Метод Монте-Карло
Monte Carlo Method

Быковских Дмитрий Александрович

09.12.2023

2023-12-13
ММК

Метод Монте-Карло
Monte Carlo Method

Быковских Дмитрий Александрович

09.12.2023

Содержание

- Определение, понятия, история
- Генератор псевдослучайных чисел
- Пример
- Квази МКМ
- Скорость сходимости
- Применение метода в компьютерной графике
- MISER algorithm

└Содержание

Содержание

- Определение, понятия, история
- Генератор псевдослучайных чисел
- Пример
- Квази МКМ
- Скорость сходимости
- Применение метода в компьютерной графике
- MISER algorithm

Краткая справка
Monte-Carlo Method (MCM)

Монте-Карло метод (ММК) — численный метод решения математических задач при помощи моделирования случайных величин.
Название происходит от города Монте-Карло в княжестве Монако, знаменитого своим игорным домом.
Датой рождения: 1949 г., когда появилась статья под названием The Monte Carlo Method. Создатели: Дж. Нейман и С. Улам
Теоретическая основа известна давно.

Краткая справка
Monte-Carlo Method (MCM)
Монте-Карло метод (ММК) — численный метод решения математических задач при помощи моделирования случайных величин.
Название происходит от города Монте-Карло в княжестве Монако, знаменитого своим игорным домом.
Датой рождения: 1949 г., когда появилась статья под названием The Monte Carlo Method. Создатели: Дж. Нейман и С. Улам
Теоретическая основа известна давно.

Рулетка — простейший прибор получения случайной величины.
Метод Монте-Карло — [84; 85].
Первая работа по использованию вероятностного метода была опубликована А. Холлом [86] в 1873 году именно при организации стохастического процесса экспериментального определения числа π путем бросания иглы на лист линованной бумаги. Идея такого эксперимента возникла у Ж.Л.Л. Бюффона для вычисления числа π в 1777 году.
Бурное развитие и применение методов статистического моделирования (Монте-Карло) в различных областях прикладной математики началось с середины прошлого столетия. Это было связано с решением качественно новых задач, возникших при исследовании новых процессов. Одним из первых, кто применил ММК для моделирования траекторий нейтронов был Дж. фон Нейман. Первая работа с систематическим изложением была опубликована в 1949 году Н.К. Метрополисом и С.М. Уламом [87]. Метод Монте-Карло применялся для решения линейных интегральных уравнений, в котором решалась задача о прохождении нейтронов через вещество.

Краткая справка

Особенности метода:

1. Простая структура вычислительного алгоритма, т.е. необходимо описать действие одного шага. А потом множество шагов усреднить. Метод статистических испытаний.
2. Ошибка вычислений, как правило, пропорциональная

$$\sqrt{\frac{D}{N}},$$

   где D — некоторая постоянная, а N — число испытаний. Метод эффективен, когда высокая точность не сильно важна.

Задачи решаемые с помощью ММК:

1. Любой процесс, на протекание которого влияют случайные факторы.
2. Для любой задачи можно искусственно придумать вероятностную модель или даже несколько.

Краткая справка

Особенности метода:

1. Простая структура вычислительного алгоритма, т.е. необходимо описать действие одного шага. А потом множество шагов усреднить. Метод статистических испытаний.
2. Ошибка вычислений, как правило, пропорциональная

$$\sqrt{\frac{D}{N}},$$

   где D — некоторая постоянная, а N — число испытаний. Метод эффективен, когда высокая точность не сильно важна.

Задачи решаемые с помощью ММК:

1. Любой процесс, на протекание которого влияют случайные факторы.
2. Для любой задачи можно искусственно придумать вероятностную модель или даже несколько.

Случайные числа

Случайная величина - это математический объект, которая представляет собой функцию, сопоставляющая каждому элементарному исходу случайного эксперимента числовое значение. Функция распределения случайной величины (ФРСВ) предоставляет информацию о том, как вероятность распределена между различными значениями случайной величины. В теории вероятностей часто используются различные распределения (например, равномерное, биномиальное, нормальное), чтобы моделировать различные типы случайных величин.
Генератор псевдослучайных чисел (ПСЧ) - это алгоритм или устройство, создающее последовательность чисел, которые кажутся случайными, но на самом деле образуют детерминированную последовательность. Эти числа обычно используются в компьютерных программах для имитации случайности в различных приложениях, таких как моделирование, шифрование, игры и другие.

