционно используемого Монте-Карло. Теоретическое преимущество метода квази-

Монте-Карло давно обосновано исследователями в области мат. моделирования. Тем не менее, остается ряд теоретических и практических трудностей применения этого метода в программных средствах визуализации и виртуальной реальности.

Цель работы

Обоснование целесообразности применения метода квази-Монте-Карло в алгоритмах генерации изображений. Оптимизация существующих генераторов квазислучайных последовательностей точек для алгоритмов трассировки фотонов. Реализация алгоритма квази-Монте-Карло генерации физически аккуратных изображений. Разработка метода оценки погрешности реализованного алгоритма. Изучение возможности модификации реализованного алгоритма для эффективной генерации интерактивных последовательностей изображений.

Научная новизна работы

Предложены новые эффективные способы генерации квазислучайных последовательностей Холтона и $ЛП_{\tau}$, оптимизированные для алгоритмов трассировки фотонов. Предложены и обоснованы ряд новых рекомендаций по использованию квазислучайных последовательностей точек в алгоритмах генерации изображений. Разработана новая методика оценки погрешности алгоритма квази-Монте-Карло и дано ее экспериментальное обоснование. Предложена новая методика использования свойств последовательности Холтона для нахождения фотонов с близкими траекториями, что дает возможность концентрировать вычисления в нужном районе сцены.

Практическая значимость

Разработаны и обоснованы рекомендации по переводу существующих программных средств визуализации, основанных на методе Монте-Карло, на использование квазислучайных последовательностей точек. Рассмотрен и решен ряд практических вопросов, специфичных для данной прикладной области, а именно:

- оптимизация генераторов квазислучайных точек для алгоритмов трассировки фотонов на современных ЭВМ;

- необходимые изменения в алгоритмах моделирования рассеивающих свойств поверхностей и источников света общего вида;
- необходимые изменения в алгоритме оценки погрешности;

Разработан и реализован новый алгоритм генерации интерактивных последовательностей изображений. Предложенный алгоритм может использоваться для достижения большей реалистичности в интерактивных приложениях, таких как прикладные пакеты дизайна освещения, компьютерные игры и приложения виртуальной реальности.

Апробация работы и публикации

Основные результаты диссертации представлены в публикациях [1, 2, 3, 4, 5, 6]. Все предлагаемые алгоритмы генерации статических изображений внедрены в прикладной пакет трехмерного моделирования. Алгоритм для генерации интерактивных последовательностей изображений реализован в виде отдельной программы и представлен на конференции «Eurographics Workshop on Rendering'2002, Pisa, Italy» [4].

Структура и объем работы

Диссертация состоит из введения, четырех глав, заключения, списка литературы и приложений. Содержание работы изложено на 129 страницах (включая 3 страницы приложений). Список литературы включает 68 наименований. В работе содержится 20 рисунков и 3 таблицы.

Содержание работы

Во введении обосновывается актуальность и практическая значимость темы диссертации. Дается краткий обзор ранее выполненных в этой области работ, излагается краткое содержание следующих глав.

Первая глава посвящена обоснованию целесообразности применения методов квази-Монте-Карло на широком классе задач и формулированию проблем их использования в задачах генерации изображений. Сначала кратко описана модель исходных данных, которую используют предлагаемые в работе алгоритмы. Рассмотрены четыре вида исходных данных: геометрия сцены, функция яркости источников света,ДФРС (Двулучевые Функции Рассеивающей Способности) поверхностей и функции параметров наблюдения.

Для построения изображения необходимо найти яркость каждой видимой точки поверхности по направлению к наблюдателю. Эта яркость может быть найдена из известного уравнения рендеринга:

$$L(x, w) = L_e(x, w) + \tau(L(x, w)) \quad (1)$$

где $L(x, w)$ - искомая функция яркости, $L_e(x, w)$ - функция яркости источников света, а τ - интегральный оператор, зависящий от геометрии и ДФРС поверхностей сцены. Уравнение (1) является известным уравнением Фредгольма второго рода. Его решение сводится к вычислению суммы интегрального ряда. Так как размерность интегралов этого ряда в общем случае не ограничена, использование классических квадратурных формул (напр., прямоугольников, Симпсона) является неэффективным. Поэтому традиционным выбором для решения этой задачи является метод Монте-Карло, скорость сходимости которого не зависит от размерности интегрирования.

Простейший метод интегрирования заключается в оценке среднего в области интегрирования значения интегрируемой функции. Искомый интеграл оценивается как произведение найденного среднего значения и объема области интегрирования. Для оценки среднего значения функции в методе Монте-Карло вычисляют среднее арифметическое значений функции в случайно выбранных точках. Известно, что погрешность интегрирования Монте-Карло подчиняется закону:

$$M(|E_N|) = O\left(\frac{1}{\sqrt{N}}\right) \quad (2)$$

где N - количество испытаний, E_N - погрешность оценки интеграла по N испытаниям, а $M(\cdot)$ - символ математического ожидания.

Альтернативным методом интегрирования многомерных функций является метод квази-Монте-Карло. Его отличие от метода Монте-Карло состоит в использовании заведомо неслучайных, но гораздо более равномерных, последовательностей точек. Такие последовательности точек называются квазислучайными. Пусть P - квазислучайная последовательность точек $q_i (i \geq 0)$ и интегрирование ведется в единичном интервале I^s размерности s . Известная теорема Коксмы-Хлавки описывает сходимость квази-Монте-Карло на некоторых классах функций:

$$\left| \frac{1}{N} \sum_{i=0}^{N-1} g(q_i) - \int_{I^s} g(x) dx \right| \leq T(g(x)) D_N^*(P) \quad (3)$$

где $T(g(x))$ - вариация $g(x)$ в I^s , а $D_N^*(P)$ - отклонение последовательности P , характеризующее ее равномерность. В настоящее время построены квазислучайные последовательности точек с отклонением:

$$D_N^*(P) = O\left(\frac{(\log N)^s}{N}\right) \quad (4)$$

Таким образом, сходимость интегрирования квази-Монте-Карло (3) на функциях с ограниченной вариацией гораздо лучше, чем Монте-Карло (2).

