DOI: 10.15514/ISPRAS–2020–33(2)–1

Современное состояние методов расчёта глобальной освещённости в задачах реалистичной компьютерной графики

[1,2] *В.А. Фролов, ORCID: 0000-0001-8829-9884 <vfrolov@graphics.cs.msu.ru>*
[1] *А.Г. Волобой, ORCID: 0000-0003-1252-8294 <voloboy@gin.keldysh.ru>*
[1] *С.В. Ершов, ORCID: 0000-0002-5493-1076 <sergey_65@mail.ru>*
[1] *В.А. Галактионов, 0000-0001-6460-7539 <vlgal@gin.keldysh.ru>*
[1] *Институт прикладной математики им. М.В. Келдыша РАН, 125047, Россия, Москва, Миусская пл., д. 4*
[2] *Московский государственный университет имени М.В. Ломоносова, 119991, Россия, Москва, Ленинские горы, д. 1*

Аннотация. Современная реалистичная компьютерная графика базируется на физически корректном моделировании распространения света. Одной из основных и трудно вычислимых задач при этом является расчет глобальной освещенности, т.е. распределения света в виртуальной сцене, учитывающий множественные отражения и рассеяния света и всевозможные виды взаимодействия его с объектами сцены. Этой проблеме посвящены сотни публикаций, описывающие десятки методов вычисления глобальной освещенности и их модификации. В данной обзорной статье мы бы хотели не просто перечислить и кратко описать эти методы, но и дать некоторую «карту» существующих работ, которая позволит читателю сориентироваться, понять их достоинства и недостатки и, тем самым, выбрать для себя подходящий базовый метод. Особое внимание уделяется таким характеристикам методов как надёжность и универсальность в отношении используемых моделей, прозрачность их верификации, возможность эффективной реализации на GPU, а также накладываемые на сцену или феномены освещённости ограничения. В отличие от существующих обзорных работ анализируется не только эффективность методов, но также их ограничения и сложность программной реализации. Кроме того, мы предоставляем результаты собственных численных экспериментов с различными методами, служащих иллюстрациями к выводам.

Ключевые слова: расчёт освещённости, глобальная освещённость, трудновычислимые феномены освещённости

Для цитирования: Фролов В.А., Волобой А.Г., Ершов С.В., Галактионов В.А. Современное состояние методов расчёта глобальной освещённости в задачах реалистичной компьютерной графики. Труды ИСП РАН, том 33, вып. 2, 2021 г., стр. 7-48. DOI: 10.15514/ISPRAS–2021–33(2)–1

Благодарности. Авторы выражают благодарность Кириллу Гаранже (K. Garanzha) за ряд ценных критических замечаний, высказанных в процессе подготовки статьи.

The current state of the methods for calculating global illumination in tasks of realistic computer graphics

[1,2] *V.A. Frolov, ORCID: 0000-0001-8829-9884 <vfrolov@graphics.cs.msu.ru>*
[1] *A.G. Voloboy, ORCID: 0000-0003-1252-8294 <voloboy@gin.keldysh.ru>*
[1] *S.V. Ershov, ORCID: 0000-0002-5493-1076 <sergey_65@mail.ru>*
[1] *V.A. Galaktionov, 0000-0001-6460-7539 <vlgal@gin.keldysh.ru>*
[1] *Keldysh Institute of Applied Mathematics Russian Academy of Science, 4, Miusskaya sq., Moscow, 125047, Russia*
[2] *Lomonosov Moscow State University, GSP-1, Leninskie Gory, Moscow, 119991, Russia*

Abstract. Modern realistic computer graphics are based on light transport simulation. In this case, one of the main and difficult to calculate tasks is to calculate the global illumination, i.e. distribution of light in a virtual scene, taking into account multiple reflections and scattering of light and all kinds of its interaction with objects in the scene. Hundreds of publications and describing dozens of methods are devoted to this problem. In this state-of-the-art review, we would like not only to list and briefly describe these methods, but also to give some "map" of existing works, which will allow the reader to navigate, understand their advantages and disadvantages, and, thereby, choose a right method for themselves. Particular attention is paid to such characteristics of the methods as robustness and universality in relation to the used mathematical models, the transparency of the method verification, the possibility of efficient implementation on the GPU, as well as restrictions imposed on the scene or illumination phenomena. In contrast to the existing survey papers, not only the efficiency of the methods is analyzed, but also their limitations and the complexity of software implementation. In addition, we provide the results of our own numerical experiments with various methods that serve as illustrations for the conclusions.

Keywords: light transport simulation; global illumination; hard sampling lighting phenomena

For citation: Frolov V.A., Voloboy A.G., Ershov S.V., Galaktionov V.A. The current state of the methods for calculating global illumination in tasks of realistic computer graphics. Trudy ISP RAN/Proc. ISP RAS, vol. 33, issue 2, 2021, pp. 7-48 (in Russian). DOI: 10.15514/ISPRAS–2021–33(2)–1.

Acknowledgments. The authors are grateful to Kirill Garanzha for a number of valuable critical comments during the preparation of this article.

1. Введение

Расчёт освещённости (другие названия области – глобальное освещение, global illumination, lighting simulation, light transport) в реалистичной компьютерной графике – это бездонный колодец, в котором необходимость в повышении производительности не иссякает. Ускорение расчетов – доминирующий мотив большинства научных работ в данной области. Причина этого заключается в том, что моделирование явлений реального мира – это задача, не имеющая предела. Большая производительность на практике, как ни странно, не всегда приводит к уменьшению времени расчёта кадра. Она приводит к тому, что пользователь за отведённое время может посчитать более сложную 3D сцену или промоделировать более сложные явления.

За последнее десятилетие компьютерная графика добилась впечатляющих успехов в области моделирования и расчёта глобальной освещённости. Сегодня системы моделирования умеют рассчитывать такие явления и конфигурации оптических систем, которые ранее было практически невозможно моделировать из-за того, что методы просто не сходились к точному решению за разумное время. Кроме того, с развитием вычислительных систем стало возможным моделировать освещение для сцен с массивной геометрией, а также рассчитывать точное освещение в анимации для киноиндустрии.

Однако на данный момент в области глобальной освещённости так много разрозненных научных работ, что прикладному исследователю или разработчику крайне сложно выбрать

правильный метод для конкретной задачи. Эта проблема усугубляется несколькими трудностями.

Во-первых, сравнение производительности существующих методов – довольно сложная задача. В компьютерном зрении, например, существует большое количество фиксированных наборов данных, на которых проверяется эффективность и точность алгоритмов классификации. Они являются de facto стандартами, по отношению к которым оценивается тот или иной алгоритм. В графике ситуация обстоит иначе. Не существует открытых наборов сцен, в которых разные исследователи могли бы получить совпадающие изображения (в основном из-за отсутствия необходимых для этого стандартов). Это приводит к практике так называемого «cherry picking» – аккуратного подбора сцен и условий освещения таким образом, чтобы продемонстрировать преимущества алгоритма, разработанного авторами. Cherry picking в общем случае не является недостатком работы, т.к. новые алгоритмы, как правило, разрабатываются с тем, чтобы решить определённый класс проблем предыдущих методов. Если эти проблемы решены, то другие проблемы могут оставаться за рамками исследования. Однако разработчикам практических приложений, перед которыми стоит непростой вопрос выбора метода, от этого легче не становится.

Вторая проблема заключается в том, что некоторые современные и эффективные двунаправленные методы не просто более сложны, но и значительно более ограничены условиями, в которых эти методы работают правильно. Кроме того, нет общей методологии верификации, которая гарантировала бы корректность метода на любой сцене. А это означает, что во многих случаях эти методы не могут быть использованы для инженерных целей, как, например, проектирование оптических устройств, где корректность важна в первую очередь.

В результате, в каждом конкретном случае выбор базового расчётного метода и его развитие становится нетривиальной задачей. Мы полагаем, что наша работа поможет исследователям и разработчикам в области компьютерной графики и оптического моделирования сделать обоснованный выбор базового метода и грамотно определить собственное направление развития.

2. Используемые сокращения и термины

Далее мы в алфавитном порядке расшифруем основные сокращения из нашей работы и рис. 1. Большинство из них являются общепринятыми.

- BDPM – Bidirectional Photon Mapping [37], метод двунаправленных фотонных карт;
- BSDF – Bidirectional Scattering Distribution Function или двунаправленная функция отражения-рассеяния; именно эта функция описывает взаимодействие света с поверхностью, т.е. определяет модель материала;
- BPT – Bidirectional Path Tracing [3], двунаправленная трассировка путей.
- CC-BPT – Caustic Connection Strategies for Bidirectional Path Tracing [14];
- CMIS – Continuous Multiple Importance Sampling [57], метод многократной выборки по значимости, расширенный на случай континуума стратегий;
- ERPT – Energy Redistribution Path Tracing [105];
- FG – Final Gathering, финальный сбор, метод, откладывающий сбор из фотонной карты на одно переотражение;
- HHMC – Hessian Hamiltonian Monte Carlo light transport [94];
- HMC – Hamiltonian Monte Carlo [91];
- HSLT – Half Space Light Transport [77];
- IBPT – Instant BPT [6], урезанная версия BPT, которая может рассматриваться как оптимизация;

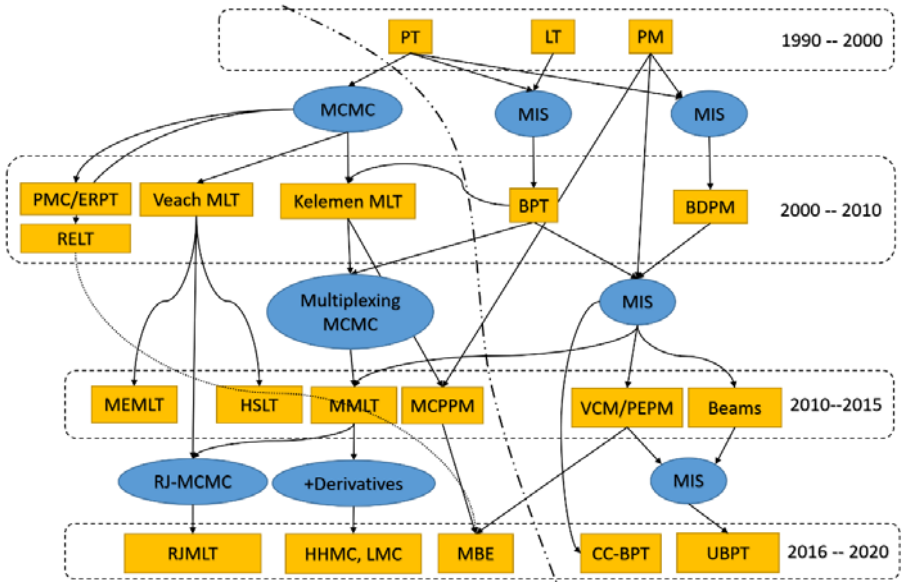

Рис. 1. Приблизительная карта современных методов интегрирования освещённости. В правой части изображены методы, основанные на обыкновенном Монте-Карло, в левой части – на основе Монте-Карло по схеме марковских цепей. Левая и правая части условно разделены пунктирной линией с двумя точками. Прямоугольники представляют собой названия конкретного метода или класса методов расчёта освещения. Овалы представляют собой базовый математический инструментарий, на основе которого строятся методы. Стрелки показывают, что одни методы построены на основе других методов или определённых математических инструментариев

Fig. 1. An approximate map of modern methods of light transport. Methods based on ordinary Monte Carlo are shown on the right side, on the left side – based on Markov Chain Monte Carlo. The left and right parts are conditionally separated by a dotted line with two dots. The rectangles represent the names of a particular method or class of light transport methods. Ovals represent the basic mathematical toolkit on the basis of which methods are built. Arrows indicate that some methods are built on the basis of other methods or certain mathematical tools

- Kelemen MLT (или Primary Sample Space MLT (PSSMLT)) – явное указание на то, что MLT реализован в первичном пространстве путей, как в работе Келемена (Csaba Kelemen) и др. [75];
- LMC – Langevin Monte Carlo light transport [97];
- leap frog – традиционный способ реализации HMC, требующий большого количества промежуточных шагов, на каждом из которых необходимо вычислять целевую функцию [91].
- LT – Light Tracing (Forward Monte Carlo), прямая Монте-Карло трассировка;
- MALA – Metropolis-adjusted Langevin algorithm [103].
- MBE – Metropolized Bidirectional Estimator [41];
- MCMC – Markov Chain Monte Carlo, Монте-Карло по схеме марковских цепей;
- MEMLT – Manifold Exploration Metropolis Light Transport [15];
- MCPPM – Markov Chain PPM (Progressive Photon Mapping) [39];
- MIS – Multiple Importance Sampling [3], многократная выборка по значимости;

- MIS PT – MIS Path Tracing, обратная трассировка путей с использованием многократной выборки по значимости;
- MLT – Metropolis Light Transport [73];
- MMLT – Multiplexed Metropolis Light Transport, Multiplexed MLT [78];
- OMC – Ordinary Monte Carlo, обыкновенный метод Монте-Карло интегрирования;
- PDF – Probability Density Function, плотность вероятности;
- PCBPT – Probability Connection Bidirectional Path Tracing [7], оптимизация BPT;
- PCLT – Pixel Cache Light Tracing [30];
- PEPM – Path space Extension for Photon Mapping [25];
- PM – Photon Mapping, фотонные карты [17];
- PMC – Population Monte Carlo [106], Монте-Карло на основе отбора популяции выборок;
- PT – Path Tracing [2] (Backward Monte Carlo), обратная трассировка путей;
- PPM – Progressive Photon Mapping [18], прогрессивные фотонные карты;
- RELT – Replica Exchange Light Transport [80];
- RJMLT – Reversible Jump Metropolis Light Transport [88-90];
- RIS – Re-sampling for Importance Sampling [58];
- SDS-пути – Specular Diffuse Specular, вид путей, в которых между двумя зеркальным отражениями встречается одно диффузное;
- SPPM – Stochastic Progressive Photon Mapping [19], стохастические прогрессивные фотонные карты;
- Startup Bias – начальное смещение. Ошибка в изображении, проявляющаяся в виде неправильной оценки яркости отдельных областей изображения (например, недостаточно яркий каустик или наоборот яркая область, которая в процессе расчёта будет темнеть). Проблема свойственна методам на основе MLT.
- strMCMC-LT -- Stratified Markov Chain Monte Carlo Light Transport [83];
- SVBSDF – Spatial Varying Bidirectional Scattering Distribution Function [109], вид BSDF, когда свойства поверхности заданы в текстурах, т.е. могут различаться для разных точек. Например, маска смешения двух материалов или параметр ``glosiness'', заданный в текстуре, позволяют классифицировать BSDF как SVBSDF.
- UBPT – Unifying points, Beams and Paths in volumetric light transport simulation [26];
- VCM – Vertex Connection and Merging [24];
- Veach MLT (или Path Space MLT) – явное указание на то, что MLT реализован в мировом пространстве путей, как в оригинальной работе Вича [73].

Кроме того, нам потребуются ещё несколько определений:

Надёжность. Алгоритм расчёта освещения называется *надёжным* (*robust*) [3] на определённом сценарии освещения, если при расчёте интеграла отсутствуют выбросы – редкие Монте-Карло выборки с крайне большими значениями, препятствующие сходимости расчёта за приемлемое время. Надёжность является крайне важной характеристикой, т.к. более надёжные методы позволяют посчитать более сложные сценарии освещения, которые мы будем иногда называть *трудновычислимыми*. Таким образом, надёжность является краеугольным камнем эффективности расчёта освещения.

Сходимостью Монте-Карло метода мы будем называть функцию $C(\dots)$, обратно пропорционально которой убывает ошибка. Например, сходимость $C(N) = \sqrt{N}$, где N – число выборок, означает, что ошибка убывает пропорционально $\frac{1}{\sqrt{N}}$. В этом случае, если мы хотим увеличить точность метода в 10 раз по сравнению с определённым значением, придётся увеличить количество выборок в 100 раз.

Эффективностью метода расчёта освещения будем называть далее:

i. процент Монте-Карло выборок, вносящих существенный вклад в изображение, для методов на основе OMC; *существенным* вкладом будем называть такой вклад, яркость которого сопоставима по порядку со средней яркостью изображения или значительно больше её;
ii. среднюю вероятность принятия предложения перехода (acceptance rate) для методов на основе MCMC.

3. Общая классификация

Развитие методов вычисления освещённости шло, в основном, двумя путями на основе:

i. обыкновенного Монте-Карло интегрирования (Ordinary Monte Carlo, OMC);
ii. Монте-Карло интегрирования по схеме марковских цепей (Markov Chain Monte Carlo, MCMC) [1].

Обе группы методов в настоящее время успешно применяются для вычисления интеграла освещённости и решения так называемого уравнения рендеринга [2] и имеют свои плюсы и минусы. На рис. 1 представлена приблизительная карта методов и их аббревиатуры, которые мы будем раскрывать по ходу статьи.

Кроме рассмотренной классификации на OMC и MCMC возможна, как минимум, ещё одна независимая классификация:

i. методы, работающие с тонкими лучами в терминах яркости;
ii. методы, работающие с элементами конечного размера в терминах светового потока.

Такое разделение возможно как для методов на основе OMC, так и для методов на основе MCMC. Однако, поскольку методы, работающие в терминах светового потока, используются в основном в обыкновенном Монте-Карло, мы будем рассматривать их в разделе методов на основе OMC, а позже сделаем ремарку про применение этих методов в MCMC (подраздел 5.9, Markov Chain PPM).

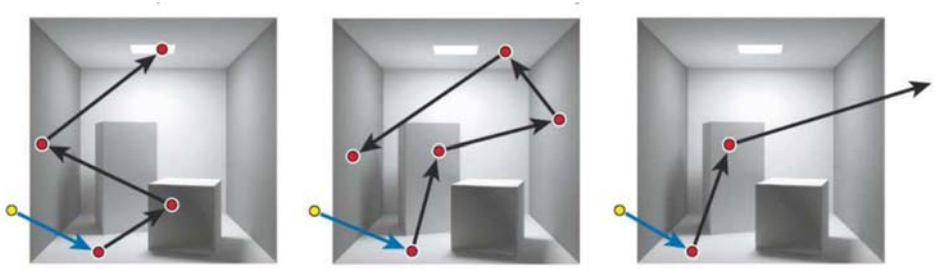

Рис. 1. Трассировка путей (Path Tracing)
Fig. 1. Path Tracing

4. Методы на основе OMC

4.1 Методы, работающие в терминах яркости

Ключевым механизмом, на основе которого строятся все современные методы интегрирования освещённости на основе OMC, является многократная выборка по значимости (Multiple Importance Sampling, MIS) [3]. Для того чтобы объяснить её суть, мы подробно рассмотрим применение MIS на примере самого базового метода – трассировки путей (Path Tracing, PT) [2].

