### \_\_\_\_\_ КОМПЬЮТЕРНАЯ ГРАФИКА \_\_ И визуализация

УДК 004.92

## ЗАВИСИМОСТЬ ШУМА ОТ ЧИСЛА ЛУЧЕЙ В ДВУНАПРАВЛЕННОЙ СТОХАСТИЧЕСКОЙ ТРАССИРОВКЕ ЛУЧЕЙ С ФОТОННЫМИ КАРТАМИ

© 2021 г. С. В. Ершов<sup>а</sup>, Е. Д. Бирюков<sup>а</sup>, А. Г. Волобой<sup>а</sup>, В. А. Галактионов<sup>а,\*</sup>

<sup>а</sup>Институт прикладной математики им. М.В. Келдыша РАН, 125047 Москва, Миусская пл., д. 4, Россия \*E-mail: voloboy@gin.keldysh.ru Поступила в редакцию 19.12.2020 г. После доработки 25.12.2020 г. Принята к публикации 14.01.2021 г.

Классическая трассировка лучей методом Монте-Карло – это мощный метод, позволяющий моделировать практически все эффекты в лучевой оптике, но он может быть недопустимо медленным для многих случаев, таких как, например, вычисление изображений, видимых объективом или камерой с точечным отверстием. Поэтому часто используются его различные модификации, в частности, двунаправленная стохастическая трассировка лучей с фотонными картами. Недостатком всех стохастических методов является нежелательный шум. Уровень шума, то есть дисперсия яркости пикселей, рассчитанной за одну итерацию метода, зависит от различных параметров, таких как количество лучей от источника света и от камеры, способ слияния их траекторий, радиус интегрирующей сферы и т.д. Выбор оптимальных параметров позволит получить минимальный уровень шума при данном времени расчета. Данной проблеме и посвящена эта статья. Показано, что дисперсия яркости пикселя представляет собой сумму трех функций, масштабируемых обратным числом лучей из источника и из камеры, причем сами эти функции не зависят от количества лучей. Поэтому, зная их, можно предсказать шум для любого количества лучей и, таким образом, найти оптимальный вариант. Вычисление этих функций на основе полученных в трассировке лучей данных является нетривиальной задачей. В статье приведен практический метод их расчета и продемонстрировано, что по результатам всего одного пробного расчета можно предсказать дисперсию для произвольного числа лучей. Таким образом, становится возможным минимизация шума благодаря выбору оптимального числа лучей.

DOI: 10.31857/S0132347421030043

#### 1. ВВЕДЕНИЕ

В настоящее время моделирование распространения света широко используется в реалистичной компьютерной графике, проектировании новых материалов и оптических систем [1]. Оно интенсивно применяется в архитектурных, автомобильных и авиационных конструкторских задачах. Если задача позволяет пренебречь волновыми эффектами, то хорошим выбором является группа методов стохастической трассировки лучей. Эта область главным образом включает моделирование переноса излучения методом Метрополиса [2] и стохастическую трассировку лучей [3]. При расчете изображения классическая прямая трассировка лучей от источника света неэффективна и поэтому заменяется двунаправленными модификациями метода [4-6]. Из них рассмотрим так называемую двунаправленную стохастическую трассировку лучей с фотонными картами (BDPM – Bidirectional Photon Mapping) [5, 7]. Слабой стороной всех стохастических методов

является то, что их результаты зашумлены. Поэтому задача снижения шума всегда актуальна, ей посвящено большое число работ, в частности, можно выделить работы [8—10].

Уровень шума в методе BDPM в основном зависит от способа случайного рассеяния прямого и обратного лучей, от выбора вершины для их слияния (или, другими словами, в какой вершине траектории камеры использовать фотонные карты для оценки освещенности) и, наконец, от количества прямых и обратных лучей, протрассированных за одну итерацию. Большинство публикаций посвящено первым двум средствам, например, [9-12], в то время как меньшее внимание было уделено количеству лучей. Между тем это важный фактор, и часто бывает так, что, например, количество прямых лучей уже избыточно, поэтому дальнейшее его увеличение не уменьшает шум, а только увеличивает время выполнения. В других случаях бывает так, что количество прямых лучей

**Таблица 1.** Среднее значение RMS по части изображения (отмеченной красным квадратиком на рис. 2) после 1000 секунд вычисления для различного числа лучей  $N_F$ ,  $N_B$ 

$N_F   N_B$	5	10	25	100
100	0.135	0.135	0.135	0.135
1000	0.101	0.101	0.101	0.101
3000	0.099	0.099	0.099	0.099
10000	0.101	0.101	0.101	0.101
30000	0.110	0.110	0.109	0.109
100 000	0.136	0.136	0.135	0.135

действительно критично, в то время как мы трассируем избыточное количество обратных лучей.