Случайные числа

Случайная величина - это математический объект, которая представляет собой функцию, сопоставляющая каждому элементарному исходу случайного эксперимента числовое значение. Функция распределения случайной величины (ФРСВ) предоставляет информацию о том, как вероятность распределена между различными значениями случайной величины. В теории вероятностей часто используются различные распределения (например, равномерное, биномиальное, нормальное), чтобы моделировать различные типы случайных величин.
Генератор псевдослучайных чисел (ПСЧ) - это алгоритм или устройство, создающее последовательность чисел, которые кажутся случайными, но на самом деле образуют детерминированную последовательность. Эти числа обычно используются в компьютерных программах для имитации случайности в различных приложениях, таких как моделирование, шифрование, игры и другие.

Случайные величины могут быть классифицированы как

1. Дискретная случайная величина: Принимает конечное или счетное множество значений. Вероятности каждого значения определены явно.

2. Непрерывная случайная величина: Принимает значения из непрерывного интервала. Вероятность любого конкретного значения равна нулю (в отличие от дискретных случайных величин). Определяется функцией плотности вероятности (например, нормальное распределение).

Случайные величины также могут быть классифицированы как одномерные (включающие только одну переменную) или многомерные (включающие несколько переменных). Многомерные случайные величины используются, например, в многомерных статистиках и теории случайных процессов.

Вычисление площади определенного интеграла
Пример

Постановка задачи:

$$I = \int_a^b f(x)dx$$

Метод центральных прямоугольников

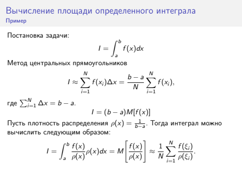
$$I \approx \sum_{i=1}^{N} f(x_i)\Delta x = \frac{b-a}{N}\sum_{i=1}^{N} f(x_i),$$

где $\sum_{i=1}^{N} \Delta x = b - a$.

$$I = (b-a)M[f(x)]$$

Пусть плотность распределения $\rho(x) = \frac{1}{b-a}$. Тогда интеграл можно вычислить следующим образом:

$$I = \int_a^b \frac{f(x)}{\rho(x)}\rho(x)dx = M\left[\frac{f(x)}{\rho(x)}\right] \approx \frac{1}{N}\sum_{i=1}^{N}\frac{f(\xi_i)}{\rho(\xi_i)}.$$



Estimator — правило для вычисления статистической оценки, определяющее скорость сходимости.

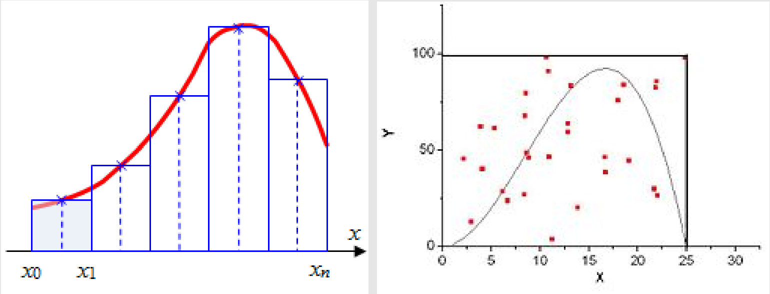


Рис. 1: Метод центральных прямоугольников (слева) и метод Монте-Карло (справа)

Оценка погрешностей при моделировании независимых испытаний

Пусть $\xi = \xi(\omega)$ — интегрируемая случайная величина. Тогда математическое ожидание $M\xi$ и дисперсия $D\xi$ случайной величины ξ определяется формулами:

$$M\xi = \int\limits_{\Omega} \xi(\omega) P(\mu(d\omega)) = \int\limits_{R_1} x dF_{\xi}(x),$$