4.1.1 Path Tracing (PT)

В простейшем варианте трассировки путей луч путешествует по сцене случайно до тех пор, пока не попадёт в источник света, не выйдет за пределы сцены или не будет достигнута заданная максимальная глубина рекурсии (рис. 2).

На рис. 3 такой алгоритм обозначен как «Simple PT» (или Naive PT). Как не трудно догадаться, основная проблема этого метода в том, что на типичных сценах вероятность случайного попадания в источник света крайне мала. Поэтому обычно в трассировку путей добавляют теневые лучи (shadow rays или visibility tests), которые при каждом переотражении летят напрямую в источник («Shadow PT» на рис. 3). Для потолочных светильников такой вариант, как правило, существенно улучшает ситуацию. Однако, на самом деле одного лишь «Shadow PT» недостаточно, даже если мы рассматриваем только прямое освещение. Shadow PT не работает, когда источник света имеет крупный размер, и/или материал освещаемой поверхности имеет глянцевые отражения (glossy reflections). В этом случае лишь небольшой участок источника будет вносить вклад в освещение, в то время как Shadow PT генерирует выборки по всей поверхности источника. Классический пример такой сцены: стол с глянцевой поверхностью, освещаемый окном. Оказывается, что в этом случае, как ни странно, Simple/Naive PT работает существенно лучше, поскольку выпускаемые им лучи учитывают свойство материала и почти всегда попадают в источник, внося ненулевой вклад в изображение.

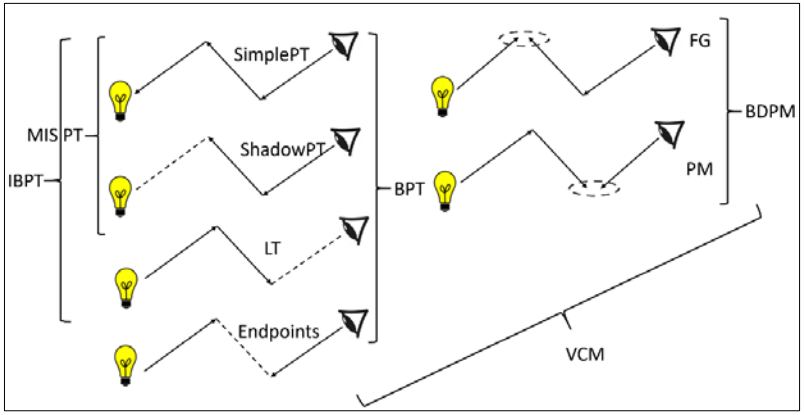

Рис. 3. Отдельные стратегии генерации выборок и методы, которые строятся на основе их комбинации

Fig. 3. Separate strategies for generating samples and methods that are based on their combination

Расмотренные нами алгоритмы используют две различные стратегии создания выборок: Simple PT использует *неявную* стратегию, а Shadow PT – *явную*, выпуская лучи непосредственно в источник света. При этом следует отметить, что даже для прямого освещения недостаточно использования какой-либо одной стратегии: явной или неявной. Поэтому в трассировке путей используют обе, и такой алгоритм называется MIS PT (рис. 3). Он комбинирует вклады от явной (теневых лучей) и неявной (лучей, случайно попавших в источник) стратегий при помощи многократной выборки по значимости. Это, однако, требует вычисления весов выборки в многократной выборке по значимости, что на самом деле является нетривиальным решением по двум причинам.

(a) Вычисление весов выборок в многократной выборке по значимости требует от реализации функций генерирования выборок для материалов (неявной стратегии) и источников (явной стратегии) корректного вычисления плотностей вероятности для каждой выборки. Это существенно усложняет дизайн функций генерировании выборок материалов и источников и делает его доступным только для узких специалистов [4], поскольку требует знания о плотностях вероятности всех стратегий. Причём весьма нетривиальным образом: когда бы ни была вычислена некоторая выборка какой-то одной стратегией, для этой конкретной выборки необходимо уметь вычислять плотности всех других стратегий, то есть *как если бы* именно эта выборка *была бы* вычислена другими стратегиями.

(b) Усложняется верификация MIS PT, т.к. для многих источников и материалов (точечный и направленный источники, источник в виде панорамы окружения и так называемые «небесные порталы», зеркальный материал, смесь материалов и др.) в коде появляется обработка специальных случаев. Причина этого в том, что, например, для идеального зеркального отражения невозможно корректно вычислить плотность вероятности [5]. В результате приходится прибегать к хитрости: в MIS PT зеркальное отражение форсированно делает вес неявной стратегии равным единице, а явной равным нулю. В двунаправленных методах, рассматриваемых далее, считают, что какой бы ни была плотность вероятности при зеркальном отражении, в прямом и обратном направлении она будет одинакова. Поэтому её можно задавать равной единице, но при этом нужно не забыть превратить её в ноль, если встречается явный способ соединения вершин.

Рассмотренный алгоритм трассировки путей с применением MIS используется в подавляющем большинстве индустриальных систем расчёта освещения в компьютерной графике (в архитектуре, кино и мультипликации). К сожалению, он даёт возможность сделать надёжный расчёт лишь прямого освещения. Как только в 3D сцене появляется существенное влияние непрямого (вторичного) освещения, возникает необходимость в добавлении новых стратегий построения выборок.

4.1.2 Bidirectional Path Tracing (BPT)

Рассмотрим сначала частный случай BPT – алгоритм *усечённой* двунаправленной трассировки, который в английском языке называется Instant Bidirectional Path Tracing (IBPT) [6]. Идея IBPT состоит в том, чтобы к двум существующим в MIS PT стратегиям добавить ещё одну: световую стратегию (LT на рис. 3). Эта стратегия работает аналогично Shadow PT, но в прямом направлении: луч стартует на источнике света, и при каждом переотражении от поверхности производится явное соединение этой точки поверхности с камерой. За счёт этой стратегии IBPT на удивление хорошо справляется со сложным вторичным освещением на ламбертовских поверхностях, но не может эффективно рассчитывать каустики, видимые через зеркала и стёкла (SDS-пути) [6].

Полноценный BPT устроен сложнее. Из источника и камеры трассируются два пути глубиной N и M переотражений соответственно. После чего делается $N \times M$ соединений между вершинами этих путей. Далее, используя полученные $N \times M$ соединений, формируются все полные пути от источника до камеры. Для каждого полного пути его вклад учитывается на основе многократной выборки по значимости.

Следует отметить, что вычисление вклада конкретного полного пути на основе веса многократной выборки по значимости в BPT никак не связано с переиспользованием вершин во время выполнения $N \times M$ соединений. То есть веса вычисляются для каждого полного пути независимо от других полных путей. Таким образом, BPT можно было бы строить иначе, соединяя лишь конечные точки: сначала выбрать случайно глубину от 0 до N и протрассировать путь от источника на глубину N (нулевое значение кодирует стратегию SimplePT, в которой мы будем пытаться поймать источник камерным лучём); затем протрассировать путь от камеры на случайно выбранную глубину от 0 до M (нулевое значение соответствует стратегии Light Tracing); наконец, соединить конечные точки путей от камеры и источника. Однако такой алгорим (мы называем его End-Points BPT) сам по себе неэффективен вследствие более низкой вероятности получить удачный путь, априорно

выбирая глубину трассировки N и M. Тем не менее, именно End-Points BPT является базовым расчётным методом в MMLT [78], где его эффективность меняется кардинальным образом благодаря марковским цепям.

Таким образом, IBPT [6] и PCBPT [7] могут рассматриваться как оптимизации оригинального алгоритма двунаправленной трассировки путей (BPT [3]). Они достигают лучшей производительности за счёт того, что реже используют (PCBPT) или совсем не используют (IBPT) стратегии с промежуточными соединениями («End-points» на рис. 3), которые редко являются удачными. Все три метода тем не менее имеют следующие недостатки.

(a) Двунаправленные методы требуют выполнения симметрии моделей. В действительности, это довольно существенное ограничение, поскольку источником ассиметрии могут быть входные данные (раздел «The Sources of Non-Symmetric Scattering» [3]): особенности геометрической модели поверхности [10], необходимость учёта поляризации, определённой только в одном направлении [11-12], а также ассиметрия, возникающая при расчёте преломлений в специфических средах [13]. Кроме того, расчёт такого явления как глубина резкости (Depth of Field, DOF) в BPT для LT стратегии затруднён из-за нарушения симметрии в силу отсутствия эффекта «огибания объектов» при проектировании точки на экранную плоскость или поверхность линзы объектива, что проводит к артефактам в виде тёмных краёв объектов. Это происходит из-за того, что не все 100% выборок, проецируемых в камеру с некоторой «дальней поверхности», в действительности достигнут её объектива, т.к. теневой луч может быть перекрыт близлежащим объектом. При расчёте же в обратном направлении все 100% лучей так или иначе достигнут какой-либо из двух рассматриваемых поверхностей – либо близлежащую, либо упомянутую ранее дальнюю поверхность.

(b) Верификация двунаправленных методов (особенно BPT и PCBPT, содержащих стратегии с промежуточными соединениями) становится ещё более сложной. Необходимо тестировать большое количество случаев, когда разные стратегии вносят существенный вклад в изображение. Тестирование при помощи покрытия кода не помогает, т.к. важно, чтобы рассматриваемый участок кода не просто выполнился достаточно большое число раз, но и статистически внёс существенный вклад в изображение. А это не просто гарантировать из-за того, что вес в многократной выборке по значимости может обнулить вклад от той или иной стратегии в другом месте кода, никак не связанной с текущей стратегией (то есть важно тестировать стратегии ещё и в сочетании друг с другом). Кроме того, в двунаправленных методах необходимо все плотности вероятности вычислять в площадной мере (вероятность / м^2) [5], что увеличивает количество упомянутых ранее специальных случаев.

Мы полагаем, что метод IBPT в целом более практичен чем BPT или PCBPT в первую очередь из-за того, что он компактный по памяти. А это существенно для реализации расчёта на GPU. Многие существующие реализации используют именно этот тип двунаправленной трассировки путей.

4.1.3 Проблема множества источников

Большое число источников света обычно является проблемой для однонаправленных методов и требует применения специальных алгоритмов для реализации эффективной выборки по значимости [8-9]. Однако интересно отметить, что алгоритмы IBT, BPT и PCBPT могут эффективно вычислять освещение при большом числе источников благодаря использованию прямой стратегии: из какого бы источника света луч не был выпущен, он, как правило, вносит не нулевой вклад в изображение. Исключением будут являться сцены, состоящие из большого количества закрытых комнат, в которых необходимо применять те же методы повышения эффективности выборки источников что и для однонаправленных методов [8-9].

4.1.4 SDS каустики

Ранее мы упомянули, что каустики, видимые в стекле или зеркале (называемые нами *каустики второго типа*), не могут быть эффективно вычислены при помощи BPT. Однако, в недавней работе [14] была предложена стратегия, позволяющая решить эту проблему. Для этого авторы [14] используют исследование многообразий [15-16]. Ядром этого метода является эффективный расчёт так называемого обобщённого геометрического члена, который связывает две диффузные (или глянцевые) поверхности через последовательность зеркальных переотражений. В [14] предложена аппроксимация расчёта обобщённого геометрического члена, благодаря чему метод обретает практическую значимость.

Наиболее существенной проблемой метода [14] (как и [15-16]) является то, что для его работы требуется наличие инструментария дифференциальной геометрии, что является сильным ограничением, т.к. непосредственно влияет на то, как именно геометрические модели представлены внутри рендер-системы. Таким образом, это является классическим примером того, что более эффективный метод глобального освещения налагает существенно больше ограничений и, таким образом, оказывается труднопременим в определённых условиях на практике (например, при наличии карт смещений и карт нормалей, о чём упоминают авторы [14-16].

4.2 Методы, работающие в терминах светового потока

Второй класс методов – это фотонные карты. Он переходит от несмещенной оценки интеграла к смещенной (в стандартном определении фотонных карт) [17] и состоятельной для прогрессивных алгоритмов, таких как SPPM [19]. Фотонные карты делают это путем введения радиуса сбора конечного размера, в пределах которого близко расположенные точки световых путей могут быть объединены вместе, как если бы это была одна точка. Такое свойство алгоритма (смещенная / состоятельная оценка) на практике приводит к тому, что приближенное решение может быть получено быстрее чем несмещёнными методами, но точные вычисления становятся проблемой. В итоге фотонные карты часто используются в демонстрационных приложениях, но на практике есть существенные недостатки.

(a) Более медленная сходимость: $O(\frac{1}{\sqrt[3]{N}})$ по сравнению с $O(\frac{1}{\sqrt{N}})$ для BPT [21], что происходит за счет уменьшения радиуса сбора в прогрессивных методах.

(b) Дорогостоящая операция сбора (особенно в произвольном случае, когда требуется оценка BSDF для каждого фотона), которая становится критической из-за наличия сгустков фотонов. Контроль плотности [22] амортизирует эту проблему, но его (как и метод релаксации фотонов [20] можно использовать лишь для ограниченного случая: сбор фотонов для ламбертовской поверхности.

(c) Фотонные карты должны применяться с осторожностью на поверхностях с микрорельефом, поскольку могут искажать вид микрорельефа.

(d) Фотонные карты требуют построения структур пространственного разбиения (ускоряющих структур) для процедуры сбора освещённости, что не является серьезным недостатком, но значительно усложняет параллельную реализацию и требует дополнительный объем памяти по сравнению с подходами на основе BPT.

Тем не менее, несмотря на упомянутые недостатки фотонные карты широко используются в индустриальных и научных системах расчёта по следующим причинам.

(a) Фотонные карты могут рассчитывать довольно сложные случаи глобальной освещённости и быстро получать для них приближённое решение.

(b) Реализация фотонных карт проста по сравнению, например, с BPT и сама по себе не требует вычисления плотностей вероятностей для каждого материала и источника.

4.3 Объединение BPT и фотонных карт

Преимущества фотонных карт (PM) побудили исследователей к созданию методов, объединяющих в себе BPT и PM. На сегодняшний день существует два достаточно стандартных способа объединения.

(a) Выполнить сбор освещённости на первом незеркальном переотражении луча во время обратной трассировки (рис. 3, PM). Этот способ широко используется для расчёта каустиков, для которых, как правило, заводится отдельная фотонная карта [23-24].

(b) Выполнить сбор освещённости после второго незеркального переотражения луча во время обратной трассировки (рис. 3, FG). Этот способ называется финальным сбором (Final Gathering, FG). Фактически, фотонные карты в финальном сборе используются как аппроксимация третьего переотражения, и в таком качестве работают очень хорошо для многих сцен, поскольку, как мы обсуждали ранее, фотонные карты позволяют получить приближённое решение быстро.

4.3.1 BDPM

Однако же комбинации рассмотренных способов недостаточно, если в сцене встречаются материалы с глянцевыми отражениями. Кроме того, финальный сбор даёт артефакты в углах геометрических объектов, где его откладывают как минимум ещё на 1 переотражение (вторичный финалный сбор [23]). Подобные проблемы привели к появлению методов, делающих выбор номера переотражения на основе анализа оптических свойств текущей поверхности [33, 35-36]. К сожалению, легко показать, что в общем случае такого анализа недостаточно, т.к. заранее неизвестно насколько далеко будет располагаться следующая поверхность. Поэтому единственным решением, гарантирущим надёжность в данном случае, является сбор на каждом переотражении и комбинация результатов от разных переотражений при помощи многократной выборки по значимости. Именно так поступает алгоритм двунаправленных фотонных карт (Bidirectional Photon Mapping, BDPM) [37] (рис. 3, справа). Однако за повышение надёжности здесь, как и в BPT, приходится расплачиваться понижением скорости в среднем, т.к. дорогостоящий сбор освещённости теперь выполняется на каждом переотражении. Поэтому на наш взгляд имеет смысл также иметь ввиду варианты комбинирования BPT и фотонных карт без многократной выборки по значимости, таких как PCLT [30] и упомянутых методов на основе анализа свойств поверхности [33, 35-36]. Хотя по сравнению с полноценным BDPM они не столь универсальные и надёжные, что на практике приводит к настройке параметров алгоритма пользователем.

Заметим, что метод BDPM, несмотря на название, не является объединением классических [17, 23] и обратных фотонных карт [31-32], поскольку в обратных фотонных картах в действительности инвертируется лишь геометрическая задача поиска ближайших фотонов, а сами вычисления интеграла и стратегии генерации выборок при этом не изменяются.

4.3.2 VCM

Наконец, по-настоящему промежуточное положение между первым (BPT) и вторым (PM) классами занимают методы, которые интегрируют фотонные карты в BPT на основе MIS: VCM [24], PEPM [25], UBPT [26]. При этом многократная выборка по значимости в этих методах не решает проблему дорогой операции сбора, т.к. работает апостериорно, т.е. уже после того, как сбор освещённости был выполнен. Проблема меньшей скорости сходимости также решается не полностью, поскольку для трудновычислимых феноменов освещённости, с которыми не справляется BPT, будет наблюдаться ухудшенная сходимость фотонных карт – $O(\frac{1}{\sqrt[3]{N}})$.

Отдельным пунктом необходимо упомянуть метод Path Space Regularization [27], который является аналогом VCM, но вместо классического сбора в пространстве использует «сбор по

углам», так называемую угловую регуляризацию. Если говорить упрощённо, Path Space Regularization заменяет идеально-зеркальные BSDF на глянцевые и постепенно уменьшает глянцевость, двигаясь с каждым новым проходом обратно к зеркалу.

На сегодняшний день VCM является чрезвычайно популярным методом. Однако, хотя с математической точки зрения VCM и его аналоги делают большой шаг вперёд и повышают надёжность расчёта (что позволяет применять метод на более широком классе сцен), на практике зачастую эти методы (например, VCM, реализованный в системе Corona [28]) являются лишь небольшим улучшением простого смешивания изображений от BPT и SPPM по маске [29]. Это является вполне очевидным следствием механизма работы многократной выборки по значимости – сначала вычислить результат двумя разными способами, и потом скомбинировать оба с весами, которые часто вырождаются в пару (0,1) или (1,0). С другой стороны, трудоёмкость реализации на основе MIS достаточно высока [38] («... so much so that the authors also released a technical report and source code explaining how to implement the VCM algorithm...»). Верифицировать VCM или тем более UBPT крайне сложно в силу огромного количества рассматриваемых вариантов комбинирования стратегий в многократной выборке по значимости.

4.4 Адаптивная генерация выборок

Идея управляемых путей (Path Guiding) строится на аппроксимации входящего освещения в различных точках поверхности каким-либо из существующих методов (фотонные карты, функции специальнгого вида или машинное обучение) с тем, чтобы использовать эту аппроксимацию как подсказку для выборки по значимости [42-46]. Path Guiding уменьшает уровень шума в целом по изображению, однако легко пропускает мелкие геометрические детали, которые функция аппроксимации не может учесть. Из-за этого на итоговом изображении присутствует разрозненный импульсный шум. Тем не менее Path Guiding становится популярным в индустрии, т.к. считается, что его относительно просто интегрировать в существующие системы [46], и, в отличие от MLT, он может быть использован для вычисления первичного освещения.