Подобное положение иллюстрируется табл. 1, где показана зависимость дисперсии (RMS) после фиксированного времени расчета от числа прямых лучей (из источника света)  $N_F$  и обратных (от камеры через пиксель)  $N_B$ . Описание сцены приведено в разделе *Результаты*. В приведенном примере зависимость от числа обратных лучей  $N_B$  (приведенных по столбцам) практически отсутствует, а вот по числу лучей из источника имеется минимум (шума) при  $N_F \approx 3000$ . Здесь мы видим, что общее утверждение "чем больше лучей, тем меньше шум", очевидно, ошибочно.

Обычно трудно предсказать, какая доля прямых и обратных лучей является оптимальной, в то время как ее хороший выбор может ускорить расчеты в несколько раз.

В этой статье мы рассмотрим последнюю проблему. В работе [8] был выведен общий закон, определяющий шум в BDPM. Он утверждает, что дисперсия яркости пикселя представляет собой сумму трех составляющих, масштабированных обратным числом лучей. Эти 3 компоненты не зависят от количества лучей и, таким образом, однажды вычисленные они могут быть позже использованы для предсказания того, как уровень шума будет зависеть от количества лучей, протрассированных за одну итерацию. Таким образом, предполагается прогнозировать оптимальное количество лучей. Другими словами, если мы знаем эти 3 компоненты, которые обычно зависят от пикселя, то можем предсказать уровень шума для любого количества лучей.

Хотя их математическое определение тривиально, эти три значения не так легко вычислить численно в процессе трассировки лучей. В данной работе описан метод их эффективного расчета и продемонстрировано их использование для выбора оптимального числа лучей.

#### 2. ШУМ В ДВУНАПРАВЛЕННОЙ ТРАССИРОВКЕ ЛУЧЕЙ С ФОТОННЫМИ КАРТАМИ

Обычно вычисления в BDPM идут итерациями. На каждой итерации мы трассируем N<sub>F</sub> световых путей, и  $N_{R}(p)$  путей из камеры для каждого пикселя р. Затем проверяем каждую пару, и если световой путь проходит достаточно близко с узлом пути из камеры, соединяем их, получая полный путь (от камеры до источника). Затем мы вычисляем вклад этого объединенного пути в яркость пикселя и увеличиваем яркость пикселя на это значение. Накопленная сумма, деленная на произведение  $N_F N_B(p) N_I$  (где  $N_I$  – число итераций), сходится к математическому ожиданию яркости. Это среднее значение не зависит от  $N_F$  и  $N_B(p)$ , а вот его дисперсия (шум) может весьма сильно меняться. Скажем, если световых лучей уже слишком много, а траекторий лучей из камеры мало, то трассировка этих избыточных световых лучей только тратит время без улучшения качества изображения, поэтому выгодно уменьшить их количество.

Оптимальное количество лучей (общее для каждой сцены для световых путей и индивидуальное для каждого пикселя для путей из камеры) это то, которое приводит к наименьшему шуму после фиксированного времени расчета. Для решения этой задачи минимизации нам нужно знать, как дисперсия значения, полученного за одну итерацию, зависит от этого числа лучей. Эта зависимость описывается несложной алгебраической формулой [8], однако ее члены не так легко найти численно.

Здесь и ниже мы всегда будем говорить о значениях для одного, заданного пикселя p и поэтому отбросим эту нотацию, написав  $N_B$  вместо  $N_B(p)$ .

Как показано в [8], дисперсия яркости в данном пикселе, *рассчитанной за одну итерацию*, равна

$$\begin{split} V &= \frac{1}{N_F N_B} (\langle \langle C^2 \rangle \rangle - \langle \langle C \rangle \rangle^2) + \\ &+ \frac{1 - N_F^{-1}}{N_B} (\langle \langle C \rangle_F^2 \rangle_B - \langle \langle C \rangle \rangle^2) + \\ &+ \frac{1 - N_B^{-1}}{N_F} (\langle \langle C \rangle_B^2 \rangle_F - \langle \langle C \rangle \rangle^2), \end{split}$$

где  $C(X^{(F)}, X^{(B)}) - вклад$  в яркость пикселя от "слияния" пути из источника  $X^{(F)}$  и пути из камеры  $X^{(B)}$ , его среднее  $\langle \langle C \rangle \rangle$  очевидно совпадает с предельной (точной) яркостью пикселя L, а  $\langle \cdot \rangle_F$  и  $\langle \cdot \rangle_B$ обозначают усреднение по ансамблю путей из источника и из камеры, соответственно.

