УДК 004.272:004.92

О. Н. Романюк, д-р. техн. наук, професор
О. Л. Бобко, аспірант
[1,2]Вінницький національний технічний університет, Вінниця, Україна
[1]rom8591@gmail.com
[2]oleksii.bobko@gmail.com

# МЕТОД БАЛАНСУВАННЯ ОБЧИСЛЮВАЛЬНОГО НАВАНТАЖЕННЯ В БАГАТОПОТОКОВИХ ПРОГРАМНИХ СИСТЕМАХ ТРИВИМІРНОГО РЕНДЕРИНГУ

*У роботі розглянуто питання підвищення продуктивності фотореалістичного рендерингу за рахунок більш рівномірного розподілу обчислень між паралельними потоками. Основну увагу приділено функції двонапрямного розподілу відбиття світла BRDF, обчислення якої для різних матеріалів має неоднакову складність. Для матових поверхонь BRDF змінюється відносно плавно, тоді як для дзеркальних, металевих і шорстких матеріалів характер зміни функції може бути значно складнішим. Через це традиційний розподіл задач за кількістю пікселів, променів або тайлів не завжди забезпечує рівномірне завантаження CPU/GPU. Запропоновано метод адаптивного багаторівневого розпаралелювання, у якому складність обчислення BRDF оцінюється за локальною кривиною функції відбиття. Для цього використовуються градієнт, матриця Гессе та її власні значення. На основі отриманої оцінки ділянки параметричного простору поділяються на гладкі, середньої складності та складні. Для кожної ділянки формується вагова характеристика, яка враховується під час розподілу задач між потоками. Це дозволяє розподіляти навантаження не лише за кількістю задач, а й за їх реальною трудомісткістю. Метод передбачає динамічне балансування навантаження. Якщо один із потоків перевантажений, частина його задач передається менш завантаженому потоку. Додатково може застосовуватися кешування та локальна апроксимація BRDF для близьких параметрів освітлення та спостереження. Для оцінювання ефективності використовуються час рендерингу, прискорення, ефективність використання потоків, коефіцієнт використання ресурсів і коефіцієнт дисбалансу. Для перевірки методу передбачено сцени Cornell Box, Sponza, Bistro, San Miguel і синтетичну BRDF-сцену. Попередні оцінки показують можливість зменшення часу рендерингу приблизно на 14–28 % для сцен із неоднорідними матеріалами.*

***Ключові слова: BRDF, рендеринг, розпаралелювання, GPU, балансування навантаження, рендеринг, обчислювальний процес, програмна оптимізація, паралельне програмування.***

***DOI: 10.31474/1996-1588-2026-2-43-59-67***

## *Вступ*

Фотореалістичний синтез зображень є одним із найресурсомісткіших напрямів комп'ютерної графіки. Зростання деталізації сцен, кількості джерел світла, складності матеріалів і вимог до фізичної коректності освітлення призводить до істотного збільшення обсягу обчислень. У системах трасування променів і гібридного рендерингу значна частина часу витрачається на обчислення функції двонапрямного розподілу відбиття світла – BRDF. Функція BRDF визначає, яку частину падаючого світлового потоку поверхня відбиває у заданому напрямку спостереження. Для простих дифузних матеріалів обчислення є відносно дешевими. Для мікрофасетних моделей, зокрема Cook-Torrance або GGX, необхідні обчислення нормалізованих векторів, скалярних добутків, функцій розподілу мікрофасет, геометричних множників і коефіцієнтів Френеля.

У більшості практичних реалізацій паралелізація рендерингу виконується за просторовим принципом: зображення розбивають на тайли, сцену – на групи примітивів, а промені – на незалежні потоки виконання. Недоліком такого підходу є те, що він не враховує неоднорідність матеріалів. Тайли однакового розміру можуть містити різну кількість дзеркальних, шорстких, анізотропних або прозорих поверхонь, тому час їх обробки істотно відрізняється.

Метою роботи є розроблення методу розпаралелювання, у якому одиницею планування виступає не геометричний фрагмент сцени, а ділянка параметричного простору BRDF, оцінена за локальною складністю функції відбиття. Це дозволяє сформувати розподіл задач, ближчий до реального обчислювального навантаження.

### *Аналіз літератури*

У загальному випадку BRDF можна подати як багатовимірну функцію параметрів освітлення, спостереження та матеріалу

$$f_r = f(\omega_i, \omega_o, \lambda, \alpha, \beta, \gamma),$$

де $f_r$ – значення BRDF; $\omega_i$ – напрямок падаючого світла; $\omega_o$ – напрямок спостереження; $\lambda$ – довжина хвилі або колірний канал; $\alpha$ – шорсткість поверхні; $\beta$ – параметр металевості або дзеркальності; $\gamma$ – додаткові параметри матеріалу, наприклад анізотропія, індекс заломлення або параметри мікрогеометрії

Кожен виклик BRDF відповідає точці цього простору. Якщо точки мають близькі параметри, то їх функціональна поведінка та очікувана обчислювальна складність також є близькими.