Path Guiding можно рассматривать как частный случай адаптивной генерации выборок, которая, в свою очередь, является частным случаем области компьютерной графики, называемой Sampling and Reconstruction [47]. Следуя классификации [47], все методы Sampling and Reconstruction делятся на априорные и апостериорные (хотя существуют методы, занимающие промежуточное положение [48] между этими двумя классами). Априорные методы строят адаптивные стратегии генерации выборок, помещая больше выборок в более сложные области изображения или пространства интегрирования. Апостериорные методы удаляют шум, работая на выходе системы расчёта освещённости, и не влияют на сам процесс построения изображения.

Однако, несмотря на достаточно высокую степень развитости, большинство априорных методов, рассматриваемых в [47], не могут быть использованы для построения эффективных алгоритмов рендеринга. Причина этого в том, что они используют один базовый алгоритм расчета. Если этот базовый метод неэффективен в некоторой области изображения или многомерного пространства интегрирования, априорные методы стараются помещать больше выборок в эту область, чтобы «задавить» шум количеством. Такой подход, очевидно, имеет определенный недостаток – при наличии трудно вычислимых феноменов освещённости адаптивная генерация выборок начинает сосредотачивать большую часть вычислительных ресурсов в областях пространства, где расчёт неэффективен. При этом страдает качество в остальных участках изображения, а трудно вычислимые области по-прежнему остаются шумными. Исключением является работа [49] и упомянутый ранее Path Guiding, в которых адаптивная генерация выборок производится в пространстве высокой размерности. Однако размерность пространства в [49] всё же ограничена небольшой

величиной (4D–5D), и отмечается, что для более высокой размерности возникнут проблемы, которые ещё только предстоит решить. Методы на основе MCMC, рассматриваемые в разд. 5, выполняют адаптивную генерацию выборок в многомерном пространстве автоматически.

Интересно отдельно обозначить метод RIS [58], суть которого в том, что можно использовать пространственную (соседние пиксели) или темпоральную (соседние кадры) когерентность пространства интегрирования. RIS объединяет использованные распределения от нескольких соседних пикселей (и/или кадров) изображения для того, чтобы сформировать новое распределение выборок для данного пиксела. Таким образом, хорошее распределение выборок в RIS получается как взвешенная сумма большого числа распределений из соседних пикселей или кадров. В настоящий момент метод применяется только для первичного освещения и работает в пространстве экрана [58].

4.5 Рендеринг в пространстве градиентов

Идея рендеринга в пространстве градиентов состоит в том, чтобы одновременно с обычным изображением генерировать выборки ещё и для изображения градиентов. Полагая далее, что градиент посчитать проще (т.к. он разреженный по сравнению с обычным изображением), можно восстанавливать итоговое изображение по градиентам. В [50] был представлен алгоритм, названный Gradient-domain Path Tracing (GDPT), и показано улучшение относительно трассировки путей.

Metropolis Light Transport, к которому мы будем неоднократно возвращаться, обычно строит распределение пропорционально некоторой целевой функции освещения, которая имеет линейную зависимость с изображением. В работе [54] эта идея была расширена на нелинейную зависимость между изображением и целевой функцией. Генерация выборок производилась пропорционально градиентам изображения, а само изображение реконструировалось при помощи решения уравнения Пуассона. Впоследствии в работах [51-52] рендеринг в пространстве градиентов был расширен на двунаправленную трассировку путей, а в [53] и на фотонные карты.

На наш взгляд преимущества и недостатки рендеринга в пространстве градиентов в целом схожи с преимуществами и недостатками, получаемыми от простого применения апостериорных методов шумоподавления (рассматриваемых в [47] к изображению или отдельным выборкам, поскольку в обоих случаях используется идея «меньше считать, больше восстанавливать». Даже характер артефактов схожий. С другой стороны, реализация рендеринга в пространстве градиентов существенно усложняет программную систему по сравнению с применением методов шумоподавления. По этой же причине на наш взгляд не стоит подробно рассматривать методы кэширования освещённости [55-56].

4.6 Заключение по OMC

Таким образом, для методов на основе OMC существует много эвристических подходов, пытающихся улучшить расчёт отдельных феноменов и хорошо работающих на определённом классе сцен, обычно, в условиях сильных ограничений. Однако можно сказать, что эти подходы на практике трудно применять, поскольку пространство интегрирования многомерное и крайне сложное: в таком пространстве любое предположение, сделанное эвристикой рано или поздно перестанет выполняться. Единственным методом, обеспечивающим надёжность расчёта в таком случае, остаётся многократная выборка по значимости. Но чем более метод надёжен, тем он сложнее в реализации и тестировании и, как правило, медленнее в среднем. Что касается ограничений, постановка задачи зачастую не позволяет их ввести. Это приводит к необходимости использовать разные методы в разных случаях и чрезвычайно усложняет жизнь как разработчикам, так и пользователям систем расчёта освещения.

На сегодняшний день итогом развития методов на основе многократной выборки по значимости (MIS) можно считать метод CMIS [57], который позволяет расширить MIS на континуум стратегий генерации выборок (представленный, как правило, набором параметрических функций), что улучшает сходимость в определённых случаях. Однако проблема оптимального выбора стратегий в CMIS не решается сама по себе. Алгоритм, который решает эту проблему, называется Multiplexed Metropolis Light Transport и рассматривается нами разд. 5.

4.7 Реализации на GPU

Мы полагаем, что исследования эффективности алгоритмов на GPU крайне важны для современных практических приложений, особенно с учётом появления аппаратного ускорения трассировки лучей в современных GPU [59-60]. Поэтому мы не можем обойти вниманием эти работы.

Для эффективной реализации методов расчёта освещения на GPU важно уделять внимание нескольким вещам.

(a) Важно уменьшать потребление памяти на 1 поток, поскольку число запускаемых потоков (которые выполняют несколько вычислительных ядер последовательно) на GPU может исчисляться сотнями тысяч. Например, в [61] особое внимание уделяется экономии памяти для BPT. Для этого предлагается специальный механизм расчёта весов многократной выборке по значимости, для которого не нужно хранить все вершины пути в памяти. А в [6] предлагается использовать усечённую двунаправленную трассировку путей на GPU.

(b) Необходимо стремиться уменьшить количество дивергентных потоков и решать проблему нерегулярного распределения работы, что обычно делается при помощи регенерации, уплотнения и сортировки потоков, а также разбиения сложного кода на набор простых вычислительных ядер [62-64].

(c) Для методов на основе фотонных карт нужно использовать быстрые методы построения пространственных структур данных на GPU для поиска ближайших фотонов на основе хэш-таблиц или деревьев, использующих коды Мортона [65-69]. Отдельно следует отметить здесь работу [70], где в отличие от предыдущих работ все уровни дерева строятся параллельно.

Интересно отметить работу [71], где на GPU был реализован алгоритм VCM. Для эффективной реализации VCM в сочетании с BPT предлагается специальная структура данных, названная Light Vertex Cache, которая позволяет одновременно решать первую и третью проблему из данного списка.

Наконец, нельзя обойти вниманием работу [72], в которой предлагается механизм эффективной диспетчеризации GPU ядер, поддержка рекурсии при помощи сохранения данных для ядер в раздельных очередях (для каждого ядра своя очередь) и использования механизма создания работы на GPU. Важно, что механизм, предлагаемый в [72], решает проблему дивергентных потоков, поскольку диспетчеризация фактически производится без ветвления: сначала для програмы/ядра каждого типа в очереди набирается достаточно большое число потоков, после чего они все выполняют одну программу/ядро.

5. Методы на основе MCMC

Монте-Карло по схеме марковских цепей (Markov Chain Monte Carlo, MCMC) можно рассматривать как обобщение обыкновенного Монте-Карло. Если в обыкновенном Монте-Карло выборки независимы, то в MCMC они наоборот коррелированы. Благодаря этому в MCMC в отличие от OMC информацию об областях функции с высокой значимостью удаётся переиспользовать. Один из наиболее распространённых вариантов MCMC – алгоритм Метрополиса и его более общий вариант, алгоритм Метрополиса-Гастингса. Цель всех

MCMC алгоритмов заключается в том, чтобы построить распределение выборок пропорционально произвольной целевой функции.

Таким образом, в противовес методам, основанным на многократной выборке по значимости, алгоритм Метрополиса для расчёта освещённости (Metropolis Light Transport, MLT) [73] генерирует выборки не пропорционально какой-либо одной части подынтегральной функции (что делает каждая из стратегий BPT), а пропорционально всему интегралу освещённости как многомерной функции, спроецированной на множество пикселей изображения: $F(x, y, r_0, r_1, \dots, r_n) \xrightarrow{project} F(x, y)$. MLT автоматически помещает больше выборок в более сложные участки функции F, значительно уменьшая дисперсию [76]. Вопрос сходимости для MLT более сложный. В частности, в [74] приводится оценка $O(\gamma^N)$, где $\gamma \in (0; 1]$.

Следует, однако, отметить, что многократная выборка по значимости и алгоритм Метрополиса *могут и должны использоваться вместе*. Именно так и было сделано в самой первой работе по MLT [73], где алгоритм Метрополиса был предложен для BPT, и использованы переходы *в пространстве путей* (path space) как небольшие изменения позиций вершин пути.

5.1 Основы MLT

Прежде чем перейти к дальнейшему рассмотрению методов на основе MCMC, необходимо оговорить условия, в которых эти методы позволяют получить корректный результат. Здесь есть несколько принципиальных моментов.

(a) Ни один из MCMC методов не позволяет напрямую посчитать интеграл освещённости. Эти методы строят многомерное распределение выборок и накапливают в двумерном массиве гистограмму, которая пропорциональна целевому изображению, но не равна ему. Для того чтобы получить итоговое изображение, эту гистограмму нужно скалировать – то есть умножить на некоторую константу нормализации. Оценить константу нормализации можно, например, если посчитать среднюю яркость изображения – средняя яркость скалированной гистограммы должна быть такой же. Это можно сделать при помощи расчёта грубого изображения любым из методов на основе OMC. Таким образом, OMC и MCMC всегда работают вместе.

(b) Для того чтобы MCMC методы работали корректно, марковский процесс должен быть эргодичным. Это означает, что всё пространство состояний должно быть достижимо во время случайных блужданий марковской цепи (для алгоритма Метрополиса, где прямые и обратные переходы равновероятны). Все предложения переходов разбиваются на 2 части: большие и маленькие шаги. Маленькие шаги являются той самой «рабочей лошадкой», которая обеспечивает эффективную генерацию выборок, используя тем или иным образом локальность интегрируемой функции в многомерном пространстве. Большие же шаги, как правило, призваны лишь обеспечить эргодичность. На практике, однако, низкая вероятность принятия большого шага может приводить к росту начального смещения (startup bias). Отдельно взятые большие шаги в MLT фактически являются базовым расчётным методом на основе OMC и могут служить, например, для оценки константы нормализации.

(c) Каждый шаг марковской цепи добавляет близкий к единичному (либо даже строго единичный) вклад в гистограмму изображения (пункт 1). Из этого необходимо сразу сделать вывод: при помощи Metropolis Light Transport не стоит считать первичное освещение, т.к. он просто «уткнётся» в блики и другие яркие области (например, источник), и практически никогда не будет из них выходить, набирая значения в ярких областях единичками.

5.2 Paths Space vs Primary Sample Space

MCMC методы в рендеринге можно условно разделить на два класса по тому, в каком пространстве они работают. К первому классу относятся методы, работающие в *мировом пространстве путей* (Path Space; мы будем также называй такой тип VeachMLT) – пространстве, составленном конкатенацией всех координат вершин путей в мировом пространстве (например, это Veach MLT [73], MEMLT [15], HSLT [77] на рис. 1). Ко второму классу можно отнести методы, работающие в так называемом *первичном пространстве путей* (Primary Sample Space; мы будем также называй такой тип KelemenMLT) – пространстве всех случайных чисел, используемых Монте-Карло выборкой, т.е. многомерном единичном кубе (например, это Kelemen MLT [75] и MMLT [78] на рис. 1). На рис. 4 приведена иллюстрация обоих пространств. В практических приложениях, как правило, используются методы второго класса, т.к. они более универсальны по отношению к различным феноменам освещённости. Path Space методы, напротив, более специфичны. Например, MEMLT хорошо работает с SDS-путями, а HSLT разработан для многократных glossy-отражений.

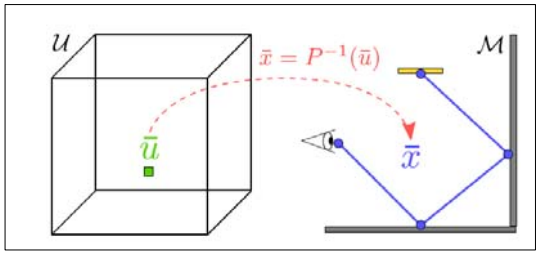

Рис. 4. Иллюстрация перехода из первичного пространства путей U (слева, представленное многомерным единичным кубом) в мировое пространство путей M (справа, представленное конкатенацией всех позиций вершин пути в мировом пространстве). Отметим, что в работах [88-90] необходимо уметь строить обратный переход $\bar{u} = P(\bar{x})$, что является нетривиальной операцией. Использован рисунок из работы [78]

Fig. 4. An illustration of the transition from the primary path space U (on the left, represented by the multidimensional unit cube) to the world path space M (on the right, represented by the concatenation of all positions of the vertices of the path in world space). Note that in [88-90] it is necessary to be able to construct the inverse transition $\bar{u} = P(\bar{x})$, which is a nontrivial operation. Used drawing from work [78]

Наиболее существенным недостатком MLT в пространстве путей является то, что под каждый феномен освещённости необходимо не только тщательно проектировать специфические стратегии пертурбаций, но и уметь вычислять плотности вероятности переходов – $T(x \leftarrow y), T(x \rightarrow y)$, для правила Метрополиса (формула 5.1):

$$a(x \rightarrow y) = \frac{f(y)T(x \leftarrow y)}{f(x)T(x \rightarrow y)}. \quad (5.1)$$

Современные системы синтеза реалистичных изображений содержат десятки различных типов материалов и источников освещения, что приводит к сотням всевозможных типов взаимодействий (феноменам освещённости). Это чрезвычайно усложняет процесс разработки вышеупомянутых стратегий и процесс вычисления плотностей вероятности $T(x \rightarrow y)$ и $T(x \leftarrow y)$, поэтому MLT в пространстве путей практически не применяется на практике, а используется в основном в исследовательских и демонстрационных приложениях. В Primary Sample Space MLT, с другой стороны, прямая и обратная вероятности (вернее, плотности вероятности) одинаковы. Из этого следует, что $T(x \rightarrow y)$ и $T(x \leftarrow y)$ в формуле 5.1 сокращаются и, следовательно, их не нужно вычислять.

5.3 Multiplexed MLT (MMLT)

Как уже было отмечено, многократная выборка по значимости и алгоритм Метрополиса **могут и должны использоваться вместе**. Однако это можно сделать разными способами – Veach MLT [73], Kelemen MLT [75], MMLT [78]. Ключ к успеху MMLT лежит в построении такого пространства интегрирования, в котором алгоритм Метрополиса и многократная выборка по значимости не конкурируют, а усиливают друг друга.

Для того чтобы комбинировать алгоритм Метрополиса и многократную выборку по значимости более эффективно, в MMLT [78] строится «мультиплексированное» пространство интегрирования. Это происходит путём добавления двух степеней свободы: глубины трассировки d и стратегии генерации выборки пути s. Далее марковская цепь статистически находит оптимальный способ построения пути в ВРТ, варьируя параметр s (рис. 5, формула 5.2). Благодаря этому алгоритм Метрополиса автоматически перераспределяет вычислительные ресурсы таким образом, что малозначимые стратегии и соединения в ВРТ считаются редко. Причём это происходит в том числе и с учётом функции видимости, т.к. алгоритм Метрополиса строит распределение пропорционально итоговому ответу.

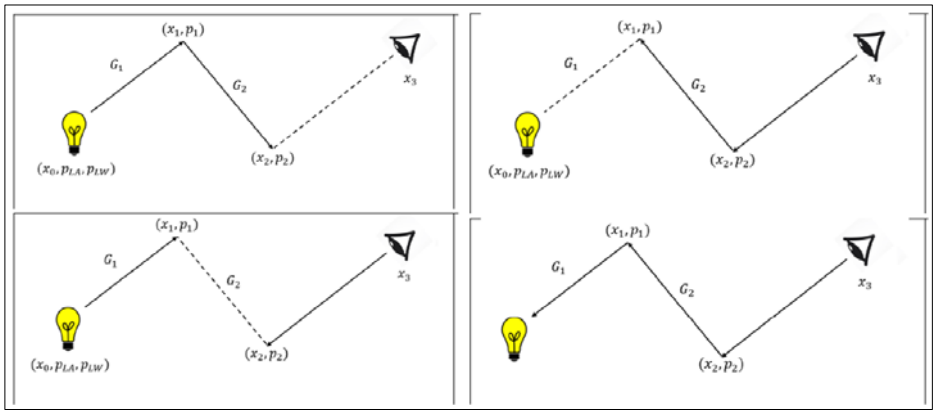

Рис. 5. Пример четырех стратегий для глубины трассировки, равной 3. Пунктирная черта изображает явное соединение вершин. Оптимальный выбор стратегии происходит автоматически алгоритмом Метрополиса, т.к. функция вклада в MMLT построена в виде взвешенной суммы различных стратегий (формула 5.2)

Fig. 5. An example of four strategies for a tracing depth of 3. The dashed line represents an explicit vertex connection. The optimal choice of strategy occurs automatically by the Metropolis algorithm, since the contribution function in MMLT is built as a weighted sum of various strategies (formula 5.2)

$$C(\bar{u}) = \sum_{i=1}^{4} \hat{\omega}_i \, \hat{C}_I^*(\bar{u}). \qquad (5.2)$$

Здесь $\hat{C}_I^*$ –результат выборки i-ой стратегии, а $\hat{\omega}_i$ – соответствующий ей вес.

Более передовые методы, как правило, построены поверх MMLT и нацелены на его улучшение или устранение его проблем. Эти проблемы происходят из-за того, что мультиплексированное пространство интегрирования в MMLT, благодаря которому алгоритм Метрополиса и многократная выборка по значимости хорошо работают в связке, само по себе более сложно для ОМС (т.е. вероятность попадания в существенную область пространства равномерно случайной выборкой в нём меньше), чем первичное пространство путей в Kelemen MLT [75]. Это происходит из-за априорного разбиения

(«мультиплексирования») путей по глубине и стратегиям, что, в свою очередь, создаёт две проблемы: (1) низкая эффективность равномерно-случайной генерации выборок, и как следствие, низкая вероятность принятия большого шага и (2) невозможность смены стратегии генерации выборки во время маленького шага.