где $P(\mu(d\omega))$ — вероятность того, что величина ω определена на интервале $d\omega$, $F_{\xi}(x) = \int\limits_{-\infty}^{x} f_{\xi}(t)\mu(dt)$ — функция распределения случайной величины ξ в точке x.

$$D\xi = \int\limits_{R_1} [x - M\xi]^2 dF_{\xi}(x) = M\xi^2 - (M\xi)^2$$

Оценка погрешностей при моделировании независимых испытаний

Пусть $\xi = \xi(\omega)$ — интегрируемая случайная величина. Тогда математическое ожидание $M\xi$ и дисперсия $D\xi$ случайной величины ξ определяется формулами:

$$M\xi = \int\limits_{\Omega} \xi(\omega) P(\mu(d\omega)) = \int\limits_{R_1} x dF_{\xi}(x),$$

где $P(\mu(d\omega))$ — вероятность того, что величина ω определена на интервале $d\omega$, $F_{\xi}(x) = \int\limits_{-\infty}^{x} f_{\xi}(t)\mu(dt)$ — функция распределения случайной величины ξ в точке x.

$$D\xi = \int\limits_{R_1} [x - M\xi]^2 dF_{\xi}(x) = M\xi^2 - (M\xi)^2$$

Оценка погрешностей при моделировании независимых испытаний

Доверительный интервал $(M\xi - \delta, M\xi + \delta)$, в котором находится истинное значение $\tilde{\xi}$ случайной величины ξ с заданной вероятностью P, определяется следующим образом:

$$P\left\{\left|\frac{1}{n}\sum_{i=1}^{n}\xi_i - \tilde{\xi}\right| \leqslant \frac{\alpha\sigma}{\sqrt{n}}\right\} = 2\Phi(\alpha),$$

где ξ_i — неизвестная величина, полученная в результате i-го испытания; n — независимые истории (число испытаний); $\sigma = \sqrt{D\xi}$ — среднеквадратичное отклонение; $\Phi(\alpha) = \frac{1}{2\sqrt{\pi}}\int\limits_0^{\alpha} e^{-t^2}dt$ — функция Лапласа.

Правило трех сигм.
Доверительный интервал $(\tilde{\xi} - 3\sigma, \tilde{\xi} + 3\sigma)$, в котором находится истинное значение μ случайной величины ξ, распределенной по нормальному закону, с заданной вероятностью P, определяется следующим образом

$$P\left\{\left|\frac{1}{n}\sum_{i=1}^{n}\xi_i - \tilde{\xi}\right| \leqslant \frac{3\sigma}{\sqrt{n}}\right\} \approx 0.9973,$$

Физически-корректный рендеринг (Physically Based Rendering, PBR) - это подход к генерации изображений, который стремится моделировать физические свойства света и материалов для достижения более реалистичных результатов.

$$L_o(p, \omega_0) = \int_\Omega \left(k_d \frac{c}{\pi} + k_s \frac{DFG}{4(\omega_0 \cdot n)(\omega_i \cdot n)} \right) L_i(p, \omega_i)(n \cdot \omega_i) d\omega_i$$

Физически-корректный рендеринг
Physically-Based Rendering
Физически-корректный рендеринг (Physically Based Rendering, PBR) - это подход к генерации изображений, который стремится моделировать физические свойства света и материалов для достижения более реалистичных результатов.
$$L_o(p, \omega_0) = \int_\Omega \left(k_d \frac{c}{\pi} + k_s \frac{DFG}{4(\omega_0 \cdot n)(\omega_i \cdot n)} \right) L_i(p, \omega_i)(n \cdot \omega_i) d\omega_i$$

Основные принципы физически-корректного рендеринга включают в себя:

- Моделирование света: Учет различных источников света, их цветовых температур, направления и интенсивности. Учет закона сохранения энергии в процессе отражения и преломления света.
- Моделирование материалов: Физически-корректные модели отражения, преломления и поглощения света в зависимости от типа материала.
- Моделирование теней и окружающей среды: Учет влияния теней от различных объектов и источников света. Моделирование окружающей среды для более точного воссоздания условий освещения.
- Многоканальные текстуры: Использование текстур с несколькими каналами для более точного представления различных характеристик материалов, таких как шероховатость, металлические свойства и др.
- Моделирование камеры: Учет характеристик камеры, таких как диафрагма и выдержка, для более реалистичного эффекта глубины резкости и затемнения по краям кадра.