Заметим, что сама эта функциональная форма не зависит от того, используется ли Multiple Im-

рогапсе Sampling [13] или нет, и даже от того, используем мы слияние вершин (Vertex Merging) или соединение вершин дополнительным сегментом траектории (Vertex Connection) [9]. Хотя величины  $\langle \langle C^2 \rangle \rangle$ ,  $\langle \langle C \rangle_F^2 \rangle_B$  и  $\langle \langle C \rangle_B^2 \rangle_F$  будут меняться в зависимости от конкретной стратегии вычислений. При этом в любом случае они не зависят от числа лучей, хотя могут зависеть от радиуса интегрирующей сферы (т.е. "радиуса слияния вершин").

#### 3. ВЫЧИСЛЕНИЕ КОЭФФИЦИЕНТОВ ФОРМУЛЫ ШУМА

Вычисление  $\langle \langle C^2 \rangle \rangle$  тривиально (точно как при расчете выборочной дисперсии): добавляя *C* к накопленной яркости пикселя, мы заодно добавляем  $C^2$  к накопленному в этом пикселе  $\langle \langle C^2 \rangle \rangle$ . То есть, на каждой итерации мы вычисляем

$$C \equiv \frac{1}{N_B N_F} \sum_{j=1}^{N_B} \sum_{i=1}^{N_F} C^2(X_i^{(F)}, X_j^{(B)})$$

и среднее от этой величины *по итерациям*  $C_{\text{avg}}$  сходится к тому, что нам и надо:  $C_{\text{avg}} \rightarrow \langle \langle C^2 \rangle \rangle$ . Однако с  $\langle \langle C \rangle_F^2 \rangle_B$  и  $\langle \langle C \rangle_B^2 \rangle_F$  такой метод, конечно же, невозможен.

В самом деле, допустим, что  $N_F$  так велико, что для каждого луча из камеры сумма  $\frac{1}{N_F} \sum_{i=1}^{N_F} C(X_i^{(F)}, X_j^{(B)})$  дает хорошее приближение к  $\langle C \rangle_F (X_j^{(B)})$  В то же время, как правило, число лучей через один пиксель  $N_B$  невелико, и потому усреднение  $\langle C \rangle_F^2 (X_j^{(B)})$ по этому малому числу лучей недостаточно. Однако тут мы можем добавить усреднение по итерациям, так как в каждой из них у нас свой случайный и независимый набор лучей из камеры, и так по-

лучить уже хорошую оценку для  $\langle \langle C \rangle_F^2 \rangle_B$ .

Это, разумеется, лишь мысленный эксперимент, и на практике подобный метод работать не может, так как обычно количество лучей из источника тоже не столь велико, чтобы иметь достаточно точную оценку  $\langle C \rangle_F(X_j^{(B)})$  в пределах одной итерации. Использовать же тут усреднение по итерациям крайне сложно, так как эта величина должна вычисляться для каждой трассы из камеры  $X_j^{(B)}$ , а она встречается лишь однажды. Конечно, на самом деле в окрестность этого пути мы будем попадать регулярно, так что, если на каждой итерации вычислять и запоминать сеточную функцию  $X_i^{(B)}$  (на достаточно частой сетке), то все и получит-

ПРОГРАММИРОВАНИЕ № 3 2021

ся. Но это весьма неэкономично, так как размерность пространства, равная максимальной длине трассы из камеры, достаточно велика.