Одним из возможных путей решения первой проблемы (т.е. повышение эффективности большого шага) являются работы [80-81], где используется «обмен репликами» (Replica Exchange). К этой работе мы вернёмся в подразделе 5.7.

5.4 Veach и Kelemen MLT вместе

Невозможность смены стратегии генерации выборки в MMLT во время маленького шага служит источником дополнительных артефактов на изображении, поскольку отдельные участки изображения (где MIS-веса далеки от нуля и единицы) строятся разными стратегиями и зачастую, как следствие, разными цепями Маркова. Для решения этой проблемы в работах [88-90] были разработаны гибриды MMLT с методами, работающими в мировом пространстве путей. В этих работах строится обратимый переход между мировым пространством путей и первичным пространством путей, за счёт чего удаётся добиться смены стратегии генерации выборки во время *маленького шага* при помощи Reversible-Jump MCMC, который способен работать сразу в нескольких доменах/пространствах интегрирования.

Улучшения, достигнутые таким способом, на наш взгляд являются не очень значительными. Однако при этом они существенно усложняют расчётный метод, поскольку не только требуют реализации пертурбаций в Path Space и вычисления транзитивных вероятностей перехода, но и дополнительно требуют реализации нетривиального преобразования самого пути обратно из Path Space в Primary Sample Space: для любого пути в сцене необходимо найти все возможные наборы случайных чисел, по которым такой путь мог бы быть получен (рис. 4).

5.5 Гибридный метод Монте-Карло

Гибридный метод Монте-Карло [92] – это довольно большой, постепенно расширяющийся класс методов. Эти методы генерируют выборки при помощи траектории динамической системы: гамильтоновой механики (HMC) или броуновского движения в вязкой среде на основе уравнения Ланжевена (LMC). Поскольку мы считаем эти методы перспективным направлением в рендеринге, остановимся на них более подробно.

5.5.1 Общая идея гибридного МС

В сложных сценах фазовое пространство заполнено несвязанными областями (``островками'') весьма сложной формы (пример на рис. 6). Целевая интегрируемая функция (значимость) равна нулю в большей части пространства и отлична от нуля только в пределах этих островков, а на их границах обычно испытывает разрыв. Поэтому если мы будем выбирать точки случайным образом, то почти все они попадут в область нулевой значимости и никакого вклада в интегрирование не внесут. Тем не менее, по островкам мы все же иногда попадаем, и потому эргодичность марковского процесса не нарушается.

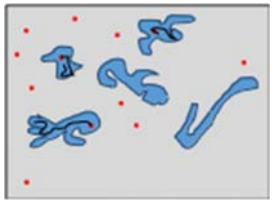

Рис. 6. Гибридный метод Монте-Карло. Прямоугольник – фазовое пространство, контуры – области, где функция значимости отлична от нуля (т.е. точка является допустимой траекторией, соединяющей камеру с источником света), отдельные точки – первичная случайная выборка, ломаные линии внутри контуров – траектории, построенные из попавших в области первичных точек

Fig. 6. Hybrid Monte Carlo method. Rectangle – phase space, contours – areas where the significance function is nonzero (i.e., a point is a valid trajectory connecting the camera to the light source), individual points – primary random sampling, broken lines inside the contours – trajectories constructed from the primary points that are in the area

Однако с ростом размерности любая тонкая область в многомерном пространстве будет «ощущаться» интегратором ещё тоньше, поскольку объём части (относительно целого) стремительно падает с ростом размерности пространства. В результате изотропное предложение перехода MCMC перестаёт работать: находясь в пределах островка, маловероятно попасть в пределы того же самого островка при помощи случайного блуждания. Чтобы это исправить, случайное блуждание точки в этот момент заменяется на *направленный* процесс обхода островка многомерного пространства при помощи упомянутой динамической системы.

После попадания в островок строим из этой точки *траекторию* $x(t)$, которая не покидает островок и исследует его с плотностью $\rho(x)$, пропорциональной интегрируемой функции (рис. 6). x – вектор координат трассы в первичном пространстве путей для трассы некоторой фиксированной длины. Траектория строится *фиксированное время* t_n (параметр метода), и ее конечные положения $x(t_n)$ включаются в выборку для интегрирования методом Монте-Карло.

Для реализации такого *направленного* обхода островка нужна динамическая система, траектория которой будет, во-первых, ограничена островком, а во-вторых, в пределах него будет двигаться так, чтобы плотность точек была максимально близка к нормированной функции значимости. Небольшое отличие вполне допустимо, его несложно скомпенсировать, например, методом Гастингса, сравнивая значимость для $x(t_{n+1})$ и $x(t_n)$ и выбирая, которую из них добавить к выборке.

5.5.2 Типы динамических систем

Придумать динамическую систему, которая будет порождать траектории с заданным распределением, весьма непросто. Поэтому, естественно взять аналогию из статистической физики с *каноническим распределением* $\rho(x) = Ze^{-E(x)/T}$, где E – «энергия состояния», T – температура в подходящих единицах (т.е. умноженная на постоянную Больцмана), Z – просто нормировка. В качестве энергии берем логарифм функции значимости $f(x)$: $E(x) = const - T\log f(x)$. Если моделировать динамику и брать результат в некоторые случайные моменты времени, то мы будем получать искомое распределение точек x, пропорционально целевой функции $f(x)$ [91].

Среди методов расчета глобального освещения существует две реализации гибридного Монте-Карло с динамическими системами, восходящими к механике: консервативная гамильтонова динамика [94] и броуновское движение в вязкой среде, т.е. один из типов уравнения Ланжевена [97].

5.5.3 Hamiltonian MC

В работе [94] в качестве динамической системы выбирается гамильтонова динамика

$$\frac{dx}{dt} = \frac{\partial H}{\partial p}, \qquad \frac{dp}{dt} = -\frac{\partial H}{\partial x},$$

где обобщенные координаты – это и есть x, обобщенный импульс p вводим, исходя из простейшего выражения кинетической энергии $K = \sum_i \frac{p_i^2}{2m_i}$. Потенциальная же энергия будет тем самым логарифмом функции значимости (формула 5.3):

$$U(\bar{x}) = -\log\big(f(\bar{x})\big). \tag{5.3}$$

Из-за ее обнуления на границе островка получаем бесконечно высокий потенциальный барьер, пересечь который наша траектория не сможет. Коль скоро полная энергия H есть сумма (формула 5.4) потенциального члена, зависящего только от обобщенных координат, и кинетического, зависящего только от обобщенного импульса, то каноническое распределение будет *произведением* $\rho_x(x)\rho_p(p)$. Поэтому мы просто игнорируем импульс и рассматриваем распределение только координат, что удобно.

$$H(\bar{x},\bar{p}) = U(\bar{x}) + K(\bar{p}) = -\log\big(f(\bar{x})\big) + \sum_i \frac{p_i^2}{2m_i}. \tag{5.4}$$

Сама по себе гамильтонова динамика часто эргодичностью не обладает. Это видно хотя бы уже из того, что полная энергия не меняется вдоль траектории, так что она ограничена подобластью, определяемой начальными условиями (начальной энергией). Поэтому надо добавить к нему взаимодействие с внешним стохастическим миром. Традиционно это делается так. В начальной точке задаются обобщенные координаты (это вектор координат трассы в первичном пространстве путей) и импульсы (которые в трассировке лучей не существуют). Импульсы определяют, как случайный вектор (например, с гауссовым распределением) с нулевым средним. С этой точки траекторию строят некоторое время. Затем импульсы в конечной точке заменяют на случайный вектор, внося в траекторию «излом», и ведут интегрирование дальше. Грубо говоря, это «случайный удар» по нашей обобщенной частице. Он-то и обеспечивает переходы между состояниями с разной полной энергией, поскольку в точке «удара» мы меняем кинетическую энергию. В каком-то смысле эта приходящая извне случайность может быть ассоциирована с *температурой*, которая будет мерой средней энергии возмущения. Температура в гамильтоновом Монте-Карло обычно выбирается единичной, а распределение случайных моментов – с единичной дисперсией.

Реализация этого метода достаточно сложна и содержит эмпирические решения. Скажем, кинетическую энергию можно ввести как произвольную положительную квадратичную форму от импульса (p, Ap):

$$K(\bar{p}) = \frac{1}{2}\,\bar{p}^T A \bar{p}.$$

Работать метод будет при любой симметричной положительно-определенной матрице A, однако эффективность будет заметно разной [93]. Поэтому матрицу эту лучше выбирать под конкретный случай.

Другой принципиально важной особенностью практической реализации гамильтонова Монте-Карло является применение метода сравнения Гастингса к следующей точке траектории. При численном решении гамильтониан на траектории сохраняется лишь приближенно, и необходимо компенсировать это явление. Наконец, весьма нетривиален вопрос о том, как часто «ломать» траекторию, заменяя импульс на случайный вектор, и с какой дисперсией. Все эти тонкости весьма серьезно влияют на эффективность метода.

5.5.4 HHMC

Основная сложность с реализацией HMC для рендеринга связана с использованием метода интегрирования второго порядка leap frog [91], выполняющего симуляцию гамильтоновой динамики. Он требует производной от правой части, которая в гамильтоновой динамике сама является производной от гамильтониана, так что нам в результате нужна матрица вторых производных от логарифма функции значимости (т.н. гессиан). Вычисления матрицы вторых производных от гамильтониана довольно дороги, а делать их надо на каждом шаге. Поэтому в работе [94] предложен вариант, где вместо реального гамильтониана берется его аппроксимация квадратичной формой (ряд Тэйлора). Такая аппроксимация позволяет в течение нескольких шагов по времени вычислять вторые производные аналитически, что заметно снижает стоимость шага.

5.5.5 DRMLT

Возможная оптимизация HHMC заключается в том, чтобы на каждом шаге сначала попробовать более простое изотропное предложение перехода, и если только оно будет отвергнуто, тогда уже использовать анизотропное предложение на основе HHMC. Так было сделано в работе [102], где был предложен гибридный алгоритм, названный Delayed Rejection MLT (DRMLT).

5.5.6 Идея Langevin MC

Работа [96] рассматривает броуновское движение в качестве динамики в пределах островка. Пусть у нас есть частица, движущаяся в потенциальном поле в вязкой среде. Вязкость приводит к тому, что частица остановится в одном из локальных минимумов потенциальной энергии. Приложим теперь к ней случайную силу, она будет выталкивать нашу частицу из потенциальной ямы и заставлять обходить целевую функцию в различных местах, не застревая в локальном минимуме. Если это случайное возмущение не слишком велико, то чем глубже локальный минимум потенциала, тем дольше частица будет из него выбираться. Равновесное распределение, определяемое «борьбой» градиента потенциала и случайной силы, будет, тем самым, иметь максимум там, где потенциал имеет минимум.

Математически такое движение выражается стохастическим дифференциальным уравнением

$$d = -\nabla U dt + \sqrt{2T} dw,$$

где $U(x)$ – потенциал, а w – винеровский процесс, т.е. такой, что приращение $dw = w(t + dt) - w$ есть случайная величина с нулевым средним и дисперсией $\sqrt{t}$ [98]. Данное уравнение – один из вариантов уравнения Ланжевена. Из-за винеровского члена в этом уравнении есть перемешивание, сходимость и эргодичность. Среднее по траектории сходится к равновесному распределению $\rho(x) = Ze^{-U(x)/T}$. Заметим, что температура тут возникает достаточно естественно как амплитуда винеровского члена (случайной силы). Обычно ее берут единичной. Таким образом, как и в случае гамильтоновой механики, в качестве x берем вектор переменных узлов полной трассы луча (без источника и камеры), $T = 1$, а потенциал равен логарифму функции значимости $f(x)$: $U(x) = -\log f(x)$. На границе островка имеем опять бесконечный потенциальный барьер, который наша траектория преодолеть не может. А внутренность его она заполняет как раз с нужной нам плотностью.

В отличие от HMC, в Langevin MC нет математических проблем с перемешиванием, эргодичностью и сходимостью распределений. Температура здесь не является чужеродным фактором. С математической точки зрения этот вариант выглядит привлекательнее. Он должен работать просто при реализации в соответствии с математической идеей. Тем не менее, и для Ланжевена подчас используют метод сравнения Гастингса, а также необходимо как-то выбрать длину траектории и пр.

5.5.7 Langevin MC в практике

Работа [96] не только использует более простое представление для производных (градиент), но и кэширует их вычисление, благодаря чему удаётся существенно снизить накладные расходы на их вычисление. Основная идея кэширования заключается в том, чтобы для алгоритма MALA не вычислять градиенты целиком для каждого шага, а обновлять их при помощи ограниченной порции вычислений по мере движения цепи (эта идея в работе [96] называется «online adaptation»). При этом необходимо отметить, что алгоритм MALA, используемый в [96], является частным случаем полноценного HMC, когда leap frog [91] делает один шаг, и этот шаг сразу берётся в качестве следующего предложения перехода (формула 5.5):

$$\bar{u}^{t+1} = \bar{u}^t + \epsilon \nabla L(\bar{u}^t) + \sigma\sqrt{\epsilon}\mathcal{N}(0,1), \quad (5.5)$$

где $\bar{u}$ – вектор состояния цепи, $L(u) = -\log(F(\bar{u}))$, ϵ и σ – параметры алгоритма, а $\mathcal{N}(0,1)$ – единичный многомерный гауссиан.

К сожалению, наивное применение MALA к задаче рендеринга на даёт существенного улучшения, что демонстрирует рис. 3 работы [96]. Как там отмечается, MALA всё же предполагает, что случайные величины в векторе L имеют приблизительно одинаковую дисперсию. Если это не так, сходимость ухудшается. Существующие решения этой проблемы через так называемую матрицу предварительной обработки (preconditioning matrix) [104] оказываются слишком дорогими для рендеринга в силу дороговизны вычисления целевой функции. Работа [96] фокусируется на оптимизации вычисления этой матрицы с использованием *адаптивной предварительной обработки* (adaptive preconditioning) и реализации таким образом алгоритма *MALA с адаптацией* (MALA with adaptation). Она предлагает 3 варианта оптимизации:

(a) MALA с подкреплением (online adaptation);
(b) MALA на основе кэширования (cache-driven adaptation);
(c) гибрид двух предыдущих подходов (hybrid adaptation).

5.5.8 Заключение по гибридному MC

К сожалению, все три упомянутые работы требуют применения методов автоматического дифференцирования [99], что оставляет вопрос практической применимости открытым, т.к. делает добавление новых математических моделей материалов и источников освещения в расчётную систему крайне сложным и трудоёмким процессом. Например, в системе Mitsuba2 [101] предлагается расширяемая система со встроенными возможностями дифференцирования. Тем не менее, Mitsuba2 не реализует HMC, а использует дифференцирование для решения задачи так называемого обратного рендеринга, когда по заданному изображению требуется восстановить некоторые характеристики сцены или отдельных моделей.

Важный нюанс, на который должны обратить внимание разработчики расчётного ПО в работах [94] и [96], – это дифференцирование при помощи шаблонного мета-программирования C++ [99]. Такой код крайне трудно развивать и отлаживать, а его эффективность зачастую невозможно оценить. Кроме того, это налагает определённые ограничения на используемый компилятор и затрудняет перенос на GPU. Хотя применение инструментов на основе source-to-source преобразований исходного кода при помощи специального компилятора выглядит перспективным направлением с практической точки зрения [100] и позволяет надеяться на внедрение HMC в расчётные системы в будущем.

Кроме того, в работах [94] и [96] отмечается, что при наличии высокочастотных карт нормалей или карт смещений (имитация микрорельефа поверхности) дифференцирование не будет корректно работать. При этом очевидно, что поддержка Spatial Varying BSDF (SVBSDF [109]) также находится под вопросом, поскольку для них невозможно получить производную

аналитически. На наш взгляд, для современных приложений это даже более критичная функциональность, чем карты смещений.

Мы полагаем, что в настоящее время высокоэффективная реализация этих методов на GPU выглядит вряд ли возможной в силу того, что HHMC [94] имеет высокие затраты памяти на 1 поток из-за необходимости вычисления гессиана. Аналогичные проблемы наблюдаются и в работе [96], где на каждый поток нужно хранить *preconditioning matrix*.

Тем не менее, методы расчёта освещения на основе гибридного Монте-Карло являются перспективным направлением исследований. Эти методы достаточно универсальны, их математическое обоснование хорошо проработано, они обладают лучшей сходимостью при росте размерности пространства интегрирования: $d^{5/4}$ для HMC против d^2 для MCMC, где d –размерность ([91], стр. 29, конец раздела «Scaling of HMC and random-walk Metropolis»). Численное сравнение можно найти в работе [95], раздел 7.2. Это происходит благодаря анизотропному предложению перехода, который строится на основе производных. HMC существенно выигрывает у MCMC потому, что с ростом размерности любая «тонкая» область пространства становится ещё тоньше, из-за чего у изотропного предложения перехода в MCMC очень быстро снижается шанс остаться в области функции с высокой значимостью.

5.6 Population Monte Carlo

Обособленно от других находится класс методов, основанный на отборе популяции выборок (поэтому они называются Population Monte Carlo, или PMC) [105-106]. Основная идея этих методов состоит в том, чтобы запоминать информацию о выборках во времени, и затем переиспользовать наиболее удачные выборки как стартовое состояние для небольших шагов (мутаций). Схожую идею использует упомянутая нами ранее работа [80], которая переиспользует части состояний марковской цепи для повышения вероятности принятия большого шага в стандартном MCMC с длинными цепями. Можно сказать, что эти три метода близки по способу исследования пространства к генетическим алгоритмам (что на наш взгляд в целом нельзя говорить по отношению к другим MCMC методам).

Огромное преимущество PMC по сравнению с традиционным MCMC заключается в большей параллельности метода за счёт коротких цепей, что чрезвычайно важно для реализации MCMC на графических процессорах. Например, известно, что коммерческая система расчёта освещения на GPU Octane использует PMC [107]. Кроме того, рассматриваемый класс методов обладает большей интерактивностью (т.е. в начале расчёта изображение выглядит более правильно) по сравнению с традиционными MCMC алгоритмами с длинными цепями. Также PMC может вычислять прямой свет, что категорически не рекомендуется делать во всех видах MLT. Среди недостатков, как правило, меньшая точность в пределе по сравнению с MLT [87].