Физически-корректный рендеринг
Physically-Based Rendering

Формулы энергетической яркости поверхности в конкретном направлении

$$L = \frac{d^2\Phi}{\cos\theta d\omega dA}$$

$$L\cos\theta d\omega = \frac{d^2\Phi}{dA}$$

где L — энергетическая яркость (Radiance), описывает количество светового потока, излучаемого поверхностью в определенном направлении, на единичную площадку и в единичный угловой диапазон $(W/(m \cdot sr))$; $d^2\Phi$ — элемент светового потока (Flux) через малую площадку dA в малом угловом диапазоне $d\omega$; $\cos\theta$ — косинус угла между нормалью к поверхности и направлением, в котором измеряется энергетическая яркость; dA — элемент площади поверхности, через которую измеряется световой поток; $d\omega$ — элемент углового диапазона, в пределах которого измеряется световой поток.

Физически-корректный рендеринг
Physically-Based Rendering
Формулы энергетической яркости поверхности в конкретном направлении

$$L = \frac{d^2\Phi}{\cos\theta d\omega dA}$$

$$L\cos\theta d\omega = \frac{d^2\Phi}{dA}$$

где L — энергетическая яркость (Radiance), описывает количество светового потока, излучаемого поверхностью в определенном направлении, на единичную площадку и в единичный угловой диапазон $(W/(m \cdot sr))$; $d^2\Phi$ — элемент светового потока (Flux) через малую площадку dA в малом угловом диапазоне $d\omega$; $\cos\theta$ — косинус угла между нормалью к поверхности и направлением, в котором измеряется энергетическая яркость; dA — элемент площади поверхности, через которую измеряется световой поток; $d\omega$ — элемент углового диапазона, в пределах которого измеряется световой поток.

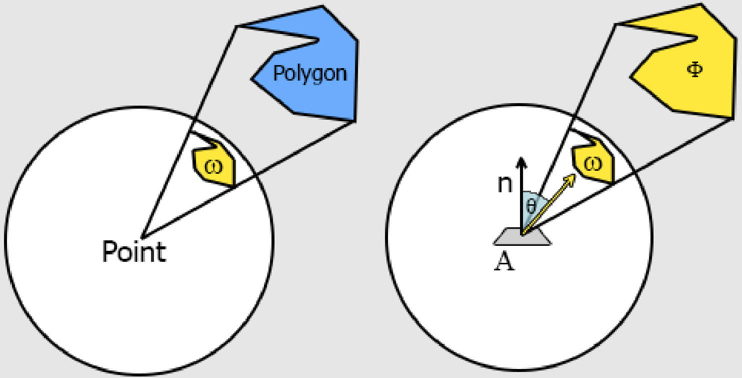


Рис. 2: Схема расчета телесного угла и энергетической яркости

Физически-корректный рендеринг

Physically-Based Rendering

Первая формула представляет собой уравнение отраженной энергетической яркости L_o в точке p на поверхности, ориентированной в направлении с углами ϕ_o и θ_o.

$$L_o(p, \phi_o, \theta_o) = k_d \frac{c}{\pi} \int_{\phi=0}^{2\pi} \int_{\theta=0}^{\pi/2} L_i(p, \phi_i, \theta_i) \cos(\theta) \sin(\theta) d\phi d\theta$$

Примечание. Расчет для полусферы.
Дискретная форма формулы отраженной энергетической яркости

$$L_o(p, \phi_o, \theta_o) \approx k_d \frac{c\pi}{N_1 N_2} \sum_{\phi=0}^{N_1} \sum_{\theta=0}^{N_2} L_i(p, \phi_i, \theta_i) \cos(\theta) \sin(\theta) d\phi d\theta$$