Решение, как ни странно, весьма просто. Пусть для удобства число лучей из камеры и из источника не зависит от итерации. Чтобы вычислить  $\langle \langle C \rangle_F^2 \rangle_B$ , разобьём все множество  $N_F$  лучей из источника на две половины (не обязательно равные), содержащие  $N_{F_1}$  и  $N_{F_2}$  лучей соответственно, и будем на каждой итерации вычислять

$$B = \frac{1}{N_B} \sum_{j=1}^{N_B} \left( \left( \frac{1}{N_{F_1}} \sum_{i \in half_1} C(X_i^{(F)}, X_j^{(B)}) \right) \times \left( \frac{1}{N_{F_2}} \sum_{i \in half_2} C(X_i^{(F)}, X_j^{(B)}) \right) \right)$$

Среднее от этой величины по итерациям  $B_{avg}$  есть не что иное, как среднее по *ансамблю*  $N_F$  лучей из источника и  $N_B$  лучей из камеры. Эти два усреднения независимы и потому перестановочны. Усредним сначала B по ансамблю лучей из источника:  $B_{avg} = \langle \langle B \rangle_F \rangle_B$ . Поскольку лучи первой и второй половины этого ансамбля очевидно независимы, то среднее от произведения двух внутренних сумм есть произведение средних по этим половинкам:

$$\langle B \rangle_F \equiv \frac{1}{N_B} \sum_{j=1}^{N_B} (\langle C \rangle_{F_1}(X_j^{(B)}) \langle C \rangle_{F_2}(X_j^{(B)})),$$

а поскольку статистические характеристики лучей первой и второй половины, конечно же, одинаковы, то средние по половинкам ансамбля и по всему ансамблю одинаковы тоже

$$\langle B \rangle_F \equiv \frac{1}{N_B} \sum_{j=1}^{N_B} \langle C \rangle_F^2(X_j^B)$$

Оставшееся усреднение по ансамблю лучей из камеры дает

$$B_{\text{avg}} = \langle \langle B \rangle_F \rangle_B = \langle \langle C \rangle_F^2 \rangle_B$$

Таким образом, второй коэффициент формулы шума может быть вычислен со сколь угодно высокой точностью по последовательности итераций, даже если в каждой из них лучей мало.

Вычисление  $\langle \langle C \rangle_B^2 \rangle_F$  организуется совершенно аналогично, разбивая на две части теперь уже набор лучей из камеры (через данный пиксель!) и вычисляя на каждой итерации:

$$F \equiv \frac{1}{N_F} \sum_{i=1}^{N_F} \left( \left( \frac{1}{N_{B_1}} \sum_{i \in half_1} C(X_i^{(F)}, X_j^{(B)}) \right) \times \left( \frac{1}{N_{B_2}} \sum_{i \in half_2} C(X_i^{(F)}, X_j^{(B)}) \right) \right)$$

Ее среднее по итерациям  $F_{\text{avg}}$  сходится к  $\langle \langle C \rangle_{R}^{2} \rangle_{F}$ .

Заметим, что разбиение набора лучей из камеры или из источника, может быть и на неравные части. Но смысла в этом нет, так как суммы по этим двум частям входят в наши формулы симметрично, и потому нет резона "улучшать" одну за счет другой. На практике проще использовать разбиение на четные и нечетные лучи.

Зная теперь квадратичные средние и предельную яркость пикселя *L*, мы можем вычислить дисперсию *вклада одной итерации* в эту яркость как

$$V = \frac{1}{N_F N_B} (C_{\text{avg}} - L^2) + \frac{1 - N_F^{-1}}{N_B} (B_{\text{avg}} - L^2) + \frac{1 - N_B^{-1}}{N_F} (F_{\text{avg}} - L^2),$$
(1)

а после  $N_I$  итераций дисперсия будет  $\frac{V}{N_I}$ .

Отметим, что  $C_{avg}$ ,  $B_{avg}$ ,  $F_{avg}$ , как и L, зависят от пикселя, но *не зависят от числа лучей*. Правда, как мы уже говорили,  $C_{avg}$ ,  $B_{avg}$ ,  $F_{avg}$  могут зависеть от радиуса интегрирующей сферы. Это очень важное свойство, так как оно позволяет предсказать шум для *произвольного числа лучей*. Таким образом, можно провести *один* предварительный расчет для как-либо выбранного, пусть и весьма неудачного, числа лучей, найти значения  $C_{avg}$ ,  $B_{avg}$ ,  $F_{avg}$ , L и затем вычислить *оптимальное* число лучей (в общем случае при этом  $N_B$  будет в каждом пикселе свое), которое обеспечивает минимально возможный шум.