5.7 Replica Exchange

Метод обмен репликами (RELT) [79-81] можно считать расширением обычного MLT в первичном пространстве путей. RELT подразумевает, что несколько марковских цепей (число K в алгоритме 1, работа [80]) работают параллельно. Отличие от обычного MLT заключается в том, что после каждого шага цепи могут обменяться состояниями с некоторой вероятностью. Для того, чтобы такой обмен не нарушал стационарность распределения, вероятность должна вычисляться по формуле 5.6 [1]:

$$p(i,j) = \frac{F_i(u_j) \times F_j(u_i)}{F_i(u_i) \times F_j(u_j)}. \quad (5.6)$$

Здесь u_i и u_j – это состояния, между которыми предполагается произвести обмен, а F – целевая функция, пропорционально которой необходимо построить распределение выборок. Необычным здесь является то, что функций F на самом деле несколько. В методе обмена

репликами используется так называемая процедура закалки функций (function tempering [82]). Суть этой процедуры в том, чтобы применить к целевой функции преобразование следующего вида (формула 5.7):

$$F(x) = F(x)^{1/T_I}. \quad (5.7)$$

Здесь $T_I \geq 1$ – параметр, называемый температурой. Чем больше температура, тем более плоской является преобразованная функция. Цепи упорядочиваются в соответствии с их температурой, чтобы обмен производился только между соседними цепями, что повышает вероятность принятия обмена.

Преимущества RELT (как и PMC) – лучшее исследование пространства интегрирования в глобальном смысле и лучшая параллелизуемость. Основным недостатком RELT в [1] считают тот факт, что сходимость сильно зависит от количества цепей, а это количество трудно подобрать правильно. С другой стороны, при реализации на GPU большое количество цепей не является проблемой. RELT используется во многих работах [40-41, 83] как метод, улучшающий глобальное исследование пространства Марковскими цепями.

Стоит отметить, что применение идеи обмена реплик на практике имеет нюансы. Ключевой вопрос заключается в выборе стратегий генерации выборок для каждой цепи (реплики). В [80] было использовано 4 типа реплик: (1) равномерная стратегия генерация выборок (большой шаг), (2) реплика «полного пути», (3) реплика «непрямого освещения» и (4) специальная реплика для дельта-функций. При этом вторая и третья реплики на самом деле являются комбинациями различных типов путей (получаемых разными стратегиями генерации выборок), а четвёртая реплика используется исключительно для SDS каустиков при помощи неявной стратегии. Таким образом, практическое применение RELT тесно переплетается с использованием различных стратегий в многократной выборке по значимости в BPT или PT, и при этом вклад от различных реплик между собой также учитывался на основе многократной выборки по значимости.

Причина повышения эффективности RELT (по сравнению, например, с Kelemen MLT) заключается в том, что каждая реплика работает со специфическим типом путей. Благодаря этому вероятность сделать удачное предложение перехода внутри реплики повышается (можно, например, сделать шаг меньше для четвёртого типа реплик), а фактическая размерность пространства интегрирования понижается.

5.8 Стратификация (Tiled MLT)

Впервые идея стратификации была применена к MCMC в рендеринге в работе [83]. Изображение разбивалось на перекрывающиеся участки, каждый из которых обрабатывался отдельной марковской цепью независимо. Для каждого из участков оценивалась своя константа нормализации, после чего можно было собрать изображение из участков, зная константу нормализации для всего изображения. Благодаря стратификации в [83] достигался более равномерный обход изображения марковскими цепями, за счёт чего изображение сходится к эталону более предсказуемо.

5.9 Markov Chain PPM

Существует подход, объединяющий в себе фотонные карты и марковские цепи. Так в [39-40] (Markov Chain PPM, MCPPM) марковские цепи были применены к стохастическим прогрессивным фотонным картам (SPPM [19]), а в [41] (MBE) к алгоритму VCM [24]. MCPPM позволяет решить проблему плохого распределения фотонов на сценах с большой площадью. Однако MCPPM, как и оригинальный SPPM, плохо справляется с многократными глянцевыми отражениями.

Объединение фотонных карт и марковских цепей может улучшить расчётные системы, которые основаны на фотонных картах. Однако нам не кажется это направление

перспективным по сравнению с полностью несмещёнными методами. Причина нашего скепсиса заключается в том, что и Метрополис (марковские цепи), и фотонные карты накапливают значения интеграла единичными выборками и потому имеют схожие недостатки, которые будут усиливаться в комбинации: в тёмных областях изображение получается шумным (именно для устранения этого недостатка в MBE [41] был применён метод RELT). Отметим, что низкочастотный шум от фотонных карт крайне тяжело удалять [48].

Другой недостаток, который может усиливаться в комбинированном методе, – это образование сгустков фотонов, которые тяжелы для обработки в вычислительном плане. Как мы отмечали ранее, контроль плотности [22] возможен не для всех типов BRDF, что существенно ограничивает применимость Markov Chain PPM в компьютерной графике.

Что касается MBE [41], этот метод прежде всего является комбинацией других методов: VCM, RELT и предыдущих MCPPM подходов. Для лучей из камеры он использует обыкновенную трассировку путей, а для путей из источника использует MCPPM. По опыту реализации гибридных подходов реальная трудоёмкость его реализации и верификации должна многократно превышать сумму соответствующих величин отдельных методов. Поэтому, прежде чем приступить к его реализации, на наш взгляд придётся в том или ином виде сначала реализовать базовые методы. После этого можно будет приступить к более детальному изучению работы [41], чтобы построить гибридный подход. Интересно также отметить, что Multiplexed MBE метода, комбинирующего идеи MMLT с VCM и RELT, в настоящий момент ещё не существует.

5.10 Реализации MLT на GPU

Существенное отличие GPU реализации от CPU реализации для MCMC методов состоит в том, что длина Марковских цепей на GPU будет на несколько порядков меньше. Хотя при этом общее количество используемых цепей намного больше. Несущественное на первый взгляд, это отличие на самом деле является критическим для MCMC, о чём говорится во всех рассматриваемых далее работах.

В [85] на GPU впервые был реализован метод Kelemen MLT [75] на основе традиционной двунаправленной трассировки путей (BPT). В качестве проблем в [85] отмечены высокие траты памяти (обусловленные двунаправленной трассировкой путей и необходимостью хранить весь вектор случайных чисел на каждый поток) и высокое значение начального смещения (start-up bias). В качестве основного направления улучшения исследуются метод регенерации путей (при помощи уплотнения оставшихся активных потоков на GPU после некоторого количества переотражений) и параллельное вычисление соединений в BPT, что в сумме повышает скорость на 20-30% по сравнению с наивной реализацией.

В [86] для Kelemen MLT было предложено решение проблемы большого начального смещения на GPU при помощи отбора начальных состояний марковских цепей обыкновенным Монте-Карло (так называемого параллельного прожига). Также в [86] рассматриваются методы уменьшения занимаемой памяти, а для повышения когерентности лучей используется сортировка потоков. Это даёт примерно такой же прирост в 20-30% производительности, как и в [85].

В [84] впервые сообщается об успешной реализации MMLT на графических процессорах. Предлагаемые методы – Speculative MLT (SMLT) и Rejection Chain MLT (RCMLT) – позволяют вычислять несколько предложений переходов в MCMC параллельно, однако ухудшают ошибку в целом. Идея метода RCMLT состоит в том, чтобы вычислить сразу несколько предложений перехода с запасом. Тогда если первое предложение будет отвергнуто, то можно сразу попробовать следующее. SMLT является развитием RCMLT. Сначала параллельно вычисляется дерево всех возможных переходов, после чего производится фактический переход в один из листов.

Однако следует отметить, что MMLT – это последний из рассматриваемых нами MCMC алгоритмов, для которых известны реализации на GPU. То есть для более новых методов реализации на GPU ещё не существует, что на наш взгляд является результатом увеличения сложности подходов и возрастающих ограничений, к чему мы ещё вернёмся. При этом в [84, 87] отмечается, что из-за мультиплексированного пространства интегрирования проблема начального смещения для MMLT более существенна, чем, например, для более простого Kelemen MLT. Если для Kelemen MLT в существующих реализациях [85-86] проблема начального смещения может быть амортизирована при помощи параллельного прожига [86], то для MMLT этого метода оказывается недостаточно [87].

5.11 Дальнейшее изучение MCMC

Мы рекомендуем также ознакомиться с работой [1] для углублённого анализа методов расчёта освещения на основе MCMC. Эта работа подробно рассматривает принципы работы методов на основе Монте-Карло по схеме марковских цепей, и может служить в качестве отправной точки для понимания их на более глубоком уровне с целью практической реализации. Интересным результатом этой работы на наш взгляд является оценка эффективности MCMC методов по 3 типам и постулирование открытых проблем в области.

(a) Local exploration. Эффективность локального исследования пространства. Это тип эффективности можно назвать основным, т.к. именно он определяет точность расчёта в пределе.

(b) Global Exploration. Эффективность глобального исследования пространства. Этот параметр влияет на то, насколько быстро из изображения уходит начальное смещение, что важно, например, для реализации методов на GPU. В [1] отмечается, что на сегодняшний день эффективное глобальное исследование пространства является первой нерешённой проблемой.

(c) Uniform image error. Равномерность ошибки на изображении. Этот параметр является ключевым для анимации и отчасти зависит от предыдущего. В [1] говорится что почти все MCMC алгоритмы страдают от временной нестабильности, и эта вторая нерешенная в настоящее время проблема. Кроме того, для MCMC методов отмечается отсутствие надёжной метрики оценки ошибки на изображении, которая могла бы быть использована для остановки счёта. Это – третья проблема.

Мы не считаем перечисленные проблемы критичными по следующим причинам.

(a) Современные методы обработки изображений и шумоподавления (в том числе на основе нейросетей) по нашему опыту в состоянии амортизировать многие из артефактов, вызванных, например, второй проблемой.

(b) Для многих применений на практике временн*ая* стабильность алгоритма не критична. Более важными параметрами могут быть надёжность (robusntess) и эффективность расчёта.

(c) При реализации MCMC методов на GPU количество марковских цепей большое, но сами цепи при этом успевают сделать меньше шагов [86-87]. Это приводит к совершенно иному влиянию обозначенных проблем на изображение, чем при расчёте на CPU. Проблема временной стабильности перестаёт быть критичной, т.к. при таком большом количестве цепей мельтешение из анимации уходит. Однако проблема начального смещения начинает проявляться в виде недооценённой яркости некоторых участков изображения, в которых марковские цепи не успели накопить в гистограмме высокие значения яркости [87]. Хотя это также является серьезной проблемой, в некоторых приложениях (например, визуализация архитектурных проектов) на наш взгляд такой тип артефактов является менее критичным.

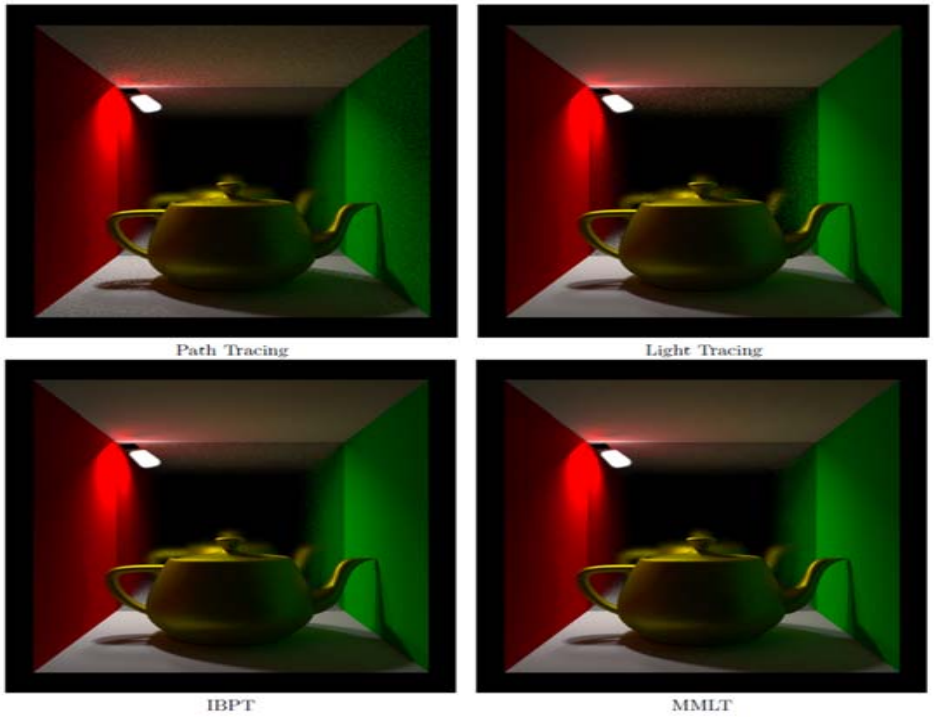

Рис. 7. Сравнение различных алгоритмов интегрирования освещённости. Время рендеринга – 10 минут на процессоре Intel Core i7 3770, 3.4 GHz. В данной сцене источник в виде площадки повёрнут в направлении диффузной стены слева и задней глянцевой стены, благодаря чему значительная часть света – вторичное освещение, а первичное представлено большим бликом на стене слева. Благодаря этому Path Tracing производит значительный шум на всём изображении. Light Tracing неплохо справляется со вторичным освещением, но по сравнению с обратной трассировкой существенно ухудшает точность на глянцевой стене. IBPT работает значительно лучше двух предыдущих алгоритмов, но шум на глянцевой стене всё ещё остаётся. Кроме того, нетрудно заметить, что этот шум больше чем в PT. Это демонстрирует классический недостаток MIS: чем больше стратегий, тем хуже могут вычисляться отдельные элементы сцены, т.к. за то же самое время они получают меньше удачных выборок. MMLT надёжно и эффективно посчитал все видимые эффекты на данной сцене

Fig. 7. Comparison of various algorithms for integrating illumination. Rendering time - 10 minutes on an Intel Core i7 3770 processor, 3.4 GHz. In this scene, the source in the form of a platform is turned towards the diffuse wall on the left and the back glossy wall, due to which a significant part of the light is secondary lighting, and the primary is represented by a large glare on the wall to the left. As a result, Path Tracing produces a significant amount of noise throughout the image. Light Tracing does a good job with secondary lighting, but compared to back tracing, it significantly degrades accuracy on a glossy wall. IBPT performs significantly better than the previous two algorithms, but there is still noise on the glossy wall. In addition, it is easy to see that this noise is greater than in PT. This demonstrates the classic disadvantage of MIS: the more strategies, the worse the individual scene elements can be calculated, since in the same time, they get fewer hits. MMLT reliably and efficiently counted all visible effects on a given scene

6. Эксперименты

В процессе работы с различными методами расчёта освещенности мы проводили экспериментальные сравнения различных методов, используя наши собственные реализации.

Здесь мы не ставим задачу объективного сравнения, поскольку считаем, что такое сравнение невозможно без создания соответствующего набора сцен, который бы объективно отражал некоторую усреднённую реальность. А это является работой, выходящей за рамки данной статьи. Кроме того, некоторые методы были реализованы нами на CPU, а некоторые на GPU. Однако, мы можем использовать эксперименты для того, чтобы зафиксировать некоторые выводы и обозначить занимаемую нами позицию по отношению к тем или иным методам.

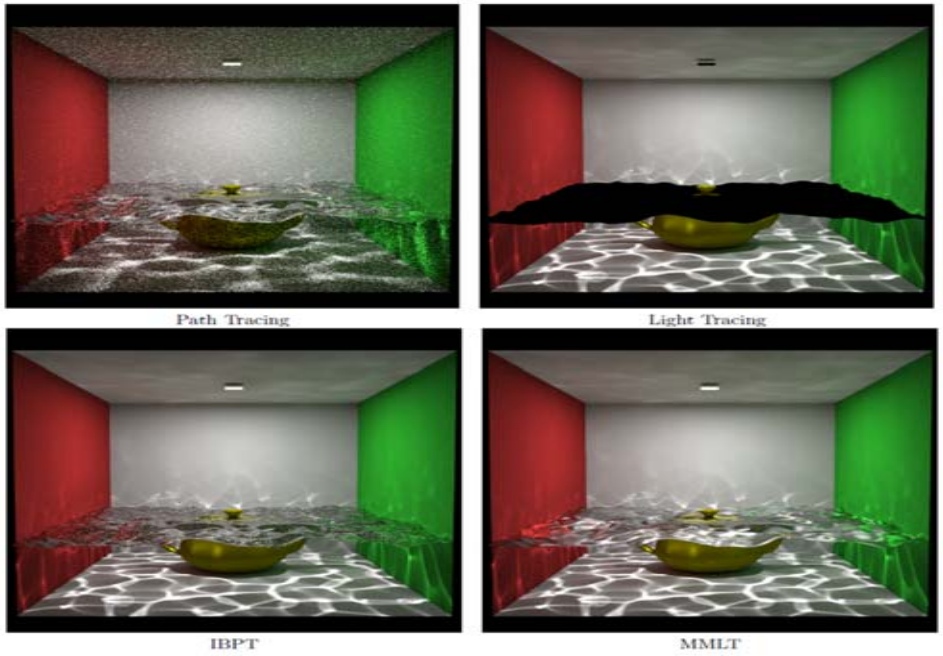

Рис. 8. Сравнение различных алгоритмов интегрирования освещённости. Время рендеринга – 10 минут на процессоре Intel Core i7 3770, 3.4 GHz. Классический пример трудновычислимых каустик. В данной сцене представлены два типа каустик: видимые камерой напрямую (колонка Ind.) на дне и стенах коробки, и каустики, видимые через зеркальное переотражение в воде (колонка SDS, само изображение поверхности воды). На данных изображениях видно, что хотя эффективность расчёта SDS каустик на поверхности воды у MMLT всё же ниже, чем эффективность расчёта более простой каустики на дне и стенах коробки, метод всё ещё можно считать надёжным

Fig. 8. Comparison of various algorithms for integrating illumination. Rendering time – 10 minutes on an Intel Core i7 3770 processor, 3.4 GHz. A classic example of hard-to-calculate caustics. There are two types of caustics in this scene: directly visible by the camera (Ind. Column) on the bottom and walls of the box, and caustics visible through specular reflection in water (SDS column, the actual image of the water surface). These images show that although the efficiency of calculating SDS caustics on the water surface of MMLT is still lower than the efficiency of calculating simpler caustics on the bottom and walls of the box, the method can still be considered reliable.

Эксперимент №1: Хотя двунаправленные методы более надёжны чем однонаправленные (рис. 7-10), нельзя сказать, что они более быстрые. Это хорошо видно из рис. 9, где прямая Монте-Карло трассировка (Light Tracing) лучше посчитала каустику от тора на полу, чем двунаправленная (IBPT). Недостаток MIS проявляется здесь в полной мере: увеличивая надёжность метода, мы понижаем скорость расчёта отдельных явлений. Метод Multiplexed MLT (MMLT) лишён данного недостатка, поскольку стохастически оптимально выбирает стратегии генерации выборок и показывает лучшие результаты на всех 4 сценах (рис. 7-10).