Очень часто функция яркости в уравнении рендеринга (1) не обладает ограниченной вариацией. К этому приводят разрывы в ней, не располагающиеся в плоскостях, параллельных координатным. Но и в этом случае известно, что погрешность интегрирования квази-Монте-Карло со сколь угодно большой вероятностью оказывается по крайней мере не хуже, чем погрешность Монте-Карло.

Таким образом, применение методов квази-Монте-Карло для решения уравнения рендеринга целесообразно. Тем не менее, перевод существующих основанных на методе Монте-Карло пакетов визуализации на использование квазислучайных последовательностей точек связан с рядом проблем:

1. В то время, как датчики псевдослучайных чисел можно найти в стандартных библиотеках многих языков программирования, датчики квазислучайных последовательностей точек не так распространены. Они требуют аккуратной реализации и использования с учетом требований конкретных алгоритмов;
2. Хотя в пакетах на основе метода Монте-Карло допускается формальная замена псевдослучайных чисел на квазислучайные точки, в ряде случаев это не приводит к значительному ускорению. Необходимо предложить и обосновать рекомендации по дополнительным возможным оптимизациям, связанным с переходом на новый метод;
3. В пакетах прикладных программ необходимо оценивать погрешность вычисляемых результатов. Это единственный способ остановить расчет по достижении заданной точности. Аппарат теории вероятностей, используемый для этого в алгоритмах Монте-Карло, неприменим после перехода на квазислучайные последовательности точек ввиду исчезновения псевдослучайности. Необходимо разработать методику, позволяющую оценивать погрешность вычислений модифицированных программ;

4. Создание программ генерации интерактивных последовательностей изображений является нерешенной проблемой при применении методов Монте-Карло. Необходимо исследовать возможности методов квази-Монте-Карло для решения этой проблемы.

Во **второй главе** предложены и обоснованы рекомендации по реализации метода квази-Монте-Карло в современных прикладных пакетах визуализации. Более высокая по сравнению с Монте-Карло скорость сходимости по числу испытаний может быть скрыта возросшей сложностью одного испытания. Поэтому кроме теоретических аспектов реализации метода было уделено внимание и практическим деталям разработки конкретных алгоритмов. В первом разделе этой главы предложены алгоритмы реализации генераторов некоторых квазислучайных последовательностей, оптимизированные для задач трассировки фотонов на современных ЭВМ.

Для дальнейшего изложения важно пояснить требования к генераторам квазислучайных точек в алгоритмах трассировки фотонов. Пусть для трассировки фотонов используется s -мерная квазислучайная последовательность. Обозначим $q_{i0}, q_{i1}, \dots, q_{i(s-1)}$ координаты квазислучайной точки с индексом i . Фотон с индексом i трассируется в соответствии со значениями координат i -й квазислучайной точки. По мере полета фотона координаты этой квазислучайной точки, начиная с q_{i0} , последовательно используются для принятия решений о дальнейшей судьбе фотона (например, направления отражения при столкновении с поверхностью). Если фотон поглощается раньше, чем кончатся координаты, некоторые координаты так и остаются неиспользованными. Очевидно, в таком случае неэффективно генерировать все координаты квазислучайной точки сразу. Поэтому генератор должен допускать вычисление только части координат. Этому требованию не удовлетворяют многие рекуррентные алгоритмы, основанные на предположении о наличии значения $q_{(i-1)j}$ для генерации q_{ij} . Если фотон $(i-1)$ был поглощен раньше, чем была сгенерирована координата $q_{(i-1)j}$, то при генерации q_{ij} для фотона i , такие рекуррентные алгоритмы неприменимы.

Генерация последовательности Холтона

Генерация последовательности Холтона основана на функции радикального обращения (radical inverse) $\Phi_b(i)$ в базе b , определенной на множестве натуральных чисел:

$$\Phi_b(i) = \sum_{r=0}^{\infty} a_r(i)b^{-r-1} \Leftrightarrow i = \sum_{r=0}^{\infty} a_r(i)b^r \quad (5)$$

где i - целое число, а $a_r(i)$ - r -я цифра его представления в b -ичной системе исчисления. По сути, для вычисления $\Phi_b(i)$ необходимо инвертировать порядок цифр в представлении i и поставить плавающую точку в начале полученного числа («традиционный» алгоритм) Например, $\Phi_{10}(1234) = 0.4321$. Квазислучайная точка с индексом i формируется по правилу:

$$i \Rightarrow (\Phi_{b_0}(i), \Phi_{b_1}(i), \dots, \Phi_{b_{s-1}}(i)) \quad (6)$$

где b_0, b_1, \dots, b_{s-1} - первые s простых чисел. В настоящее время для представления индекса фотона i начали использовать тип 64-битного целого, поскольку скорость современных ЭВМ позволяет трассировать столько фотонов, что при использовании типа 32-битного целого все фотоны исчерпываются менее, чем за час вычислений. Вычисление (5) при использовании 64-битных целых занимает существенное время, т.к. необходимые операции деления и деления по модулю реализованы для них крайне неэффективно на современных 32-битных ЭВМ (напр., PentiumIII и Pentium4).

Для исключения дорогостоящих операций с 64-битными целыми, предлагается выполнять радикальное обращение в два этапа. Разложим число i на младшую и старшую части i_{low} и i_{high} , состоящие соответственно из k_{low} и k_{high} разрядов разложения i в базе b . Причем $b^{k_{high}} < 2^{32}$ и $b^{k_{low}} < 2^{32}$. Тогда процедуру радикального обращения можно производить так:

$$\Phi_b(i) = \frac{\Phi_b(i_{high})}{b^{k_{low}}} + \Phi_b(i_{low}) \quad (7)$$

Так как $b^{k_{high}} < 2^{32}$ и $b^{k_{low}} < 2^{32}$, то для представления i_{high} и i_{low} достаточно 32-битных целых. Поэтому при вычисления правой части (7) нет необходимости в дорогостоящих операциях с 64-битными целыми.