# 4. ОСОБЕННОСТИ УЧЕТА ПРЯМОГО ОСВЕЩЕНИЯ

В методе BDPM прямое освещение может учитываться двумя разными способами. Во-первых, можно все лучи из источника — как первичные (не испытавшие ни одного рассеяния), так и вторичные (рассеянные), обрабатывать единообразно. То есть прямое освещение тоже берется из фотонных карт. Во-вторых, можно вычислять его непосредственно: каждую вершину пути из камеры мы соединяем с источником, вычисляем его яркость вдоль данного сегмента и складываем с оценкой яркости из фотонных карт [14] (формула (2) при  $w_0 = 0, w_1 = 1$ ). Как правило, второй способ выгоднее, так как в нем компонента яркости, связанная с прямым освещением, не содержит шума фотонных карт.

Разумеется, в обоих случаях шум также описывается формулами из раздела 3. В самом деле, если все компоненты яркости берутся из фотонных карт, как в первом способе, то и *ненулевых* членов  $C(X_i^{(F)}, X_j^{(B)})$  окажется много меньше предельного значения  $N_F N_B N_p$  (где  $N_p$  – число пикселей изображения), поскольку вероятность "пересечения" трасс мала. Поэтому вычисление сумм в C, B, F будет достаточно эффективно.

Иное дело второй способ, в котором *каждый* путь из камеры приносит какой-то вклад из-за прямого освещения. Теперь уже *все*  $C(X_i^{(F)}, X_j^{(B)})$  оказываются отличны от нуля, даже если  $X_i^{(F)}$  и  $X_j^{(B)}$  не пересекаются. Вычисление сумм в *C*, *B*, *F* становится слишком дорогим. Поэтому для второго способа выгодно его несколько преобразовать, что позволит упростить вычисления.

Для борьбы с этим эффектом необходимо в суммах обособить вклад от прямого освещения, ибо он не зависит от луча из источника, а потому может быть вынесен за суммирование:

$$C(X_i^{(F)}, X_j^{(B)}) = C^{(I)}(X_i^{(F)}, X_j^{(B)}) + C^{(0)}(X_j^{(B)}),$$

где  $C^{(I)}(X_i^{(F)}, X_j^{(B)})$  – вклад от пересечения камерного и *вторичного* лучей из источника, а  $C^{(0)}(X_j^{(B)})$  – вклад от прямого освещения, получаемого соединением узла луча от камеры с источником света.

Теперь С запишем так:

$$C = \frac{1}{N_B N_F} \sum_{j=1}^{N_B} \sum_{i=1}^{N_F} (C^{(I)}(X_i^{(F)}, X_j^{(B)}) + C^{(0)}(X_j^{(B)}))^2 =$$
  
=  $\frac{1}{N_B N_F} \sum_{j=1}^{N_B} \sum_{i=1}^{N_F} ((C^{(I)}(X_i^{(F)}, X_j^{(B)}))^2 +$   
+  $2C^{(I)}(X_i^{(F)}, X_j^{(B)})C^{(0)}(X_j^{(B)}))$   
+  $\frac{1}{N_B} \sum_{j=1}^{N_B} (C^{(0)}(X_j^{(B)}))^2$ 

Суммы теперь содержат столько же (ненулевых) членов, как и в отсутствии прямого освещения (т.е. при  $C^{(0)} = 0$ ).

Затем, выражение для В преобразуем:



Рис. 1. Виртуальная фотография модельной сцены Cornell Box.

$$B = \frac{1}{N_B} \sum_{j=1}^{N_B} \left( \left( \frac{\sum_{i \in half_1} C^{(I)}(X_i^{(F)}, X_j^{(B)})}{N_{F_1}} + C^{(0)}(X_j^{(B)}) \right) \times \left( \frac{\sum_{i \in half_2} C^{(I)}(X_i^{(F)}, X_j^{(B)})}{N_{F_2}} + C^{(0)}(X_j^{(B)}) \right) \right) = \frac{1}{N_B} \sum_{j=1}^{N_B} B_1(X_j^{(B)}) B_2(X_j^{(B)}),$$

где

$$B_k(X_j^{(B)}) \equiv C^{(0)}(X_j^{(B)}) + \frac{1}{N_{F_k}} \sum_{i \in half_k} C^{(I)}(X_i^{(F)}, X_j^{(B)})$$