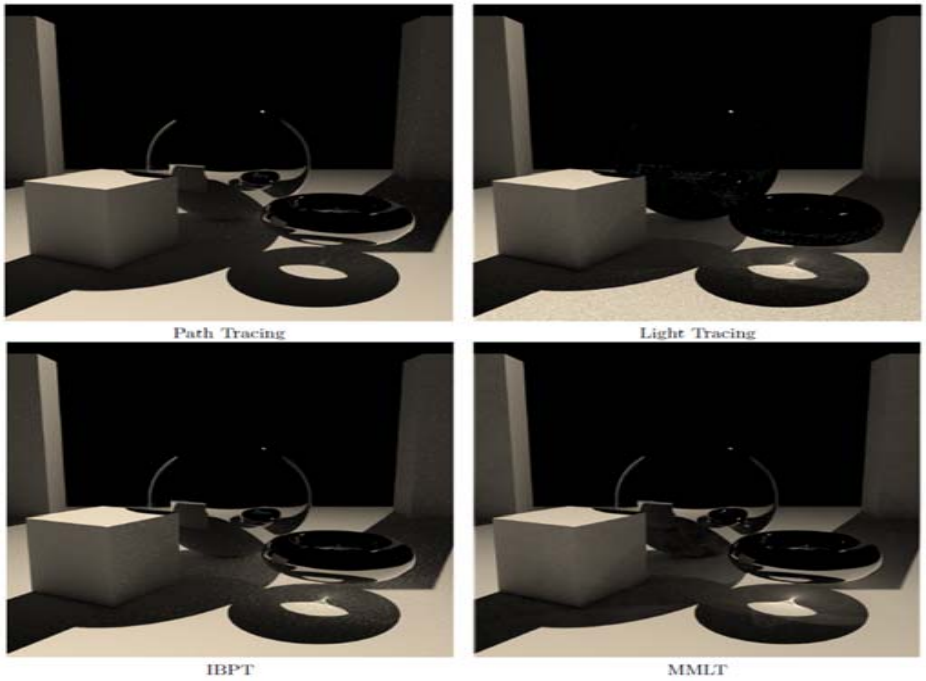

Рис. 9. Сравнение различных алгоритмов интегрирования освещённости. Время рендеринга – 5 минут на процессоре Intel Core i7 3770, 3.4 GHz. На данном изображении демонстрируется важность марковских цепей, поскольку видимая напрямую каустика внизу под тором в MMLT получилась точнее, чем в Light Tracing. Отметим, что на рассматриваемых ранее сценах это не наблюдалось, т.к. ранее объём сцены был крайне ограничен, и Light Tracing, как и фотонные карты, на подобных сценах обладают высокой эффективностью. На данной же сцене эффективность прямого Монте-Карло снижена из-за необходимости производить генерацию случайных лучей по всей площади сцены для источника, имитирующего солнце

Fig. 9. Comparison of various algorithms for integrating illumination. Rendering time – 5 minutes on an Intel Core i7 3770 processor, 3.4 GHz. This image demonstrates the importance of Markov chains, since the caustics seen directly below the torus in MMLT is more accurate than in Light Tracing. Note that this was not observed in the previously considered scenes, since Previously, the volume of the scene was extremely limited, and Light Tracing, like photon cards, is highly efficient in such scenes. On the same scene, the efficiency of the direct Monte Carlo is reduced due to the need to generate random rays over the entire area of the scene for a source simulating the sun.

Эксперимент №2: Фотонные карты хорошо показывают себя при небольшом времени расчёта и в ограниченном наборе сценариев, где преобладают ярко выраженные диффузные и зеркальные поверхности. Но в чистом виде (т.е. без применения ряда усовершенствований) они проигрывают методам на основе Марковских цепей (Kelemen MLT и MMLT) при более длительном расчёте и при увеличении сложности освещения (рис. 11). К счастью, фотонные карты и Марковские цепи можно комбинировать [39-41].

Эксперимент №3: Применение полностью несмещённых методов предпочтительнее использованию смещённых по целому ряду причин: большая точность в пределе, универсальность по отношению к BSDF, отсутствие необходимости в построении ускоряющих структур (что облегчает реализацию на GPU).

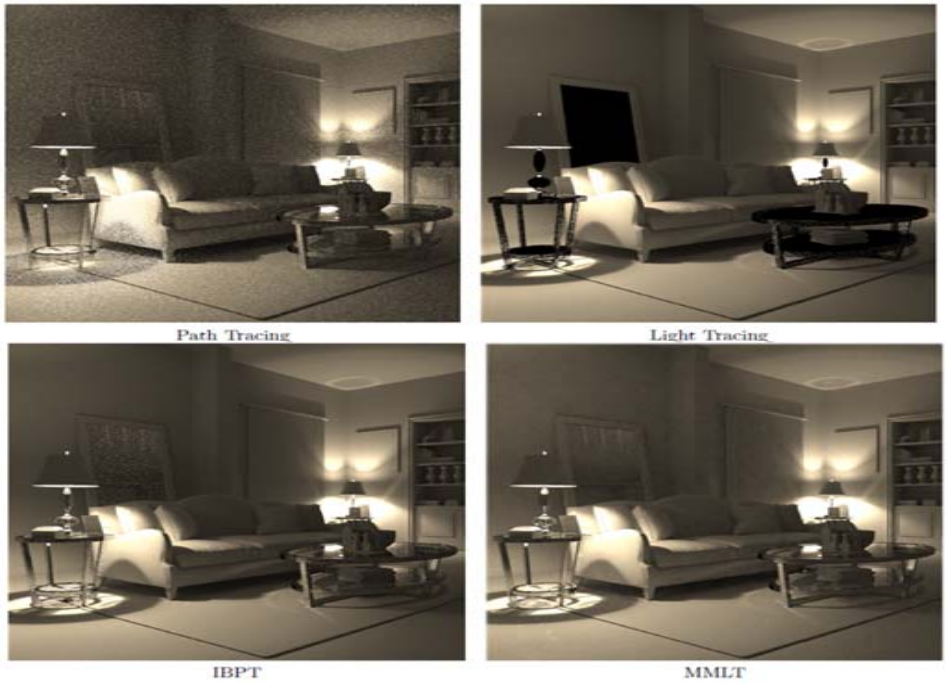

Рис. 10. Сравнение различных алгоритмов интегрирования освещённости. Время рендеринга – 60 минут на процессоре Intel Core i7 3770, 3.4 GHz. Более сложная сцена, содержащая небольшие геометрические детали и три типа материалов: ламбертовские, идеально-зеркальные и материал с глянцевым отражением (ножки столов). Алгоритм MMLT демонстрирует надёжность (robustness) в отличие от других методов. Если сопоставить данный рисунок с рис. 7-9, можно заметить, что при росте сложности сцены преимущество MMLT становится более заметным

Fig. 10. Comparison of various algorithms for integrating illumination. Rendering time – 60 minutes on an Intel Core i7 3770 processor, 3.4 GHz. A more complex scene containing small geometrical details and three types of materials: Lambertian, perfect mirror, and glossy reflective material (table legs). The MMLT algorithm demonstrates robustness in contrast to other methods. If we compare this figure with Fig. 7-9, you can see that as the complexity of the scene increases, the advantage of MMLT becomes more noticeable.

Мы использовали фотонные карты на раннем этапе нашей работы, однако впоследствии заменили их на прямую Монте-Карло трассировку лучей (Light Tracing) в целях изучения вкладов от различных статегий в BPT. Метод PCLT [30] похожим образом заменяет сбор для первично видимых поверхностей на проекцию точек на плоскость изображения, но при этом выполняет сбор для остальных случаев. Если бы мы добавили фотонные карты к сравнениям на рис. 7-10, то увидели бы улучшенные версии изображений для Light Tracing (если добавлять их без MIS) или IBPT (если добавлять их с MIS, как это делается в VCM) на зеркальных поверхностях. Однако изображения поверхностей с многократными глянцевыми отражениями не получат улучшения от фотонных карт. Кроме того, во всех случаях, где видно преимущество MMLT над Light Tracing на диффузных поверхностях, фотонные карты также не принесут каких-либо улучшений. Скорее наоборот, результат ухудшиться, т.к. высокочастотный шум от метода Light Tracing (который легко фильтровать при необходимости) перейдёт в низкочастотный трудноустранимый шум в виде цветных пятен [48].

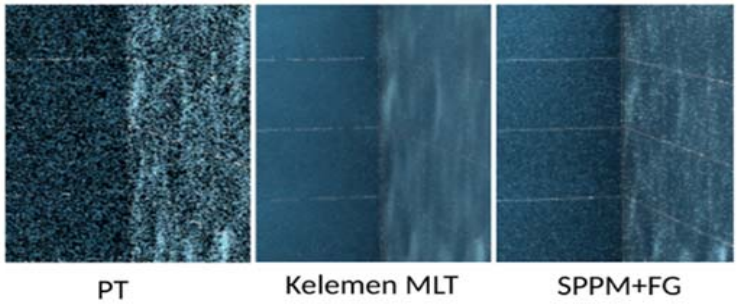

Рис. 11. Сравнение фотонных карт (SPPM+FG) против Kelemen MLT на сцене бассеина с каустиками. Время рендеринга – 10 минут на GPU AMD RX580. Низкая эффективность фотонных карт на этой сцене обусловлена тем, что камера видит не более 10\% площади всей сцены (угол бассеина). Из-за этого значительная часть фотонов не долетает до видимой области сцены, и вычисления по их трассировке производятся впустую. Прямоугольник на рис. 12 отмечает участок изображения, который был увеличен

Fig. 11. Comparison of photon maps (SPPM + FG) against Kelemen MLT on the stage of a pool with caustics. Rendering time is 10 minutes on AMD RX580 GPU. The low efficiency of photon maps in this scene is due to the fact that the camera sees no more than 10% of the area of the entire scene (the corner of the pool). Because of this, a significant part of the photons do not reach the visible area of the scene, and calculations for their tracing are wasted. The rectangle in fig. 12 marks the area of the image that has been enlarged

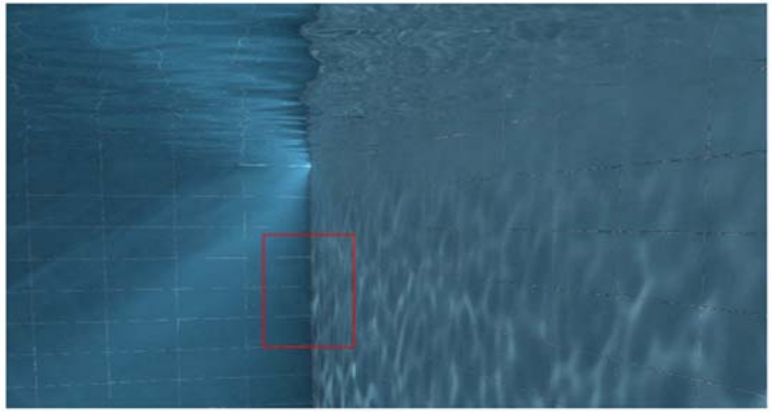

Рис. 12. Сцена с бассейном (использованная в сравнении на рис. 11)
Fig. 12. Pool scene (used in comparison in fig. 11)

Эксперимент №4: Методы на основе Марковских цепей в целом существенно лучше себя показывают на сложном вторичном освещении, чем любые методы на основе обычного Монте-Карло (рис. 7-11). В условиях сложного вторичного освещения их однозначно следует выбирать при реализации алгоритмов на CPU. Однако при реализации алгоритмов на GPU можно столкнуться с эффектом неожиданно возросшего начального смещения, что особенно сильно проявляется в MMLT (рис. 13). Поэтому мы считаем, что для GPU больше подходит Kelemen MLT [75] и PMC [105-106].

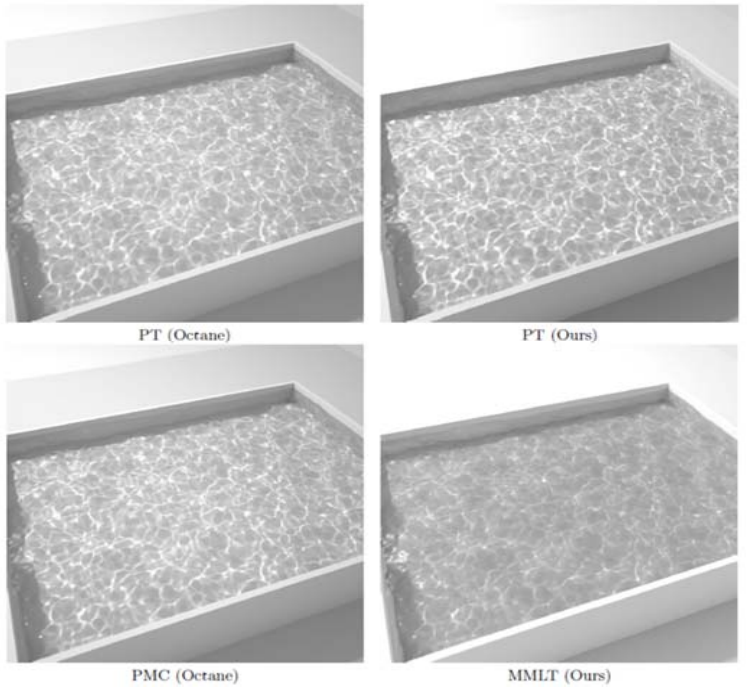

Рис. 13. Демонстрация проблемы большого начального смещения в MMLT при реализации на GPU, которое проявляется в виде недооценённой яркости каустиков. Время рендеринга – 30 минут на GTX1070. Следует отметить, что подобная сцена могла бы быть эффективно посчитана при помощи фотонных карт

Fig. 13. Demonstration of the problem of a large initial displacement in MMLT when implemented on a GPU, which manifests itself in the form of underestimated brightness of caustics. Rendering time is 30 minutes on GTX1070. It should be noted that such a scene could be efficiently computed using photon maps

Эксперимент №5: Сильно недооценённым является метод PMC, который хотя и проигрывал в среднем нашей реализации MMLT [108] (рис. 14), но не катастрофически. Кроме того, надо принять во внимание, что реализация PMC в Octane однонаправленная, в то время как MMLT – двунаправленный алгоритм, что отчасти и обуславливает его преимущества на рис 14. С учётом того, что PMC использует короткие цепи, его параллельная реализация на GPU представляется нам разумным решением, которое и сделали разработчики Octane.

7. Выбор метода

На сегодняшний день нельзя выделить универсальный или лучший метод. Если это допускает постановка задачи, следует всегда принимать во внимание специфику типичных сцен, для которых создается рендерер. Так может оказаться излишним применение развитых и более сложных методов расчёта освещённости. Кроме того, специфика сцен может очень сильно изменить баланс эффективности в сторону того или иного метода. Например, для расчёта освещения от панорамы на открытом пространстве в большинстве случаев подойдёт применение обычного MIS PT, т.к. почти всё освещение в таком случае будет первичным. Если в High Dynamic Range (HDR) панораме, например, встречаются отдельные сверхяркие области, то могут помочь простые MCMC методы, вроде Kelemen MLT или PMC поверх MIS PT. Применять двунаправленные методы в этом случае бессмысленно по причине низкой эффективности прямой стратегии. С другой стороны, для закрытых помещений и с большим

числом SDS каустик хорошо подходят методы на основе фотонных карт. Если же постановка задачи не допускает априорных предположений о сцене и требуется построение универсального и надёжного решения, тогда следует использовать передовые методы на основе MCMC. Исходя из нашей практики, мы бы рекомендовали метод MMLT.

Рис. 14. Сравнение нашей реализации PT и MMLT с Octane на GPU RTX2070 (без аппаратной поддержки трассировки лучей). Время рендеринга – 30 минут. В данной сцене вторичное освещение получено путём отражения от глянцевой поверхности, близкой к идеальному зеркалу (как стол посередине комнаты). На полу присутствует микрорельеф. Сравнение производилось между методом PMC, реализованным в системе Octane, и MMLT, реализованным в системе Hydra Renderer. На рис. 15 показаны увеличенные части изображений, демонстрирующие преимущество метода MMLT. Прямоугольники с номерами на левом верхнем рисунке обозначают участки изображений, которые представлены на рис. 15

Fig. 14. Comparison of our implementation of PT and MMLT with Octane on GPU RTX2070 (no hardware support for ray tracing). Rendering time is 30 minutes. In this scene, the secondary lighting is obtained by reflecting off a glossy surface close to an ideal mirror (like a table in the middle of a room). There is a micro-relief on the floor. The comparison was made between the PMC method implemented in the Octane system and the MMLT method implemented in the Hydra Renderer system. In fig. 15 shows enlarged portions of images showing the advantage of the MMLT method. The numbered rectangles in the upper left figure denote the image areas shown in Fig. 15

Мы полагаем, что для сложного освещения наиболее эффективными могут быть признаны методы на основе современных работ: VCM и аналоги [24, 25, 27], HMC [94, 96, 102], MMLT [78, 84], RJMLT [88-90] и MBE [41]. Например, RJMLT, задуманный его авторами как улучшение MMLT, является более эффективным классом методов, чем MMLT. Однако ключевой вопрос в цене, которую приходится платить за это улучшение. В этом разделе мы хотели бы дать рекомендации для читателя с учётом целевой задачи, нашего опыта, сложности реализации метода (в первую очередь упоминая простые методы и лишь затем более сложные) и объективных ограничений, которые метод имеет. Мы будем считать, что

при переходе к следующему подразделу все феномены освещённости предыдущих подразделов также должны вычисляться методом эффективно, если обратное не оговорено. То есть сложность сцен и освещения растёт кумулятивно.