Из декомпозиции процедуры радикального обращения можно извлечь дополнительную пользу. При генерации квазислучайных точек в порядке возрастания их индексов (типичный случай), младшие разряды в представлении i меняются часто, в то время как старшие разряды, содержащиеся в i_{high} , остаются большую часть времени неизменными. Поэтому предлагается кэшировать значение первого слагаемого суммы в правой части (7) и пересчитывать его только при изменении i_{high} . Этот алгоритм именуется ниже как «декомпозиции».

Известно, что при наличии значения $\Phi_b(i-1)$, можно ускорить вычисление $\Phi_b(i)$, применив рекуррентную формулу. Предлагается применять эту формулу для вычисления значения $\Phi_b(i_{low})$ в правой части (7). Конечно, при трассировке фотонов предыдущее значение доступно не всегда и требуется дополни-

тельная проверка его наличия. Для максимального переиспользования ранее вычисленных значений, предложена модификация известной рекуррентной формулы. При вычислении $\Phi_b(i)$ она может использовать не только $\Phi_b(i-1)$, но и $\Phi_b(i-l)$ где $l < b$ (алгоритм «рекуррентный+декомпозиции»).

Генерация ЛП $_{\tau}$ -последовательностей

ЛП $_{\tau}$ -последовательности - это целое семейство квазислучайных последовательностей. Их определению предшествует несколько сопутствующих определений. Двоичным параллелепипедом размерности s называется интервал вида:

$$E = \prod_{j=0}^{s-1} \left[\frac{l_j}{2^{d_j}}, \frac{l_j + 1}{2^{d_j}} \right) \quad (8)$$

где целые $d_j \geq 0$ и $0 \leq l_j < 2^{d_j}$. Двоичным участком последовательности q_0, q_1, \dots называется множество членов q_i с номерами i , удовлетворяющими неравенству вида $k2^r \leq i < (k+1)2^r$. Сетка q_0, \dots, q_{2^v-1} из 2^v точек называется ЛП $_{\tau}$ сеткой, если из условия $v > \tau$ следует, что в любой двоичный параллелепипед объемом $1/2^v$ или более попадает количество точек в точности пропорциональное его объему. Бесконечная последовательность точек q_0, q_1, \dots называется ЛП $_{\tau}$ -последовательностью если любой ее двоичный участок является ЛП $_{\tau}$ -сеткой. Определения выше, так же как и способ генерации ЛП $_{\tau}$ -последовательностей любой размерности s , были предложены И.М.Соболев.

Программный модуль, реализованный в работе, позволяет генерировать первые 2^{63} точки некоторой ЛП $_{\tau}$ -последовательности. Для генерации используется таблица направляющих чисел, содержащая $63 \cdot s$ значений c_{jr} . В соответствии с оригинальным алгоритмом, координата j квазислучайной точки с индексом j может генерироваться по следующему правилу («исходный» алгоритм):

$$q_{ij} = a_0(i)c_{j0} \wedge a_1(i)c_{j1} \wedge \dots \wedge a_{62}(i)c_{j62} \quad (9)$$

где $a_0(i), \dots, a_{62}(i)$ - биты в двоичном представлении числа i , а (\wedge) - операция побитового исключающего или. Заметим, что даже на современных 32-битных ЭВМ побитовые операции с 64-битными целыми выполняются достаточно эффективно. Тем не менее, можно дополнительно оптимизировать алгоритм, уменьшив их количество.

Известно несколько алгоритмов для ускорения вычисления (9). Их общим недостатком является то, что они требуют наличия $q_{(i-1)j}$ для вычисления

q_{ij} . Для использования в алгоритмах трассировки фотонов, когда предыдущее значение $q_{(i-1)j}$ часто бывает недоступно, в работе предложен альтернативный алгоритм, именуемый «рекуррентным»:

$$q_{ij} = q_{vj} \wedge ((a_0(v) \wedge a_0(i))c_{j0} \wedge (a_1(v) \wedge a_1(i))c_{j1} \wedge \dots \wedge (a_{62}(v) \wedge a_{62}(i))c_{j62}) \quad (10)$$

где v - любое целое число. Заметим, что так как $a_r(v)$ и $a_r(i)$ являются битами двоичного представления соответственно чисел v и i , все члены вида $(a_r(v) \wedge a_r(i))$ по сути вычисляются одной инструкцией ЭВМ $(v \wedge i)$. v может быть любым числом, для которого ранее было вычислено q_{vj} . Предлагается после вычисления q_{ij} кэшировать полученное значение для его последующего применения в качестве q_{vj} . В работе теоретически показано, что использование (10) вместо (9) приводит к значительному уменьшению числа операций \wedge .

Оценка эффективности разработанных алгоритмов

Ниже описаны результаты численных экспериментов, демонстрирующие предложенные оптимизации для генераторов обоих рассмотренных последовательностей. В таблицах 1 и 2 приведено время работы алгоритмов на модельной задаче генерации последовательности 5-мерных квазислучайных точек в порядке возрастания их индексов с различными приращениями l . Для сравнения дано время известных полностью рекуррентных алгоритмов Келлера и кода Грея, которые могут работать только при $l = 1$. В таблице 3 представлено сравнение предложенных алгоритмов с исходными на реальной задаче трассировки фотонов.

Алгоритм генерации	Время генерации, сек (в зависимости от приращения l)					
	$l = 1$	$l = 2$	$l = 20$	$l = 200$	$l = 2000$	$l = 20000$
Традиционный	15.234	15.953	18.531	20.969	23.485	26.063
Декомпозиции	3.532	3.703	3.547	3.547	3.563	4.070
Рекуррентный+декомпозиции	0.985	1.937	3.609	3.609	3.625	4.141
Келлера	0.812	-	-	-	-	-

Таблица 1: Время генерации 1'000'000 5-мерных квазислучайных точек Холтона.