Аналогично,

$$F = \frac{1}{N_F} \sum_{i=1}^{N_F} ((F_1^{(I)}(X_i^{(F)}) + F_1^{(0)})(F_2^{(I)}(X_i^{(F)}) + F_2^{(0)})) =$$
  
$$= \frac{1}{N_F} \sum_{i=1}^{N_F} ((F_1^{(I)}(X_i^{(F)})F_2^{(I)}(X_i^{(F)}) +$$
  
$$+ F_1^{(I)}(X_i^{(F)})F_2^{(0)} + F_2^{(I)}(X_i^{(F)})F_1^{(0)})) + F_1^{(0)}F_2^{(0)},$$

где

$$\begin{split} F_k^{(I)}(X_i^{(F)}) &\equiv \frac{1}{N_{B_k}} \sum_{i \in half_k} C^{(I)}(X_i^{(F)}, X_j^{(B)}) F_k^{(0)} \equiv \\ &\equiv \frac{1}{N_{B_k}} \sum_{i \in half_k} C^{(0)}(X_j^{(B)}) \end{split}$$

Теперь все суммы содержат столько же ненулевых членов, как и в отсутствии прямого освещения (т.е. при  $C^{(0)} = 0$ ), и потому вычисления достаточно экономичны. Затрачиваемое время примерно то же, что и при вычислении самой яркости изображения.

ПРОГРАММИРОВАНИЕ № 3 2021



Рис. 2. Шум как  $\sqrt{V}$  для  $N_B = 25$ ,  $N_F = 1000$ . Сверху (рис. 2а): эталонное значение, найденное как выборочная дисперсия по последовательности итераций. Снизу (рис. 2б): V вычислено по формуле (1). Усредненное по обозначенной красным квадратиком области RMS равно 1.2767 на верхнем изображении и 1.2769 на нижнем.

#### 5. РЕЗУЛЬТАТЫ

В качестве примера использовалась известная тестовая сцена Cornell Box. Изотропный точечный источник света, расположен немного ниже центра потолка. Все поверхности — ламбертовские с альбедо 0.5. Метод BDPM использовался без Multiple Importance Sampling, после второго диффузного рассеяния луч из камеры принудительно обрывался. Радиус интегрирующей сферы — 1/120 размера сцены. Прямое освещение бралось не из фотонных карт, а вычислялось непосредственным соединением точки поверхности с источником света.

Изображение, полученное виртуальной камерой, показано на рис. 1.

Три члена формулы шума были вычислены, как описано в разделах 3 и 4, и полная дисперсия вклада в яркость за одну итерацию была вычислена по формуле (1). Для сравнения она также была вычислена как выборочная дисперсия по серии итераций, что возможно, так как вклады от различных итераций независимы.



Рис. 3. Вклады в полный шум  $\sqrt{\frac{1}{N_F N_B}(C_{avg} - L^2)}$ ,  $\sqrt{\frac{1-N_F^{-1}}{N_B}(B_{avg} - L^2)}$  и  $\sqrt{\frac{1-N_B^{-1}}{N_F}(F_{avg} - L^2)}$  записаны в цветовые каналы R, G и B изображения. Сверху (Рис. 3а):  $N_B = 25$ , снизу (Рис. 36):  $N_B = 100$ ; в обоих случаях  $N_F = 1000$ . Средние значения по области, ограниченной красным квадратиком: (0.3812065, 1.21883, 0.01907) сверху и (0.19602, 0.610129, 0.0204105) снизу, так что первые две компоненты уменьшились как раз в  $\sqrt{4}$  раз. Полный RMS, т.е.  $\sqrt{V}$ , усредненный по красному квадратику, 1.27689 сверху и 0.639361 снизу.

Амплитуда шума, т.е.  $\sqrt{V}$ , вычисленная этими двумя способами, показана на рис. 2. Видно, что результаты практически неразличимы.

Для иллюстрации, который из трех членов формулы (1) дает какой вклад в полную дисперсию, все они показаны на рис. 3 в виде (R, G, B) компонент цвета изображения. А именно, вели-

чины 
$$\sqrt{\frac{1}{N_F N_B}(C_{avg} - L^2)}, \sqrt{\frac{1 - N_F^{-1}}{N_B}(B_{avg} - L^2)}$$
 и

 $\sqrt{\frac{1-N_B^{-1}}{N_F}}(F_{avg}-L^2)$  записаны в R, G и B каналы

изображения, соответственно.

Видно, что картинка на рис. 3 в основном зеленая, т.е. доминирует вторая компонента. Таким образом, в нашем примере для снижения шума



**Рис. 4.** Средний RMS по части изображения (красный квадратик на рис. 2) после 1000 с вычисления для различных  $N_B$  и  $N_F$  (по горизонтальной оси).