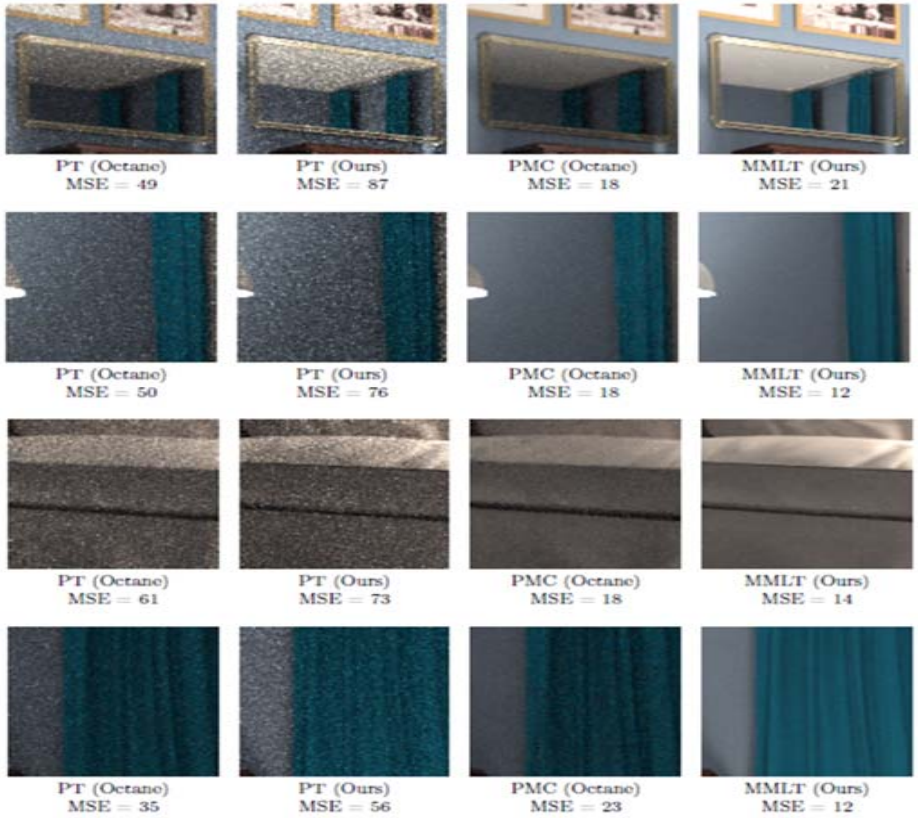

Рис. 15. Сравнение увеличенных фрагментов из рис. 14. Вырезки расположены сверху вниз в порядке их номеров от 1 до 4

Fig. 15. Comparison of enlarged fragments from fig. 14. The clippings are arranged from top to bottom in numerical order from 1 to 4

7.1 Прямое освещение

Для эффективного расчёта прямого освещения необходима реализация как минимум MIS PT с явной (для «маленьких» источников) и неявной (для «больших» источников) стратегией генерации выборок. При большом количестве источников рекомендуется дополнить MIS PT до IBPT [6], добавив ещё одну стратегию генерации выборок – Light Tracing. Ранее мы отмечали, что эта стратегия позволяет эффективно рассчитывать освещение при большом числе источников. К сожалению, у Light Tracing есть определённые сложности с моделированием объектива камеры (тёмные края объектов), что следует иметь в виду при выборе этого метода. Логичным решением в данном случае является замена на фотонные карты (что ухудшит сходимость и восприятие изображения), либо использование методов из работы [9] для эффективного выбора источника в MIS PT.

Необходимо упомянуть, что методы на основе MCMC не могут эффективно рассчитывать первичное освещение, которое, как правило, вычисляется отдельно.

7.2 Каустики первого рода

Каустиками первого рода мы называем каустики, образующиеся на диффузной поверхности и видимые непосредственно камерой. Методы IBPT [6], BPT [3], PCBPT [7] хорошо справляются с подобными каустиками, как и с любым другим типом вторичного освещения, непосредственно видимого камерой (т.е. чтобы проекция на плоскость камеры могла быть использована) и появляющегося на преимущественно диффузных поверхностях. Если применение двунаправленных методов невозможно по каким-либо причинам, нарушающим симметрию, рекомендуется использовать простые методы на основе MCMC поверх MIS PT: Kelemen MLT [75], RELT [80], PMC [105-106]. Их реализация (особенно Kelemen MLT) является достаточно простой и существенно повышает эффективность. PMC и Kelemen MLT хорошо работают на GPU, но для Kelemen MLT всё ещё может быть заметна проблема начального смещения [86].

7.3 Каустики второго рода

Каустиками второго рода мы называем SDS каустики, видимые в зеркале, стекле или под водой. То есть это такие каустики, которые не могут быть спроецированы в камеру. Во-первых, такие каустики не могут быть вычислены при помощи двунаправленных методов IBPT, BPT, PCBPT. За исключением CC-BPT [14], который является существенным усложнением BPT, т.к. требует наличия инструментария дифференциальной геометрии.

Упомянутые простые методы на основе MCMC в принципе с ними справляются, и RELT [80] лучше, чем Kelemen MLT [75]. При этом расчёт всё ещё остаётся несмещённым. Поэтому при длительном расчёте (для получения эталонного изображения) эти методы являются хорошим выбором.

Однако, если необходимо получить приближённое решение быстро, рекомендуется применять методы на основе фотонных карт (в порядке увеличения сложности реализации): SPPM [80], PCLT [75], вариации BDPM без MIS [33. 35-36], с MIS [25, 37], полноценный VCM [24]. Дополнительное применение марковских цепей к фотонным картам способно улучшить их эффективность: MCPPM [39-40] и MBE [41] при длительном расчёте.

7.4 Многократные глянцевые отражения

Ахиллесовой пятой фотонных карт являются многократные глянцевые отражения и сложные составные материалы (кроме MBE [41], который является в значительной степени гибридным методом). Поскольку в индустриальных системах компьютерной графики материал может быть образован из нескольких BSDF в виде графа, вычисление BSDF становится дорогой операцией. Особенно при наличии процедурных текстур, использующих различные шумовые функции. В гибридных методах [33, 35-36] и MBE [41] приходится разделять BSDF на ламбертовскую часть, для которой используются фотонные карты, и не-ламбертовскую, для которой они не используются. Это в значительной степени усложняет разработку. Вычислять BSDF на каждый фотон является непозволительно дорогой (для составных материалов) и неэффективной (для глянцевых отражений) операцией, поскольку большая часть фотонов приходит с направлений, не участвующих в формировании изображения.

Упомянутые простые методы на основе MCMC (Kelemen MLT [75] и PMC [105-106]) прекрасно справляются с многократными глянцевыми отражениями и не вносят существенной дополнительной сложности при работе с составными материалами (для RELT [80] это не совсем так). Метод MMLT [78] особенно хорошо подходит для данного феномена освещённости.

7.5 Микрорельеф и разномасштабные сцены

Использование карт нормалей для имитации микрорельефа можно рассматривать как развитие функциональности BSDF, являющееся в некотором смысле следующим пунктом после глянцевых отражений. Во-первых, популярные микрофасетные модели BSDF используют в своей основе вариацию нормали. Во-вторых, при увеличении частоты изменения нормали (например, путём умножения текстурной координаты на некоторое большое число), зеркальная поверхность с микрорельефом начинает выглядеть как глянцевая. Однако, на практике мелкие детали могут встречаться не только из-за имитации микрорельефа, но и в действительности возникать из-за различного масштаба отдельных элементов сцены.

Использование MLT вместе с BPT или MMLT [78] повышает устойчивость расчёта для разномасштабных сцен (см. разд. 6) и сцен с картами нормалей за счёт использования стратегий с промежуточными соединениями. К сожалению, как BPT, так и MMLT являются существенно более сложными и ограниченными (в первую очередь симметрией) методами, чем однонаправленные Kelemen MLT [75] и PMC [105-106].

Фотонные карты и методы на их основе для подобных сцен следует использовать аккуратно, т.к. для них трудно подобрать правильный радиус сбора [21] и получить корректный вид поверхности при наличии микрорельефа.

7.6 Устойчивость, несмещённость, математическая обоснованность

Мы рассмотрели несколько ключевых феноменов освещённости, влияющих на сложность сцен. Однако мы не учли ещё многие феномены, такие как интегрирование объема [112], спектральный расчёт, поляризацию света и многое другое. Такие феномены будут увеличивать размерность пространства интегрирования на каждом переотражении, что в сочетании с ростом желаемой глубины просчёта (максимально допустимое количество переотражений) может доводить размерность пространства интегрирования до 100 и более.

Если важно сделать устойчивое и универсальное решение, необходимо обращать внимание на свойства алгоритма, которые это гарантируют. Для методов на основе OMC – это, прежде всего, многократная выборка по значимости: BPT/PCBPT [7], VCM [24], CMIS [57]. Для методов на основе MCMC – это теория Монте-Карло методов при помощи марковских цепей и удачное использование особенностей пространства интегрирования: RELT [80], MMLT [78], HHMC [94] и LangevinMC [96].

7.7 Использование GPU

Графические процессоры вводят два существенных ограничения. Во-первых, объём памяти, который метод использует на 1 поток, необходимо сокращать. Поэтому здесь мы отдаём предпочтение методу IBPT [6] против, например, BPT/PCBPT или VCM, для которых необходимо существенно изменять схему вычисления весов многократной выборки по значимости только лишь для того, чтобы сэкономить память [61, 71] (Recursive MIS). Кроме того, с учётом появления аппаратного ускорения трассировки лучей в современных GPU, мы отдаём предпочтение полностью несмещённым методам против фотонных карт: трассировка луча становится существенно дешевле, чем сбор освещённости из фотонной карты, а общее количество Монте-Карло выборок, которые мы можем сделать, возрастает в десятки раз. Для такого количества более медленная сходимость фотонных карт $O(\frac{1}{\sqrt[3]{N}})$ [21] становится заметна.

Во-вторых, для методов на основе MCMC необходимо обращать внимание на то, что длина марковских цепей на GPU будет существенно меньше, чем на CPU. Поэтому проблема начального смещения в MLT на GPU более заметна [86]. Лучше всего здесь подходят методы с короткими цепями PMC [106] и ERPT [105]. Терпимо справляется Kelemen MLT при

помощи параллельного прожига [86] (RELT должен обладать теми же свойствами), а вот в MMLT из-за более сложного пространства интегрирования эта проблема уже становится критической [87].

7.8 Анимация

В [1] говорится, что почти все MCMC алгоритмы страдают от временной нестабильности (что недопустимо в анимации), и эта проблема в настоящее время ещё не решена. В киноиндустрии действительно в основном используются методы на основе OMC, а вместо марковских цепей для повышения эффективности предпочитают использовать Path Guiding [42-46]. Безусловно, MCMC методы имеют существенные недостатки при расчёте анимации. Однако, в целом ситуация не столь драматична.

Во-первых, в киноиндустрии редко требуется вычислять сложные феномены освещённости с высокой точностью. Достаточно добиться правдоподобности освещения. Поэтому можно, например, просто обрезать выбросы, присутствующие в Path Guiding, или применить более прогрессивный подход [110]. Иногда, если это требуется, применяют фотонные карты как простой и быстрый способ приближённо посчитать сложные феномены освещённости с SDS путями.

Во-вторых, для повышения регулярности обхода пространства изображения всегда можно просто увеличить вероятность большого шага до 50% и смешивать OMC (в качестве которого можно брать большие шаги) и MCMC при помощи многократной выборки по значимости, что обычно и делается в Kelemen MLT [75]. С этой точки зрения у Path Guiding нет существенных преимуществ перед PMC или Kelemen MLT, особенно при расчёте на GPU, где обновление динамических структур является нетривиальным действием.

В-третьих, можно применять стратификацию [83].

В-четвертых, временной стабильности в MLT можно добиться, если добавить временные мутации к цепи и рассчитывать несколько кадров одновременно [111].

Наконец, стоит учитывать возможности применения современных методов фильтрации и шумоподавления, особенно с учётом соседних кадров.

8. Заключение

Таким образом, за последние 10 лет появилось много новых расчётных методов, и некоторые из них можно считать чрезвычайно интересными в плане эффективности решения фудаментально трудной задачи. Заявка, сделанная методами на основе HMC [94, 96, 102], выглядит многообещающе. В этом смысле по сравнению с 2010 годом в науке расчёта глобального освещения произошёл настоящий прорыв. Мы полагаем, что за этим прорывом должен последовать соответствующий прорыв и в прикладной науке, поскольку вопрос применения многих из новых методов на практике остаётся открытым в силу большого количества ограничений. Например, система с аппаратом дифференцирования существует (Mitsuba2 [101]), однако HMC в ней не реализован.

В сегодняшней практике имеет смысл отдавать предпочтение тем методам, которые не имеют большого набора ограничений. Например, мы считаем крайне важным возможность эффективной реализации метода на GPU с учётом всё возрастающей поддержки аппаратной трассировки лучей в современных картах. По этой же причине мы полагаем, что полностью несмещённые методы более ценны, т.к. стоимость трассировки лучей по отношению к сбору освещённости из фотонной карты при использовании аппаратного ускорения существенно падает (в 5-10 раз на последних моделях GPU [59-60]). С этой точки зрения более перспективными выглядят методы IBPT [6], Kelemen MLT [75], PMC [105-106], RELT [80-81]. С другой стороны, если важно уметь получать приближённое решение быстро, то можно использовать различные методы на основе фотонных карт: SPPM [19], VCM [24], регуляризации путей [77], PCLT [30].

Список литературы / References

[1] Sik M. and Krivanek J. Survey of Markov Chain Monte Carlo Methods in Light Transport Simulation. IEEE Transactions on Visualization and Computer Graphics, vol. 26, issue 4, 2018, pp. 1821-1840.

[2] Kajiya J.T. The rendering equation. ACM SIGGRAPH Computer Graphics, 1986, vol. 20, no. 4, pp. 143-150.

[3] Veach E. Robust monte carlo methods for light transport simulation. Ph.D. Dissertation, Stanford University, 1998, 406 p.

[4] Ershov S.V., Voloboy A.G. Calculation of MIS weights for bidirectional path tracing with photonmaps in presence of direct illumination. Mathematica Montisnigri, vol. 48, 2020, pp. 86-102.

[5] Pharr M., Jakob W., and Humphreys G. Physically Based Rendering: From Theory to Implementation (3rd. ed.). Morgan Kaufmann Publishers, 2016, 1266 p.

[6] Bogolepov D., Ulyanov D. GPU-Optimized Bidirectional Path Tracing. In Proc. of the 21th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision. 2013, pp. 1-15.

[7] Popov S., Ramamoorthi R., Durand F., Drettakis G. Probabilistic Connections for Bidirectional Path Tracing. Computer Graphics Forum, vol. 34, no. 4, 2015, pp. 75–86.

[8] Walter B., Fernandez S., Arbree A. et al. A Scalable Approach to Illumination. ACM Transactions on Graphics, vol. 24, no. 3, 2005, pp. 1098-1107.

[9] Moreau P., Clarber P. Importance Sampling of Many Lights on the GPU. In Ray Tracing Gems: High-Quality and Real-Time Rendering with DXR and Other APIs. Apress, 2019, pp. 255-283.

[10] Schussler V., Heitz E., Hanika J. and Dachsbacher C. Microfacet-based normal mapping for robust Monte Carlo path tracing. ACM Transactions on Graphics, vol. 36, no. 6, 2017, pp. 1-12.

[11] Baek S.H., Jeon D.S., Tong X., Kim M.H. Simultaneous acquisition of polarimetric SVBRDF and normal. ACM Transactions on Graphics, vol. 37, no. 6, 2018, pp. 1-15.

[12] Bar C., Alterman M., Gkioulekas I., Levin A. A Monte Carlo framework for rendering speckle statistics in scattering media. ACM Transactions on Graphics, vol. 38, no. 4, 2019, pp. 1-22.

[13] Zhdanov A., Zhdanov D., Sokolov V. et al. Problems of the realistic image synthesis in media with a gradient index of refraction. Proc. of the SPIE 11548, Optical Design and Testing X. 2020, 115480W.

[14] Speierer S., Hery C., Villemin R., Jako W. Caustic Connection Strategies for Bidirectional Path Tracing. Pixar Technical Memo no. 18-01, 2018, 6 p.

[15] Wenzel J. Light Transport on Path-Space Manifolds. Ph.D. Dissertation. Cornell University, 2013, 153 p.

[16] Johannes H., Droske M., and Fascione L. Manifold next event estimation. Computer Graphics Forum, vol. 34, no. 4, 2015, pp. 87-97.

[17] Jensen H.W. Global Illumination using Photon Maps. In Proc. of Eurographics Workshop on Rendering, 1996, pp. 21-30.

[18] Hachisuka T., Ogaki S., and Jensen H.W. Progressive photon mapping. ACM Transactions on Graphics, vol. 27, no. 5, 2008, pp. 1-8.

[19] Hachisuka T., and Jensen H.W. Stochastic progressive photon mapping. ACM Transactions on Graphics, vol. 28, no. 5, 2009, pp. 1-8.

[20] Spencer B. and Jones M.W. Progressive photon relaxation. ACM Transactions on Graphics, vol. 32, no. 1, 2013, PP. 1-11.

[21] Kaplanyan A.S., Dachsbacher C. Adaptive progressive photon mapping. ACM Transactions on Graphics, vol. 32, no. 2, 2013, pp. 1-13.

[22] Suykens F., Willems Y.D. Density Control for Photon Maps. In Proc. of the Eurographics Workshop on Rendering, 2000, pp. 23-34.

[23] Jarosz W., Jensen H.W., and Donner C. Advanced global illumination using photon mapping. In ACM SIGGRAPH 2008 classes (SIGGRAPH ’08), 2008, pp. 1-112.

[24] Georgiev I., Krivanek J., Davidovic T., Slusallek P. Light Transport Simulation with Vertex Connection and Merging. ACM Transactions on Graphics, vol. 31, no. 6, 2012, pp. 1-10.

[25] Hachisuka T., Pantaleoni J., Jensen H.W. A path space extension for robust light transport simulation. ACM Transactions on Graphics, vol. 31, no. 6, 2012, pp. 1-10.

[26] Krivanek J., Georgiev I., Hachisuka T. et al. Unifying Points, Beams, and Paths in Volumetric Light Transport Simulation. ACM Transactions on Graphics, vol. 33, no. 4, 2014, pp. 1-13.

[27] Kaplanyan A.S., Dachsbacher C. Path space regularization for holistic and robust light transport. Computer Graphics Forum, vol. 32, no. 2, 2013, pp. 63–72.