Из таблиц видно, что генерация ЛП $_{\tau}$ -последовательности является более дешевой процедурой, чем генерация последовательности Холтона. Она также обладает меньшим асимптотическим отклонением и ее генерация не требует операций с плавающей точкой (что означает отсутствие погрешностей округления). Поэтому ЛП $_{\tau}$ -последовательность является наилучшим выбором для использования в алгоритмах генерации статических изображений и именно

Алгоритм генерации	Время генерации, сек (в зависимости от приращения l)					
	$l = 1$	$l = 2$	$l = 20$	$l = 200$	$l = 2000$	$l = 20000$
Исходный	2.422	2.547	2.953	3.359	3.766	4.188
Рекуррентный	0.547	0.671	1.063	1.469	1.781	2.328
Псевдослучайные числа	1.250	1.250	1.250	1.250	1.250	1.250
Код Грея	0.438	-	-	-	-	-

Таблица 2: Время генерации 1'000'000 5-мерных квазислучайных точек ЛП $_{\tau}$ -последовательности и аналогичного количества псевдослучайных чисел.

	Холтона		ЛП $_{\tau}$		Псевдослуч. числа
	Традиционный	Рекур.+Дек.	Исходный	Рекуррентный	
T_1	23.343	2.233	4.936	1.780	2.342
T_2	45.219	24.109	26.812	23.656	24.218
T_1/T_2	51.6%	9.3%	18.4%	7.5%	9.7%

Таблица 3: Сравнение производительности описанных генераторов квазислучайных точек при вычислении на реальной сцене. T_1 - чистое время, затраченное на генерацию квазислучайных точек. T_2 - полное время генерации изображения.

она и была использована в алгоритмах, предлагаемых далее в этой и следующей главах.

Тем не менее, последовательность Холтона обладает важным свойством периодичности. Это свойство делает ее более привлекательной для использования в алгоритмах генерации интерактивных последовательностей изображений, предлагаемых в главе 4.

Трассировка фотонов с использованием квазислучайных последовательностей точек

В качестве основы для реализации идей, предлагаемых в данной работе, был использован алгоритм генерации физически аккуратных изображений, называемый Оценка Плотности по Трассировке Фотонов (Density Estimation Photon Tracing), именуемый в дальнейшем ДЕРТ. Для решения уравнения рендеринга алгоритм ДЕРТ использует прямую трассировку Монте-Карло: фотоны испускаются из источников света и трассируются, отражаясь и преломляясь поверхностями сцены. Фотон трассируется до тех пор, пока он либо не будет поглощен одной из поверхностей, либо не вылетит за пределы сцены. Освещенность точки поверхности определяется плотностью в ней фотонов.

В оригинальном алгоритме ДЕРТ для реализации траектории фотона используются псевдослучайные числа. Пусть $\eta_0, \eta_1, \dots, \eta_{(s-1)}$ - случайные величины. При трассировке i -го фотона используются значения $\eta_{i0}, \eta_{i1}, \dots, \eta_{i(s-1)}$ этих случайных величин в i -м испытании. Вместо псевдослучайных чисел $\eta_{i0}, \eta_{i1}, \dots, \eta_{i(s-1)}$ можно использовать координаты $q_{i0}, q_{i1}, \dots, q_{i(s-1)}$ i -й точки

какой-либо квазислучайной последовательности. При этом индекс i полностью определяет значение этих координат, а значит и траекторию фотона.

Понижение конструктивной размерности алгоритма.

Задача интегрирования методом Монте-Карло обычно сводится к вычислению математического ожидания некоторой функции $\Phi(\eta_0, \dots, \eta_{s-1})$, где $\eta_0, \dots, \eta_{s-1}$ - независимые случайные величины. Количество s случайных величин, необходимое конкретному алгоритму для вычисления интеграла называется конструктивной размерностью этого алгоритма.

Вообще говоря, алгоритм DEPT имеет бесконечную конструктивную размерность. Траектория фотона может (теоретически) состоять из сколь угодно большого числа звеньев, так что нельзя указать заранее, сколько значений понадобится для реализации такой траектории. На практике же фотоны быстро поглощаются и поэтому имеет смысл рассмотреть вопрос о среднем количестве случайных величин, необходимых для трассировки одного фотона. В дальнейшем под конструктивной размерностью алгоритма будет подразумеваться именно это среднее значение.

Оригинальному алгоритму DEPT необходимо следующее количество псевдослучайных значений для трассировки одного фотона:

$$D_r = D_l + B_s + 3B_d + C \quad (11)$$

где B_s и B_d - количество соответственно идеально и не идеально зеркальных отражений/преломлений фотона поверхностями сцены, C равно 1 в случае, если фотон вылетел за пределы сцены и 2 в случае поглощения.

Известно, что при выборе дискретных событий можно переиспользовать одно и то же псевдослучайное значение несколько раз, применяя каждый раз процедуру перенормировки. Этот метод носит название модифицированного метода суперпозиции и был первоначально предложен для экономии псевдослучайных чисел в целях уменьшения времени, необходимого для их генерации. В работе предлагается применять его для квазислучайных значений в целях понижения конструктивной размерности алгоритма.

Пусть необходимо, используя число η_1 , выбрать одно из n дискретных событий, имеющих вероятности p_1, \dots, p_n . Будет выбрано событие k такое, что $\sum_{j=1}^{k-1} p_j \leq \eta_1 < \sum_{i=1}^k p_j$. Процедура перенормировки числа η_1 задается формулой:

$$\eta'_1 = \frac{\eta_1 - \sum_{j=1}^{k-1} p_j}{p_k} \quad (12)$$

Конструктивная размерность трассировки одного фотона в таком модифицированном алгоритме значительно меньше первоначальной (11):

$$D'_r \leq D_l + 2B_d + C \quad (13)$$

Уменьшение дисперсии: метод существенной выборки.

Для уменьшения дисперсии (а значит увеличения точности) в алгоритме ДЕРТ, так же как и во многих подобных алгоритмах, применяется метод существенной выборки. Его особенность состоит в том, что плотность фотонов, излученных/рассеянных из точки в любом направлении, в точности пропорциональна потоку энергии в этом направлении. Основные два места применения метода существенной выборки - излучение фотонов из источника света и рассеяние фотонов при столкновении их с поверхностью. В первом случае плотность излучения фотонов определяется функцией силы света источника, во втором - ДФРС поверхности в точке попадания фотона. Наиболее общим представлением для этих функций является задание их на сетке в сферической системе координат. Рассмотрим случай источника света (случай ДФРС поверхности во многом похож и здесь не рассматривается, хотя в работе рассмотрены его особенности).