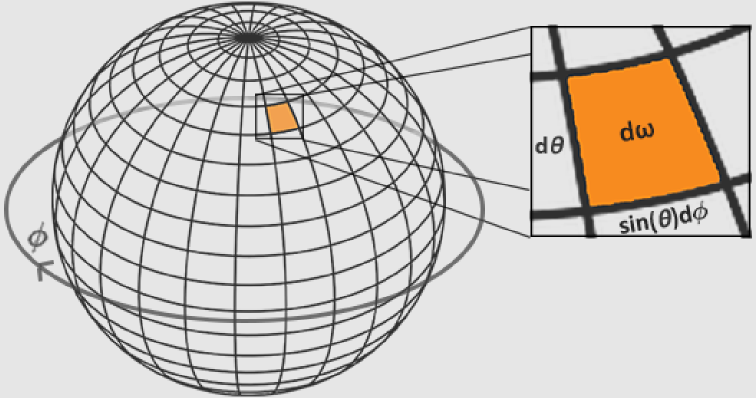


Рис. 3: Иллюстрация к интегрированию в сферической системе координат

Физически-корректный рендеринг

Physically-Based Rendering

Интегральная форма

$$L_o(p,\omega_o) = k_d\frac{c}{\pi}\int_{\Omega} L_i(p,\omega_i)(n\cdot\omega_i)d\omega_i$$

Дискретная форма

$$L_o(p,\omega_o) \approx k_d\frac{c}{\pi}\frac{1}{N}\sum_{i=1}^{N} L_i(p,\omega_i)(n\cdot\omega_i),$$

где w_i — входящий вектор направления рассчитывается случайным образом с некоторым законом распределения.

Физически-корректный рендеринг
Physically-Based Rendering
Интегральная форма
$$L_o(p,\omega_o) = k_d\frac{c}{\pi}\int_{\Omega} L_i(p,\omega_i)(n\cdot\omega_i)d\omega_i$$
Дискретная форма
$$L_o(p,\omega_o) \approx k_d\frac{c}{\pi}\frac{1}{N}\sum_{i=1}^{N} L_i(p,\omega_i)(n\cdot\omega_i),$$
где w_i — входящий вектор направления рассчитывается случайным образом с некоторым законом распределения.

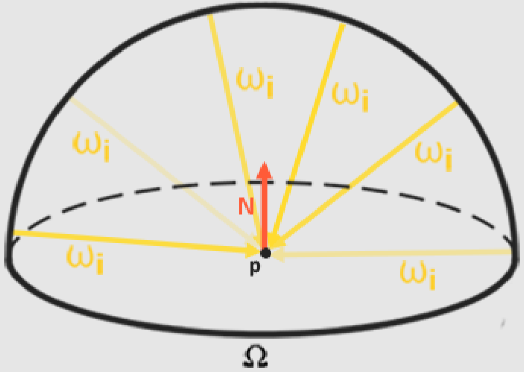


Рис. 4: Иллюстрация усредненной суммы энергетической яркости внутри полусферы

Квази-метод Монте-Карло

Quasi-Monte Carlo Method

Квази-метод Монте-Карло представляет собой вариацию метода Монте-Карло, который использует квазислучайные последовательности вместо полностью случайных чисел. В отличие от стандартных случайных чисел, квазислучайные последовательности обладают определенными детерминированными свойствами, направленными на создание последовательности чисел, которые стремятся к равномерному покрытию пространства.
Основная идея квази-метода Монте-Карло состоит в том, чтобы заменить случайные числа последовательностью чисел с некоторыми хорошими свойствами равномерного распределения. Эти последовательности разрабатываются так, чтобы минимизировать дисперсию оценок интегралов, т.е. улучшить скорость сходимости метода, и обеспечивать равномерное покрытие пространства.

Квази-метод Монте-Карло
Quasi-Monte Carlo Method
Квази-метод Монте-Карло представляет собой вариацию метода Монте-Карло, который использует квазислучайные последовательности вместо полностью случайных чисел. В отличие от стандартных случайных чисел, квазислучайные последовательности обладают определенными детерминированными свойствами, направленными на создание последовательности чисел, которые стремятся к равномерному покрытию пространства.
Основная идея квази-метода Монте-Карло состоит в том, чтобы заменить случайные числа последовательностью чисел с некоторыми хорошими свойствами равномерного распределения. Эти последовательности разрабатываются так, чтобы минимизировать дисперсию оценок интегралов, т.е. улучшить скорость сходимости метода, и обеспечивать равномерное покрытие пространства.