необходимо главным образом уменьшить этот доминирующий, второй член, т.е. увеличить  $N_B$ , так как  $C_{avg}$ ,  $B_{avg}$ ,  $F_{avg}$  и L от числа лучей не зависят. В то время как увеличение числа лучей из источника  $N_F$  почти не дает эффекта и лишь замедляет расчет. Результаты для увеличенного втрое  $N_B$  показаны на рис. Зб.

На практике значение имеет не столько дисперсия вклада одной итерации V, сколько дисперсия яркости после заданного времени расчета Т. В течение этого времени будет сделано  $N_I = \frac{T}{2}$ итераций и дисперсия полученного изображения составит  $V_T = \frac{\tau V}{T}$ . В то время как сама V монотонно убывает с ростом числа лучей, время одной итерации, напротив, монотонно возрастает. Эти два "противодействующих фактора" и обеспечивают наличие оптимума по числу лучей, как продемонстрировано на рис. 4 и в табл. 2. Они показывают поведение среднего по красному квадратику RMS (т.е.  $\sqrt{V_T}$ ) для нашей испытательной сцены в зависимости от числа лучей на 1 итерацию при T = 1000 с. Заметим, что число лучей из камеры было взято одинаковым во всех пикселях. а не только внутри красного квадрата.

Из рис. 4 хорошо видно, что зависимость от числа лучей из камеры (представленных графиками разного цвета) в данном примере очень слаба. Зависимость же от числа лучей из источника (отложенных по горизонтальной оси) имеет минимум в окрестности  $N_F = 3100$ . Точное значение минимума может слегка меняться в зависимости от числа лучей из камеры. Минимум этот, однако, весьма плоский, то есть RMS почти не меняется в широких пределах изменения  $N_F$ , и заметный его

Таблица 2. Статистика трассировки тестовой сцены в течение 1000 с при 25 лучах из камеры через пиксель в зависимости от числа лучей из источника

$N_F$	1000	3000	10000	30000	100000	300000
$\sqrt{V_T}$	0.1	0.098	0.1	0.11	0.13	0.18
$\sqrt{V}$	1.277	1.234	1.224	1.222	1.21	1.202
Ni	160	158	149	130	89	47
τ	6.25	6.33	6.7	7.7	11.25	21.47
$\tau_{BMCRT}$	6	6	6	6	6	6
$\tau_{FMCRT}$	≈0	≈0	≈0	≈0	0.05	0.17
$\tau_{merge}$	0.25	0.33	0.7	1.7	5.2	15.3

рост начинается только при отклонении от оптимума на порядки.

Необходимо пояснить, что плавная кривая по многим точкам на рис. 4 построена, вычисляя RMS за одну итерацию по *точной* формуле (1), а время на одну итерацию по аппроксимации  $\tau = \alpha N_F + \beta N_B + \gamma N_B N_F$ , коэффициенты которой получены подгонкой по расчетам из табл. 1. Аппроксимация эта работает очень хорошо.

Более детальная информация показана в табл. 2 для случая  $N_B = 25$ . Время одной итерации  $\tau$  складывается из времени, потраченного на трассировку лучей из камеры  $\tau_{BMCRT}$ , трассировку лучей из источника  $\tau_{FMCRT}$ , и времени слияния этих трасс  $\tau_{merge}$ . Помимо итогового шума  $\sqrt{V_T}$  табл. 2 показывает дисперсию по одной итерации  $\sqrt{V}$ , число итераций, сделанных за период *T*, время на одну итерацию  $\tau$  и ее составляющие.

В табл. 2 хорошо видно, что дисперсия по одной итерации уменьшается с увеличением числа лучей от источника света (что ожидаемо), в то время как итоговый шум возрастает. Также анализ времен составляющих позволяет оценить их вклад в общее время трассировки и понять, почему увеличение числа лучей ведет в конечном счете к ухудшению качества итогового изображения.