Пусть $\phi_0, \phi_1, \dots, \phi_{p-1}$ и $\theta_0, \theta_1, \dots, \theta_{t-1}$ - упорядоченные наборы соответственно азимутальных и полярных углов. Тогда узлами сетки задания силы света $I(\phi, \theta)$ источника являются всевозможные направления (ϕ_i, θ_j) где $0 \leq i < p$ и $0 \leq j < t$. Сила света по направлению (ϕ, θ) определяется билинейной интерполяцией четырех угловых значений содержащей его ячейки сетки (т.е. ячейки, ограниченной углами ϕ_i, ϕ_{i+1} и θ_j, θ_{j+1} где $\phi_i \leq \phi < \phi_{i+1}$ и $\theta_j \leq \theta < \theta_{j+1}$). Задача состоит в том, чтобы по равномерно распределенным в единичном квадрате значениям (η_1, η_2) выбирать направления излучения фотонов (ϕ, θ) так, чтобы плотность фотонов в каждом направлении была пропорциональна $I(\phi, \theta)$.

Первым шагом алгоритма является нахождение одной из $(p - 1)(t - 1)$ ячеек сетки, через которую будет излучен фотон. Для этого на основании значения η_1 выбирается одно из $(p - 1)(t - 1)$ дискретных событий, вероятность каждого из которых пропорциональна потоку через соответствующую ячейку. Выбор события и последующая перенормировка η_1 описаны в предыдущем разделе. Для вычисления направления внутри выбранной ячейки в алгоритмах Монте-Карло обычно применяется метод отказов. От состоит в том, что в соответствии с равномерным распределением генерируются пары углов (ϕ, θ) до тех пор, пока очередная пара не удовлетворит определенному условию. Известно, что использование метода отказов приводит к зна-

чительному понижению эффективности алгоритмов квази-Монте-Карло по двум причинам: возрастает конструктивная размерность алгоритма и, образно выражаясь, увеличивается разрывность интегрируемой функции. Поэтому при переходе на квазислучайные последовательности точек необходимо было найти альтернативный способ.

В работе получены формулы для нахождения углов (ϕ, θ) по равномерно распределенным значениям (η_1, η_2) без использования метода отказов. Искомые углы могут быть найдены из уравнений:

$$\eta'_1 = \frac{\phi_2 - \phi_1}{2(\theta_2 - \theta_1) \int_{\Omega} I(\phi, \theta)} \left[(B_1 + B_2 - A_1 - A_2)(\sin(\theta) - \theta \cos(\theta) - \sin(\theta_1) + \theta_1 \cos(\theta_1)) - (\cos(\theta_1) - \cos(\theta))(\theta_1(B_1 + B_2) - \theta_2(A_1 + A_2)) \right] \quad (14)$$

$$\eta_2 = \frac{(\phi - \phi_1) \left[(B_2(\phi - \phi_1) + B_1(2\phi_2 - \phi_1 - \phi))(\theta - \theta_1) + (A_2(\phi - \phi_1) + A_1(2\phi_2 - \phi_1 - \phi))(\theta_2 - \theta) \right]}{2(\phi_2 - \phi_1)(\theta_2 - \theta_1) p_1(\theta) \int_{\Omega} I(\phi, \theta)}$$

где Ω - все сфера возможных направлений, а A_1, A_2, B_1, B_2 - значения силы света в углах ячейки, в которую происходит излучение фотона. Первое из уравнений (14) является трансцендентным относительно θ и не решается аналитически. Было реализовано решение этого уравнения методами секущих, Ньютона и Риддера. Тестирование на различных типах источников света показало, что метод секущих обеспечивает наилучшую производительность при требуемой точности вычислений. Уравнение для ϕ квадратное и его решение не вызывает трудностей.

Оценка эффективности предложенных методов

Для тестирования предложенных алгоритмов использовалась сцена с минимальным количеством деталей освещения, что упрощает визуальную оценку уровня шума. Сцена состоит из квадратной пластины, освещенной источником с распределением силы света, заданным таблично. Между пластиной и источником света находится стекло с коэффициентом преломления 1.33.

На Рис. 1 представлено три серии изображений и указана относительная ошибка и время их генерации. Серия а) демонстрирует изображения сцены, полученные алгоритмом DEPT после формальной замены псевдослучайных чисел квазислучайной последовательностью точек. Серия б) получена после замены метода отказов на метод обратных функций описанным выше способом. Статистика по времени показывает, что численное решение трансцендентных уравнений замедляет трассировку каждого фотона на $\sim 4.5\%$, но позволяет достичь значительно лучшей точности (до 3.5 раз) при том же количестве фотонов.

сокой точностью. Для ее вычисления в алгоритме DEPT используется следующий способ. Множество всех фотонов разбивается на два подмножества A и B , которые состоят из фотонов с четными и нечетными индексами соответственно. С их помощью вычисляется два различных решения: $E_a(x)$ и $E_b(x)$. Известно, что оценка:

$$\tilde{E}r_{abs} = \sqrt{\frac{\int_S (E_a(x) - E_b(x))^2}{S}} \quad (15)$$

является несмещенной для абсолютной погрешности вычислений.

Покажем, что описанный метод оценки погрешности неприменим непосредственно, если для трассировки фотонов используются квазислучайные последовательности точек. Например, рассмотрим последовательность q_1, q_2, q_3, \dots квазислучайных точек Холтона. Координата q_{i0} (первая координата точки с индексом i) равна $\Phi_2(i)$, где $\Phi_2(\cdot)$ - функция радикального обращения по базе 2. Легко проверить, что $\Phi_2(i) < 0.5$ если i четное и $\Phi_2(i) \geq 0.5$ иначе. Таким образом, все четные(нечетные) точки последовательности Холтона находятся в левой(правой) половине единичного куба и составленная из них последовательность не является равномерно распределенной. Поэтому решения $E_a(x)$ и $E_b(x)$, полученные с помощью соответственно четных и нечетных точек последовательности Холтона не обязаны стремиться к искомому решению $E(x)$, а их разность не обязательно стремится к 0. Легко проверить, что то же относится и к ЛП $_{\tau}$ - последовательностям.