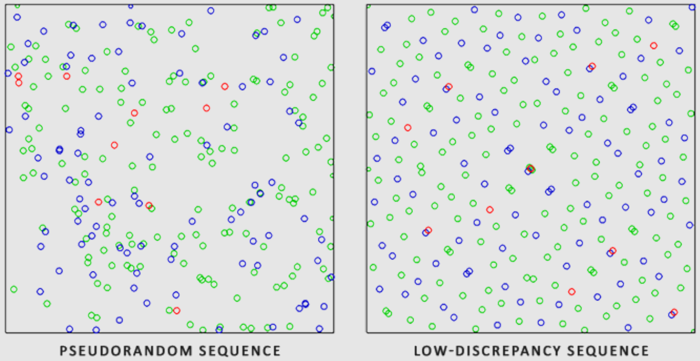


Рис. 5: Результаты моделирования: псевдослучайная последовательность (слева) и последовательность низкого несоответствия (справа)

MISER algorithm

MISER algorithm позволяет адаптивно распределять вычислительные ресурсы в зависимости от сложности функции на различных участках интервала интегрирования. Это улучшает эффективность численного интегрирования, особенно для функций с различной степенью изменчивости.

Описание метода

1. Задание начального интервала интегрирования.
2. Разбиение текущего интервала на более мелкие подинтервалы.
3. Вычисление приближенных значений на каждом подинтервале.
4. Оценка погрешности.
5. Адаптация разбиения за счет увеличения количества подинтервалов в областях с большой погрешностью.
6. Повторение процесса (шагов 3-5) до достижения заданной точности.

MISER algorithm

MISER algorithm позволяет адаптивно распределять вычислительные ресурсы в зависимости от сложности функции на различных участках интервала интегрирования. Это улучшает эффективность численного интегрирования, особенно для функций с различной степенью изменчивости.

Описание метода

1. Задание начального интервала интегрирования.
2. Разбиение текущего интервала на более мелкие подинтервалы.
3. Вычисление приближенных значений на каждом подинтервале.
4. Оценка погрешности.
5. Адаптация разбиения за счет увеличения количества подинтервалов в областях с большой погрешностью.
6. Повторение процесса (шагов 3-5) до достижения заданной точности.

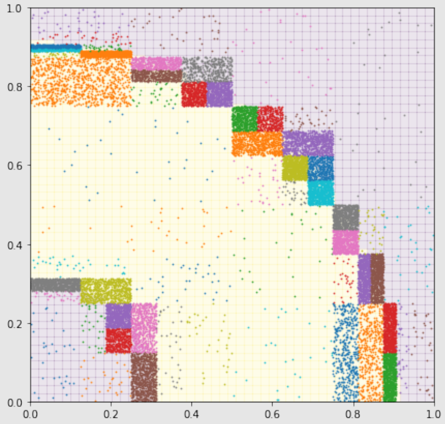


Рис. 6: Результат моделирования методом MISER при вычислении интеграла

Заключение

Литература

1. Соболь И.М. Метод Монте-Карло (Популярные лекциии по математике, выпуск 46)
2. Учебное пособие по курсу "Численные методы в оптике"
3. Coding Labs — Physically Based Rendering
4. Maxwell rules — Monte Carlo Integration
5. Learn OpenGL. Урок 6.3 — Image Based Lighting. Диффузная облученность

2023-12-13

ММК

└Заключение

Заключение

Литература

1. Соболь И.М. Метод Монте-Карло (Популярные лекциии по математике, выпуск 46)
2. Учебное пособие по курсу "Численные методы в оптике"
3. Coding Labs — Physically Based Rendering
4. Maxwell rules — Monte Carlo Integration
5. Learn OpenGL. Урок 6.3 — Image Based Lighting. Диффузная облученность