#### 6. ЗАКЛЮЧЕНИЕ

Предложен практичный метод вычисления входящих в формулу шума для BDPM коэффициентов  $\langle \langle C^2 \rangle \rangle$ ,  $\langle \langle C \rangle_F^2 \rangle_B$ ,  $\langle \langle C \rangle_B^2 \rangle_F$  на основе полученных в трассировке лучей величин. Также рассмотрен практический способ вычисления коэффициентов для случая непосредственного вычисления прямого освещения. Само итоговое значение шума может быть вычислено непосредственно по выборочной дисперсии ряда значений, накопленных в последовательных итерациях. Однако это дает только его значение для данного расчета и не позволяет предсказать, как он изменится для другого числа лучей. Между тем, это легко вычислить, зная коэффициенты  $\langle \langle C^2 \rangle \rangle$ ,  $\langle \langle C \rangle_F^2 \rangle_B$ ,  $\langle \langle C \rangle_B^2 \rangle_F$ . Таким образом, по результатам всего одного расчета можно предсказать дисперсию (вклада одной итерации) для произвольного числа лучей.

Если время, затраченное на трассировку лучей из камеры и из источника, известно вместе со временем, затраченным на их "слияние", как это приводится в табл. 2, то можно также вычислить оптимальное количество лучей или, другими словами, такое число лучей, которое дает наименьший шум в течение того же времени расчета.

#### СПИСОК ЛИТЕРАТУРЫ

- Zhdanov D., Galaktionov V., Voloboy A., Zhdanov A., Garbul A., Potemin I. and Sokolov V. Photorealistic rendering of images formed by augmented reality optical systems // Programming and Computer Software. 2018. V. 44. № 4. P. 213–224. https://doi.org/10.1134/S0361768818040126
- 2. *Sik M., Krivanek J.* Survey of Markov Chain Monte Carlo methods in light transport simulation // IEEE Transactions on Visualization and Computer Graphics. 2018. V. 26. № 4. P. 1821–1840.
- 3. *Pharr M., Humphreys G.* Physically Based Rendering.Second Edition: From Theory To Implementation. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2010.
- 4. *Dodik N.* Implementing probabilistic connections for bidirectional path tracing in the Mitsuba Renderer, Sep. 2017. https://www.cg.tuwien.ac.at/research/pub-lications/2017/dodik-2017-pcbpt/
- Jensen H.W., Christensen P. High quality rendering using ray tracing and photon mapping // ACM SIG-GRAPH 2007 Courses. Ser. SIGGRAPH '07. 2007. NY, USA: ACM.
  http://doi.org/10.1145/1281500.1281502

http://doi.acm.org/10.1145/1281500.1281593

- 6. *Veach E.* A dissertation: Robust Monte-Carlo methods for light transport simulation. 1997. http://graph-ics.stanford.edu/papers/veach\_thesis/thesis.pdf
- Vorba J. Bidirectional photon mapping / Proceedings of CESCG 2011: The 15th Central European Seminar on Computer Graphics. Prague: Charles University, 2011. P. 25–32. https://cgg.mff.cuni.cz/~jaroslav/papers/2011-bdpm/vorba2011-bdpm.pdf
- Ershov S.V., Zhdanov D.D., Voloboy A.G. Estimation of noise in calculation of scattering medium luminance by MCRT // Mathematica Montisnigri. 2019. V. XLV. P. 60–73.
- 9. Georgiev I., Krivánek J., Davidovic T., Slusallek P. Light transport simulation with vertex connection and merg-

ing // ACM Trans. Graph. 2012. V. 31. № 6. P. 192:1– 192:10. http://doi.acm.org/10.1145/2366145.2366211

- Hachisuka T., Pantaleoni J., Jensen H.W. A path space extension for robust light transport simulation // ACM Trans. Graph. 2012. V. 31. P. 191:1–191:10.
- Ershov S.V., Voloboy A.G. Calculation of MIS weights for bidirectional path tracing with photon maps in presence of direct illumination // Mathematica Montisnigri. 2020. V. XLVIII. P. 86–102.
- 12. Sbert M., Havran V., Szirmay-Kalos L. Multiple importance sampling revisited: breaking the bounds // EUR-

ASIP Journal on Advances in Signal Processing. 2018. V. 15. P. 1–15.

- Hachisuka T., Jensen H.W. Stochastic progressive photon mapping // ACM SIGGRAPH Asia 2009 Papers. 2009. P. 141:1–141:8. http://doi.acm.org/10.1145/1661412.1618487
- 14. Ершов С.В., Бирюков Е.Д., Волобой А.Г. Эффективное вычисление оптимальных весов множественной выборки по значимости в двунаправленной трассировке лучей с фотонными картами // Препринты ИПМ им. М.В. Келдыша. 2020. № 107. 22 с. https://doi.org/10.20948/prepr-2020-107