В работе предложен способ получения двух равномерно распределенных последовательностей A и B , применимый к обоим рассматриваемым последовательностям. Для этого используется следующий алгоритм:

Алгоритм 1 Разбиение квазислучайных точек на подмножества A и B .

```

1: if  $q_{i0} < 0.5$  then
2:   Назначить эту квазислучайную точку подмножеству  $A$ ;
3:    $q_{i0} = 2q_{i0} + (1/3)$ ;
4:   if  $q_{i0} \geq 1$  then
5:      $q_{i0} = q_{i0} - 1$ 
6:   end if
7: else
8:   Назначить эту квазислучайную точку подмножеству  $B$ ;
9:    $q_{i0} = 2(q_{i0}-0.5)$ ;
10: end if

```

Заметим, что выбор принадлежности точки множеству с последующей перенормировкой похож на процедуру выбора дискретного события, описанную

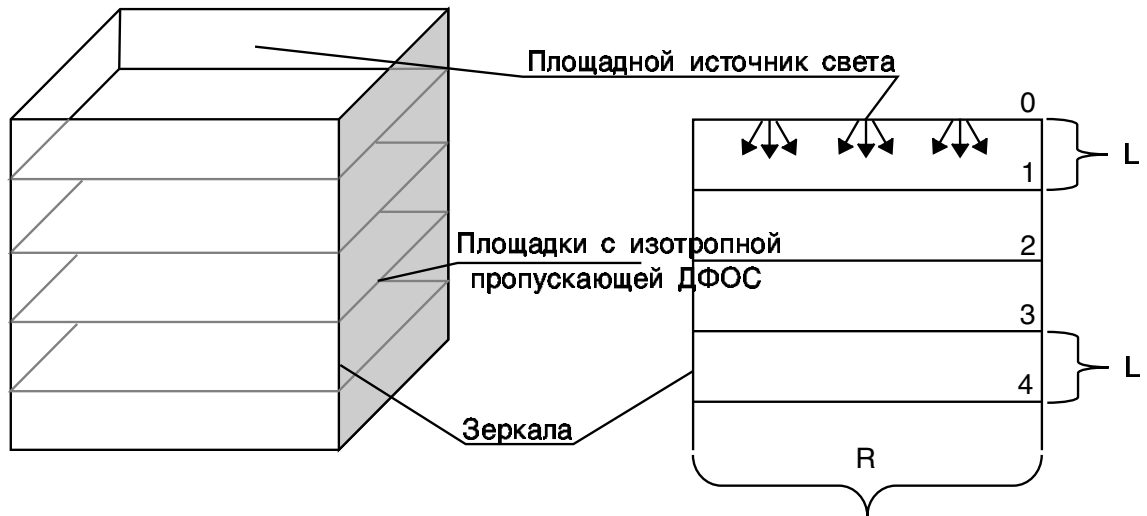


Рис. 2: Сцена, допускающее аналитическое решение и позволяющая варьировать факторы, влияющие на эффективность алгоритма квази-Монте-Карло.

выше. В работе показано, что полученные последовательности A и B равномерно распределены, а значит решения $E_a(x)$ и $E_b(x)$ стремятся к точному решению $E(x)$ при стремлении числа фотонов к бесконечности. Поэтому оценка погрешности, задаваемая формулой (15) всегда стремится к 0. Однако, это не гарантирует ее совпадение с точной погрешностью. Для проверки этого совпадения были произведены численные эксперименты.

В литературе рассматривают два основных фактора, влияющих на сходимость алгоритмов квази-Монте-Карло. Это разрывность задачи и ее размерность, которая в значительной степени определяется конструктивной размерностью. Поэтому необходимо проверить соответствие предложенной оценки погрешности ее реальному значению при изменении разрывности и размерности задачи. Для этого в работе специально спроектирована сцена, изображенная на Рис 2.

Сцена представляет из себя открытый снизу параллелепипед, в который помещены горизонтальные пластины через равные расстояния L . В основании параллелепипеда лежит квадрат со стороной R , равной 16 метрам. Боковые стенки параллелепипеда - идеальные зеркала. Крышка (или пластина 0) - площадной источник света со светимостью E , функция яркости которого не зависит от точки и всюду симметрична относительно вертикали. В качестве ДФРС пластин была выбрана взвешенная сумма идеально зеркального и идеально диффузного пропускания с весами соответственно k_{ts} и k_{td} . В работе получено аналитическое решение освещенности на каждой пластине:

$$E_k = E(k_{ts} + k_{td})^{(k-1)} \quad (16)$$

где k - индекс пластины. Интересно отметить, что освещенность пластин не зависит ни от расстояния L между ними, ни от соотношения между k_{ts} и k_{td} при условии, что их сумма остается постоянной. Тем не менее, изменение этих параметров влияет на эффективность алгоритма квази-Монте-Карло, что демонстрируется результатами численных экспериментов.

Оценка погрешности производилась отдельно на каждой горизонтальной пластине. Как и ожидалось, погрешность алгоритма Монте-Карло не зависит от индекса пластины. В тоже время погрешность квази-Монте-Карло увеличивается с отдалением от источника света, оставаясь тем не менее всегда меньше погрешности Монте-Карло. Замедление сходимости квази-Монте-Карло с отдалением от источника света связано с возрастанием конструктивной размерности задачи (фотоны претерпевают больше рассеяний для достижения более далеких пластин). В работе построена зависимость оценки погрешности от различных условий и проведено сравнение оценки и реального значения погрешности для следующих случаев:

1. $k_{td} = 0.2$, $k_{ts} = 0.8$, $L = 0.1$, количество фотонов изменяется от $\sim 250'000$ до $\sim 18'000'000$;
2. $L = 0.1$, количество фотонов $\sim 18'000'000$, k_{td} изменяется от 0 до 0.8, $k_{ts} = 1 - k_{td}$;
3. количество фотонов $\sim 18'000'000$, $k_{td} = 0.2$, $k_{ts} = 0.8$, L изменяется от 0.1 до 0.8.