[28] Render Legion. Corona Render System. URL: https://corona-renderer.com/, accessed 26.01.2021.
[29] Изотов И. Уроки 3Ds Max. Caustics CORONA. Видеоурок на Youtube, 2018 г. / Izotov I, Lessons 3Ds Max. Caustics CORONA. Pool water with caustic in the crown. Video tutorial on Youtube. 2018.URL: https://www.youtube.com/watch?v=Nv1ULR8sMZY, accessed 26.01.2021 (in Russian).
[30] Jendersie J., Rohmer K., Brüll F., Grosch T. Pixel cache light tracing. Proceedings of the Conference on Vision, Modeling and Visualization. 2017. pp. 137-144.
[31] Havran V., Herzog R., Seidel H. P. Fast final gathering via reverse photon mapping. Computer Graphics Forum, vol. 24, no. 3, 2005, pp. 323-332.
[32] Zhdanov A., Zhdanov D. The Backward Photon Mapping for the Realistic Image Rendering. In Proc. 30th Conf. on Computer Graphics and Machine Vision (GraphiCon 2020), CEUR Workshop Proceedings, 2020, vol. 2744, pp. 1-12.
[33] Жданов Д.Д., Ершов С.В., Волобой А.Г. Адаптивный выбор глубины трассировки обратного луча в методедвунаправленной стохастической трассировки лучей. Труды 25-й Международной конференции GraphiCon, 2015 г., стр. 44–49 / Zhdanov D.D., Ershov S.V., Voloboy A.G. Adaptive estimation of backward ray trace depth for the stochastic bi-directional ray trace method. In Proc. of the 25th Anniversary International Conference GraphiCon, 2015, pp. 44-49 (in Russian).
[34] Жданов А.Д., Жданов Д.Д., Бирюков Е.Д. Реалистичный рендеринг на основе прямых и обратных фотонных карт. Препринт ИПМ № 77, 2020 г., 22 стр. / Zhdanov A.D., Zhdanov D.D., Birukov E.D. The realistic rendering with forward and backward photon mapping. KIAM Preprint № 77, 2020, 22 p. (in Russian).
[35] Ershov S.V., Zhdanov D.D., Voloboy A.G., Deryabin N.B. The method of quasi-specular elements to reduce stochastic noise in illumination simulation. Light and Engineering, no. 5, 2020, pp. 39-47.
[36] Zhdanov A., Zhdanov D., Galaktionov V. Realistic image synthesis with hybrid photon maps. Proceedings of the SPIE 11550, Optoelectronic Imaging and Multimedia Technology VII, 2029, 115500G.
[37] Vorba J. Bidirectional Photon Mapping. In Proc. of the 15th Central European Seminar on Computer Graphics. 2011, pp. 1-8.
[38] Schutte J. Vertex Connection and Merging. Rendering Equations Blog, 2018. URL: https://schuttejoe.github.io/post/vertexconnectionandmerging/, accessed 26.01.2021.
[39] Chen J., Wang B., and Yong J.-H. Improved stochastic progressive photon mapping with metropolis sampling. In Proc. of the 22nd Eurographics Conference on Rendering, 2011, pp. 1205–1213.
[40] Hachisuka T. and Jensen H.W. Robust adaptive photon tracing using photon path visibility. ACM Transactions on Graphics, vol. 30, no. 5, 2011, pp. 1-11.
[41] Sik M., Otsu H., Hachisuka T., and Krivanek J. Robust light transport simulation via metropolised bidirectional estimators. ACM Transactions on Graphics, vol. 35, no. 6, 2016, pp. 1-12.
[42] Vorba J., Hanika J., Herholz S. et al. Path guiding in production. In ACM SIGGRAPH Courses, 2019, pp. 1-77.
[43] Herholz S., Elek O., Vorba J. et al. Product importance sampling for light transport path guiding. Computer Graphics Forum, vol. 35, no. 4, 2016, pp. 67-77.
[44] Guo J., Bauszat P., Bikker J., Eisemann E. Primary sample space path guiding. In Proc. of the Eurographics Symposium on Rendering: Experimental Ideas & Implementations, 2018, pp. 73–82.
[45] Rath A., Grittmann P., Herholz S. et al. Variance-aware path guiding. ACM Transactions on Graphics, vol. 39, no. 4, 2020, pp. 1-12.
[46] Muller T., Gross M., and Novik J. Practical Path Guiding for Efficient Light-Transport Simulation. Computer Graphics Forum, vol. 36, no. 4, 2017, pp. 91-100.
[47] Zwicker M., Jarosz W., Lehtinen J. et al. Recent Advances in Adaptive Sampling and Reconstruction for Monte Carlo Rendering. Computer Graphics Forum, vol. 34, no. 2, 2015, pp. 667-681.
[48] Ershov S.V., Zhdanov D.D., Voloboy A.G., Galaktionov V.A. Two denoising algorithms for bi-directional Monte Carlo ray tracing. Mathematica Montisnigri, vol. 43, 2018, pp. 78-100.
[49] Hachisuka T., Jarosz W., Weistroffer R.P. et al. Multidimensional adaptive sampling and reconstruction for ray tracing. ACM Transactions on Graphics, vol. 27, no. 3, 2008, pp. 1-10.
[50] Kettunen M., Manzi M., Aittala M. et al. Gradient-domain path tracing. ACM Transactions on Graphics, vol. 34, no. 4. 2015, pp. 1-13.
[51] Manzi M., Kettunen M., Aittala M. et al. Gradient-Domain Bidirectional Path Tracing. In Proc. of the Eurographics Symposium on Rendering - Experimental Ideas & Implementations, 2015, pp. 65-74.
[52] Manzi M., Kettunen M., Durand F. et al. Temporal gradient-domain path tracing. ACM Transactions on Graphics, vol. 35, no. 6, 2016, pp. 1-9.

[53] Hua B.S., Gruson A., Nowrouzezahrai D., Hachisuka T. Gradient-domain Photon Density Estimation. Computer Graphics Forum, vol. 36, no. 2, 2017, pp. 31-38.
[54] Lehtinen J., Karras T., Laine S. et al. Gradient-domain metropolis light transport. ACM Transactions on Graphics, vol. 32, no. 4, 2013, pp. 1-12.
[55] Krivanek J., Bouatouch K., Pattanaik S., Žára J. Making Radiance and Irradiance Caching Practical: Adaptive Caching and Neighbor Clamping. In Proc. of the 17th Eurographics Conference on Rendering Techniques, 2006, pp. 127–138.
[56] Schwarzhaupt J., Jensen H.W., Jarosz W. Practical Hessian-based error control for irradiance caching. ACM Transactions on Graphics, vol. 31, no. 6, 2012, pp. 1-10.
[57] West R., Georgiev I., Gruson A., and Hachisuka T. Continuous multiple importance sampling. ACM Transactions on Graphics, vol. 39, no. 4, 2020, pp. 1-12.
[58] Bitterli B., Wyman C., Pharr M. et al. Spatiotemporal reservoir resampling for real-time ray tracing with dynamic direct lighting. ACM Transactions on Graphics, vol. 39, no. 4, 2020, pp. 1-17.
[59] Санжаров В.В., Фролов В.А., Галактионов В.А. Исследование технологии Nvidia RTX. Программирование, том 46, no. 4, 2020 г., стр. 65-72 / Sanzharov V.V., Frolov V.A., and Galaktionov V.A. Survey of Nvidia RTX Technology. Programming and Computer Software, vol. 46, no. 4, 2020, pp. 297–304.
[60] Meister D., Boksansky J., Guthe M., Bittner J. On Ray Reordering Techniques for Faster GPU Ray Tracing. In Proc. of the Symposium on Interactive 3D Graphics and Games, 2020, pp. 1-9.
[61] Van Antwerpen D. Recursive MIS Computation for Streaming BDPT on the GPU. Technical report, Delft University of Technology, 2011, 12 p.
[62] Nocak J., Havran V., and Daschbacher C. Path regeneration for interactive path tracing. In Proc. of the 31st Annual Conference of the European Association for Computer Graphics – Short Papers, 2010, pp.61-64.
[63] Van Antwerpen D. Unbiased physically based rendering on the GPU. M.S. Thesis, Delft University of Technology, 2011, 180 p.
[64] Фролов В.А., Галактионов В.А. Регенерация путей с низкими накладными расходами. Программирование, том 42, no. 6, 2016 г., стр. 67-74 / Frolov V.A., Galaktionov V.A. Low overhead path regeneration. Programming and Computer Software, vol. 42, no. 6, 2016, pp. 382-387.
[65] Fabianowski B., Dingliana J. Interactive Global Photon Mapping. Computer Graphics Forum, vol. 28, no. 4, 2009, pp. 1151–1159.
[66] Garanzha K., Pantaleoni J., McAllister D. Simpler and Faster HLBVH with Work Queues. In Proc. of the ACM SIGGRAPH Symposium on High Performance Graphics, 2011, pp. 59–64.
[67] Фролов В.А., Харламов А.А., Галактионов В.А., Востряков К.А. Окто-деревья со множественными ссылками в применении к реализации фотонных карт и кэша освещенности на GPU. Программирование, том 40, no. 4, 2014 г., стр. 64-73 / Frolov V.A., Kharlamov A.A., Galaktionov V.A., Vostryakov K.A. Multiple reference octrees for a GPU photon mapping and irradiance caching. Programming and Computer Software, vol. 40, no. 4, 2014, pp. 208–214.
[68] Hachisuka T. and Jensen H.W. Parallel progressive photon mapping on GPUs. In Proc. of the 3rd ACM SIGGRAPH Conference and Exhibition on Computer Graphics and Interactive Techniques in Asia, Sketches, 2010, pp. 1-1.
[69] Garanzha K., Pantaleoni J., and McAllister D. Simpler and faster HLBVH with work queues. In Proc. of the ACM SIGGRAPH Symposium on High Performance Graphics, 2011, pp. 59–64.
[70] Karras T. Maximizing Parallelism in the Construction of BVHs, Octrees, and k-d Trees. In Proc. of the 4th ACM SIGGRAPH / Eurographics conference on High-Performance Graphics, 2012, pp. 33-37.
[71] Davidovic T., Krivanek J., Hasan M., and Slusallek P. Progressive Light Transport Simulation on the GPU: Survey and Improvements. ACM Transactions on Graphics, vol. 33, no. 3, Article 29, 2014, pp. 1-19.
[72] Laine S., Karras T., and Aila T. Megakernels considered harmful: wavefront path tracing on GPUs. In Proc. of the 5th High-Performance Graphics Conference, 2013, pp. 137-143.
[73] Veach E., Guibas L.J. Metropolis Light Transport. In Proc. of the of the 24th Annual Conference on Computer Graphics and Interactive Techniques, 1997, pp. 65-76.
[74] Kiivanek J, Georgiev I., Kaplanyan A.S., Canada J. Recent Advances in Light Transport Simulation: Theory and Practice. In ACM SIGGRAPH Courses, 2013, pp. 1-5.
[75] Kelemen C, Szirmay-Kalos L., Antal G., Csonka F. A Simple and Robust Mutation Strategy for the Metropolis Light Transport Algorithm. Computer Graphics Forum, vol. 21, no. 3, 2002, pp. 531-540.

[76] Ashikhmin M., Premoze S., Shirley P., Smits B. A Variance Analysis of the Metropolis Light Transport Algorithm. Computers and Graphics, vol. 25, issue 2, 2001, pp. 287-294.
[77] Kaplanyan A.S., Hanika J., Dachsbacher C. The Natural-constraint Representation of the Path Space for Efficient Light Transport Simulation. ACM Transactions on Graphics, vol. 33, no. 4, 2014, pp. 1-13.
[78] Hachisuka T., Kaplanyan A.S., Dachsbache C. Multiplexed Metropolis Light Transport. ACM Transactions on Graphics, vol. 33, no. 4, 2014, pp. 1-10.
[79] Swendsen R.H. and Wang J.-S. Replica Monte Carlo simulation of spin-glasses. Physical Review Letters, vol. 57, issue 21, 1986, pp. 2607–2609.
[80] Kitaoka S., Kitamura Y., Kishino F. Replica exchange light transport. Computer Graphics Forum, vol. 28, no. 8, 2009, pp. 2330-2342.
[81] Otsu H., Yue Y., Hou Q. et al. Replica exchange light transport on relaxed distributions. In ACM SIGGRAPH Posters, 2013, pp. 1-1.
[82] Earl D.J. and Deem M.W. Parallel tempering: Theory, applications, and new perspectives. Physical Chemistry Chemical Physics, vol. 7, issue 23, 2005, pp. 3910-3916.
[83] Gruson A., West R., Hachisuka T. Stratified Markov Chain Monte Carlo Light Transport. Computer Graphics Forum, vol. 39, no. 2, 2020, pp. 351-362.
[84] Schmidt M., Lobachev O., Guthe M. Coherent Metropolis Light Transport on the GPU using Speculative Mutations. Journal of WSCG, vol. 24, no. 1, 2016, pp. 1-9.
[85] Antwerpen D. Improving SIMD efficiency for parallel Monte Carlo light transport on the GPU. In Proc. of the ACM SIGGRAPH Symposium on High Performance Graphics, 2011, pp. 41-50.
[86] Фролов В.А., Галактионов В.А. Компактная по памяти реализация алгоритма Metropolis light transport на графических процессорах. Программирование, том 43, no. 3, 2017 г., стр. 83-92 / Frolov V.A., Galaktionov V.A. Memory compact Metropolis light transport on GPUs. Programming and Computer Software, vol. 43, no. 3, 2017, pp. 196-203.
[87] Фролов В.А. Исследование алгоритма Multiplexed Metropolis Light Transport на графических процессорах. Препринт ИПМ № 267, 2018 г., 47 стр. / Frolov V.A. Investigation of Multiplexed Metropolis Light Transport on GPUs. KIAM Preprint № 267, 2018, 47 p. (in Russian).
[88] Bitterli B., Jakob W., Novak J., and Jarosz W. Reversible Jump Metropolis Light Transport Using Inverse Mappings. ACM Transactions on Graphics, vol. 37, no. 1, 2017, pp. 1-12.
[89] Otsu H., Kaplanyan A. S., Hanika J. et al. Fusing state spaces for markov chain Monte Carlo rendering. ACM Transactions on Graphics, vol. 36, no. 4, 2017, pp. 1-10.
[90] Pantaleoni J. Charted metropolis light transport. ACM Transactions on Graphics, vol. 36, no. 4, 2017, pp. 1-14.
[91] Brooks S., Gelman A., Jones G., Meng X. MCMC using Hamiltonian dynamics. In Handbook of Markov Chain Monte Carlo. Chapman and Hall/CRC, 2011, pp. 113-163.
[92] Duane S., Kennedy A.D., Pendleton B.J., and Roweth D. Hybrid Monte Carlo. Physics Letters, vol. 195, issue 2, 1987, pp. 216–222.
[93] Michael Betancourt. A Conceptual Introduction to Hamiltonian Monte Carlo. arXiv:1701.02434, 2017, 60 p.
[94] Li T.-M., Lehtinen J., Ramamoorthi R. et al. Anisotropic Gaussian mutations for metropolis light transport through Hessian-Hamiltonian dynamics. ACM Transactions on Graphics, vol. 34, no. 6, 2015, pp. 1-13.
[95] Girolami M., Calderhead B. Riemann manifold Langevin and Hamiltonian Monte Carlo methods. Journal of the Royal Statistical Society: Series B (Statistical Methodology), vol. 73, issue 2, 2011, pp. 123-214.
[96] Luan F., Zhao S., Bala K., and Gkioulekas I. Langevin Monte Carlo rendering with gradient-based adaptation. ACM Transactions on Graphics, vol. 39, no. 4, 2020, pp. 1-16.
[97] Roberts G.O., Tweedie R.L. Exponential convergence of Langevin distributions and their discrete approximations. Bernoulli, vol. 2, no. 4, 1996, pp.341-363.
[98] Duong M.H., Lamacz A., Peletier M.A. et al. Quantification of coarsegraining error in Langevin and overdamped Langevin dynamics. arXiv:1712.09920, 2017, 48 p.
[99] Leal R.J. CppADCodeGen. Source Code Generation for Automatic Differentiation using Operator Overloading. GitHub. 2011. URL: https://github.com/joaoleal/CppADCodeGen, accessed 26.01.2021.
[100] Vassilev V., Vassilev M., Penev A. et al. Clad – Automatic Differentiation Using Clang and LLVM. Journal of Physics: Conference Series, vol. 608, no. 1, 2015, pp. 1-10.
[101] Nimier-David M., Vicini D., Zeltner T., and Jakob W. Mitsuba 2: a retargetable forward and inverse renderer. ACM Transactions on Graphics, vol. 38, no. 6, 2019, pp. 1-17.

[102] Rioux-Lavoie D., Litalien J., Gruson A., Hachisuka T. Delayed Rejection Metropolis Light Transport. ACM Transactions on Graphics, vol. 39, no. 3, 2020, pp. 1-14.
[103] Atchade Y.F. An adaptive version for the Metropolis adjusted Langevin algorithm with a truncated drift. Methodology and Computing in Applied Probability, vol.8, no. 2, 2006, pp. 235-254.
[104] Roberts G.O., Stramer O. Langevin diffusions and Metropolis-Hastings algorithms. Methodology and Computing in Applied Probability, vol. 4, no. 4, 2002, pp. 337-357.
[105] Cline D., Talbot J., Egbert P. Energy redistribution path tracing. ACM Transactions on Graphics, vol. 24, no. 3, 2005, pp. 1186-1195.
[106] Lai Y., Fan S.H., Chenney S., Dyer C. Photorealistic image rendering with population Monte-Carlo energy redistribution. In Proc. of the 18th Eurographics Conference on Rendering Techniques, 2007, pp. 287–295.
[107] OTOY. Octane Render. URL: https://home.otoy.com/render/octane-render/, accessed 26.01.2021.
[108] Hydra Renderer. Frolov V., Snazharov V., Trofimov M., Galaktionov V. Hydra Renderer. Open Source GPU Rendering System. GitHub, 2019. URL: https://github.com/Ray-Tracing-Systems/HydraAPI, accessed 10.02.2021.
[109] Haindl M., Filip J. Spatially Varying Bidirectional Reflectance Distribution Functions. In Visual Texture: Accurate Material Appearance Measurement, Representation and Modeling, Springer, 2013, pp 119-145.
[110] Zirr T., Hanika J., Dachsbacher C. Re-Weighting Firefly Samples for Improved Finite-Sample Monte Carlo Estimates. Computer Graphics Forum, vol. 37, no. 6, pp. 410-421.
[111] Van de Woestijne J., Frederickx R., Billen N., and Dutre P. Temporal coherence for metropolis light transport. In Proc. of the Eurographics Symposium on Rendering: Experimental Ideas and Implementations, 2017, pp. 55–63.
[112] Novák J. et al. Monte Carlo methods for volumetric light transport simulation // Computer Graphics Forum, vol. 37. no. 2, 2018, pp. 551-576

Информация об авторах / Information about authors

Владимир Александрович ФРОЛОВ – кандидат физико-математических наук, старший научный сотрудник ИПМ РАН, научный сотрудник факультета ВМК МГУ. Сфера научных интересов: реалистичная компьютерная графика, моделирование освещённости, разработка программных систем оптического моделирования, параллельные и распределённые вычисления.

Vladimir FROLOV – PhD in computer graphics, senior researcher in the KIAM and researcher in computer graphics of Moscow State University. Research interests: realistic computer graphics, light transport simulation, elaboration of optical simulation software systems, GPU computing.

Алексей Геннадьевич ВОЛОБОЙ, ведущий научный сотрудник, доктор физико-математических наук, доцент. Научные интересы: компьютерная графика, оптика, оптическое моделирование.

Alexey Gennadievich VOLOBOY, Leading Researcher, Doctor of Physical and Mathematical Sciences, Associate Professor. Research interests: computer graphics, optics, optical modeling.

Сергей Валентинович ЕРШОВ – кандидат физико-математических наук, старший научный сотрудник, доцент. Научные интересы: компьютерная графика, моделирование освещённости, научная визуализация.

Sergey Valentinovich ERSHOV, PhD, Senior Researcher, Associate Professor. Research interests: computer graphics, light transport simulation, scientific visualization.

Владимир Александрович ГАЛАКТИОНОВ, главный научный сотрудник, доктор физико-математических наук, профессор. Научные интересы: компьютерная графика, вычислительная оптика, компьютерная лингвистика, научная визуализация.

Vladimir Alexandrovich GALAKTIONOV, Chief Researcher, Doctor of Physical and Mathematical Sciences, Professor. Research interests: computer graphics, computational optics, computer linguistics, scientific visualization.