Во всех случаях наблюдается практическое совпадение кривых оцениваемой и реальной погрешностей. Некоторые из соответствующих графиков представлены на Рис. 3. Из графика а) видно, что предложенная оценка погрешности отслеживает даже локальные изменения градиента.

Предложенная методика была также проверена на реальной сцене ДОМ, содержащей приблизительно 377'900 треугольников и 43 источника света с таблично заданными функциями яркости. Погрешность получаемых решений вычислялась исходя из эталонного решения, которое было получено из расчета Монте-Карло с погрешностью 0.05%. Наблюдается полное соответствие между оценкой погрешности и ее реальным значением.

В **четвертой главе** рассматривается возможность модификации разработанного алгоритма для генерации интерактивных последовательностей изображений. Большинство из существующих пакетов программ генерации физически аккуратных изображений не предназначены для работы в реальном

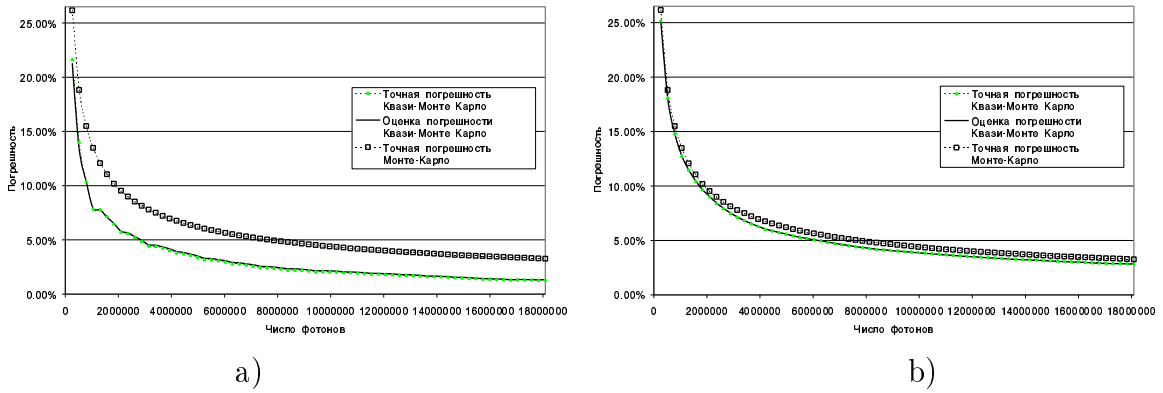


Рис. 3: Погрешность в зависимости от количества фотонов, $k_{td}=0.2$, $k_{ts}=0.8$, $L=0.1$. а) пластина 1. б) пластина 5.

времени и с динамическими сценами. Время отклика, необходимое таким программам для обновления изображений, непомерно велико даже при минимальных изменениях в сцене, потому что вычисления приходится повторять сначала. В то же время, известно, что непрямоое освещение меняется очень плавно при изменении параметров сцены, и для получения нового изображения обычно требуется лишь небольшая коррекция ранее полученного.

В этой главе предлагается новая методика коррекции ранее полученных изображений, специально предназначенная для интерактивных приложений. Предлагаемый алгоритм использует свойство периодичности последовательности Холтона, которое позволяет эффективно находить фотоны с похожими траекториями. Это делает возможным выборочную трассировку фотонов в те регионы сцены, где освещение должно быть обновлено из-за изменений в геометрии, ДФРС поверхностей или других параметрах сцены. Уникальная особенность предлагаемого алгоритма «выборочной трассировки фотонов» состоит в том, что хотя когерентность по времени непрямого освещения в значительной степени использована, алгоритму не требуется практически никаких дополнительных структур данных для хранения траекторий фотонов в изменяющейся сцене. Это позволяет применять предложенную методику даже для очень сложных сцен на обычных персональных компьютерах, оснащенных графическими ускорителями.

Периодичность последовательности Холтона.

Наблюдение о том, что последовательность Холтона имеет ярко выраженную периодическую природу, вряд ли можно назвать новым. Тем не менее, это свойство, по-видимому, до сих пор не имело применения в компьютерной графике. В работе доказано следующее свойство функции радикального обращения:

$$|\Phi_b(i) - \Phi_b(i + mN_g)| < \frac{1}{b^k}, \text{ если } N_g = lb^k \quad (17)$$

где k , l и m - целые числа такие, что $k \geq 0$ и $l > 0$. Выбирая N_g как:

$$N_g = b_0^{k_0} b_1^{k_1} \dots b_n^{k_n} \quad (18)$$

можно добиться близкого совпадения координат точек последовательности Холтона, соответствующих базам b_0, \dots, b_n . Так как точкам с близкими координатами обычно соответствуют фотоны с близкими траекториями, это позволяет разбивать все множество фотонов на когерентные пучки.

Обновление непрямого освещения

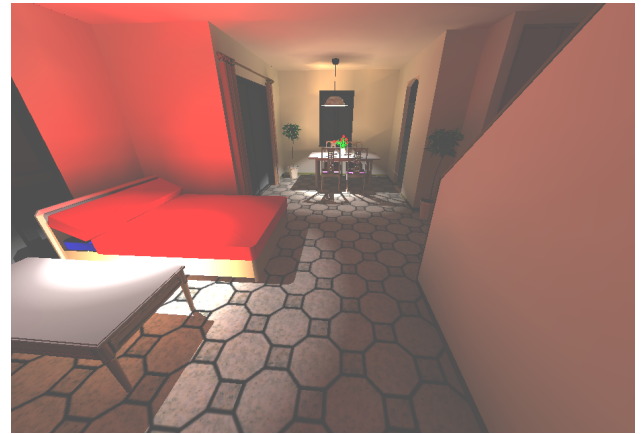
Для обновления освещения после изменения сцены необходимо перетрассировать фотоны, пересекшие модифицированные объекты, чтобы учесть изменения траекторий этих фотонов. Назовем такие фотоны «динамическими». Перетрассировка каждого динамического фотона состоит в трассировке его для старой конфигурации сцены с изменением знака его энергии на отрицательный, и затем трассировке его для новой конфигурации сцены с положительной энергией. Так как обычно только небольшое число фотонов являются динамическими, такая двойная работа оправдана по сравнению с вычислением решения заново для всей сцены.

Эффективный поиск фотонных траекторий, пересекающих модифицированные объекты, является вычислительно сложной задачей. Так как фотонная траектория не хранится в памяти ЭВМ, пересечение можно проверять только в процессе вычисления этой траектории по индексу соответствующей квазислучайной точки. Для увеличения эффективности, фотонные траектории трассируются не по одной, а когерентными пучками. Для этого все имеющиеся фотонные траектории разделяются на группы, каждая из которых содержит две подгруппы, именуемые соответственно *корректирующие* и *пробные* фотоны. Группировка производится таким образом, что индексы пробных фотонов увеличиваются с интервалом 1, индексы корректирующих фотонов имеют интервал N_g между ними, а полное количество групп также равно N_g (где N_g определяется в соответствии с формулой (18)).

В соответствии с описанным выше свойством периодичности последовательности Холтона, корректирующие фотоны одной группы составляют когерентный пучок. Пробные фотоны, напротив, не когерентны и соответствующие им квазислучайные точки равномерно распределены в единичном гиперкубе. Алгоритм трассирует пробные фотоны с тем, чтобы обнаружить траектории, пересекающие модифицированные объекты сцены. В работе по-



а)



б)

Рис. 4: Интерактивная сессия для сцен а) КОМНАТА и б) ДОМ.

казано, что зная такую траекторию можно найти пучок корректирующих фотонов, также пересекающий эти объекты. Приоритет группы, содержащий этот пучок корректирующих фотонов, увеличивается.

Алгоритм работает итеративно. Каждая итерация состоит в выборе фотонной группы с наивысшим приоритетом и ее перетрассировке. После завершения одной или нескольких итераций, частично обновленное решение выводится на экран. Пользователь не обязан дожидаться полного обновления освещения и может снова изменить сцену в любой момент времени. Алгоритм является консервативным в том смысле, что максимально через N_g итераций он сходится к полностью корректному решению. Но так как фотонные группы перетрассируются в порядке убывания их важности для обновления освещения, визуальное корректное изображение вычисляется гораздо раньше - в первые несколько секунд после модификации сцены.

Алгоритм квази-Монте-Карло используется только для вычисления непрямой освещенности. Прямая освещенность, включая тени, вычисляется с помощью графического ускорителя GeForce3. Для оценки эффективности использовались сцены: КОМНАТА ($\sim 12'400$ треугольников и четыре источника света) и ДОМ ($\sim 377'900$ треугольников и два источника света). Эти сцены изображены на Рис. 4. Использовался компьютер 1.7 Ghz Dual PentiumIII Хеон. Полученная частота смены кадров равна 8 и 1.1 кадра в секунду для сцен КОМНАТА и ДОМ соответственно. Полное (визуально) обновление освещения достигается для этих сцен за 2-4 сек и 4-8 сек соответственно.

В заключении сформулированы основные результаты работы.

Приложения содержат список использованных обозначений и терминологический указатель.

Основные результаты работы:

1. Разработаны рекомендации по реализации программных модулей генерации точек квазислучайных последовательностей Холтона и $ЛП_{\tau}$, оптимизированные специально для задач трассировки фотонов и на современных ЭВМ. Реализованные модули требуют при трассировке фотонов даже меньше вычислительных ресурсов, чем стандартный генератор псевдослучайных чисел.
2. Реализован комплексный алгоритм, использующий квазислучайные последовательности точек для вычисления непрямого освещения в программных средствах визуализации. В рамках его реализации:
 - решена задача моделирования источников света и ДФРС поверхностей общего вида без использования метода отказов;
 - предложена методика понижения конструктивной размерности алгоритма.

Проведены численные эксперименты и представлены их результаты, показывающие значительно возросшую скорость вычислений по сравнению с исходным алгоритмом.

3. Предложена методика оценки погрешности разработанного комплексного алгоритма квази-Монте-Карло. Предложенная методика позволяет уведомлять пользователя о погрешности вычисленного изображения в произвольные моменты времени и не требует значительных вычислительных затрат для ее реализации.
4. Предложена методика, использующая свойства квазислучайной последовательности Холтона для обновления данных о непрямом освещении после изменения сцены. На основе этой методики реализован программный модуль, позволяющий пользователю интерактивно вносить изменения в сцену и видеть изменения в прямом и непрямом освещении. Даны рекомендации по реализации соответствующих прикладных программ, включая аспекты использования современных графических ускорителей и распараллеливания вычислений.
5. Продемонстрированы преимущества использования метода квази-Монте-Карло для задач генерации физически аккуратных изображений и их последовательностей. Разработанные алгоритмы существенно облегчают применение этого метода в прикладных программах визуализации, так как снимают многие существовавшие ранее проблемы.

Список литературы

1. Дмитриев К.А. От Монте-Карло к квази-Монте-Карло // Труды конференции ГрафиКон'2002. — 2002. — С. 53-59.
2. Дмитриев К.А. Анализ методов оптимизации вычисления глобальной освещенности в компьютерной графике. // Технический отчет, Проект РФТР № 85/2001. 2001 — 53 С.
3. Dmitriev K. Efficiency issues on ray tracing machine // Труды конференции ГрафиКон'2000. — 2000. — С. 99-103.
4. Dmitriev K., Brabec S., Myszkowski K., and Seidel H.-P. Interactive global illumination using selective photon tracing // Rendering Techniques'2002. — 2002. — P. 100-113.
5. Kopylov E. and Dmitriev K. Light propagation visualization as a tool for 3d scene analysis in lighting design // Computers & Graphics. — 2000. — Vol. 24, No. 1. — P. 31-39.
6. Kulikova A. and Dmitriev K. Fuzzy reflections rendering // Труды конференции ГрафиКон'2001. — 2001. — С. 88-91.