Фізично коректний рендеринг тривимірних сцен потребує значних обчислювальних ресурсів, оскільки передбачає моделювання складної взаємодії світла з поверхнями об'єктів. Однією з ключових складових такого процесу є функція двонапрямного розподілу відбиття світла BRDF, яка визначає інтенсивність відбитого світла залежно від напрямків освітлення, спостереження та параметрів матеріалу. Теоретичні основи фізично коректного рендерингу, трасування променів і використання BRDF систематизовано в [1]. Показано, що підвищення фізичної достовірності моделей відбиття супроводжується зростанням обсягу обчислень, особливо для складних мікрофасетних моделей.

Питання підвищення продуктивності формування графічних зображень на основі вдосконалення функцій відбиття розглянуто в [2]. Запропоновані підходи до комбінованого використання BRDF дають змогу забезпечити компроміс між якістю візуалізації та обчислювальними витратами. Водночас у таких підходах недостатньо враховується нерівномірність складності обчислення функції відбиття в різних ділянках її параметричного простору.

Методи кількісного оцінювання ефективності паралельного рендерингу розглянуто в [3]. Для аналізу продуктивності використовуються показники прискорення, ефективності та завантаження обчислювальних ресурсів. Такі характеристики дають змогу оцінити якість розпаралелювання, однак вони переважно відображають результат розподілу задач і безпосередньо не визначають спосіб його формування з урахуванням складності BRDF. Теоретичні положення щодо формування тривимірних зображень, моделей освітлення, відбиття та зафарбовування наведено також у [4]. З них випливає, що однакова кількість геометричних примітивів або пікселів не завжди відповідає однаковому обсягу обчислювальних операцій.

Значний розвиток фізично коректних моделей відбиття пов'язаний із використанням мікрофасетного підходу. У [5] досліджено моделювання відбиття та заломлення світла на шорстких поверхнях. Такі моделі враховують статистичний розподіл мікрофасет і характеризуються підвищеною математичною складністю. У класичній моделі [6] відбиття визначається через функцію розподілу мікрофасет, геометричний множник та коефіцієнт Френеля. Це забезпечує високу реалістичність, однак збільшує кількість обчислень, особливо в областях, близьких до дзеркального відбиття. У [7] детально розглянуто функції маскування та затінення в мікрофасетних BRDF. Показано, що поведінка функції відбиття істотно залежить від напрямків освітлення й спостереження, а її локальна зміна може бути суттєво нерівномірною.

Практична реалізація високопродуктивного рендерингу значною мірою базується на GPU-обчисленнях. Сучасні підходи до обробки складних задач освітлення та великої кількості джерел світла наведено в [8]. Паралельна архітектура GPU забезпечує значне прискорення, однак її ефективність безпосередньо залежить від рівномірного розподілу навантаження між потоками.

Водночас недостатньо дослідженим залишається питання адаптивного розподілу обчислень з урахуванням локальної складності BRDF, що може призводити до нерівномірного завантаження потоків і зниження ефективності використання обчислювальних ресурсів.

Метою роботи є підвищення продуктивності фотореалістичного рендерингу шляхом розроблення методу адаптивного багаторівневого розпаралелювання обчислення BRDF на основі оцінювання локальної кривини функції відбиття, що дає змогу враховувати реальну обчислювальну складність окремих ділянках параметричного простору та забезпечувати більш рівномірний розподіл навантаження між потоками CPU/GPU.

### *Постановка задачі*

Нехай процес рендерингу складається з великої кількості незалежних звернень до BRDF. Кожне звернення визначається напрямком падаючого світла, напрямком до спостерігача та параметрами матеріалу. Традиційна постановка задачі розподіляє ці звернення між потоками за кількістю, але не за складністю. Запропоновано знайти такий розподіл звернень, за якого час виконання потоків буде максимально вирівняним.

Загальний час виконання паралельного етапу визначається як $T = \max\limits_{i=1,=,\dots,m}(T_i)$, де $T_i$ – час роботи $i$-го потоку, $m$ – кількість потоків.

Максимальний час визначає завершення всього паралельного етапу. Тому навіть один перевантажений потік знижує загальну продуктивність:

$$K_u = \frac{max(T_i)}{\frac{1}{m}\sum_{i=1}^{m} T_i},$$

де $K_u$ – коефіцієнт нерівномірності навантаження. При ідеальному балансуванні $K_u$ наближається до 1. Чим більше значення $K_u$, тим більша частина потоків простоює в очікуванні завершення найтривалішого потоку.

Задача полягає в тому, щоб побудувати критерій складності BRDF та використати його для адаптивного розбиття множини звернень до функції відбиття. На відміну від статичного розподілу, запропонований метод має враховувати зміну складності під час виконання та підтримувати динамічне балансування.

### *Метод балансування обчислювального навантаження в багатопотокових програмних системах рендерингу*

Першим рівнем оцінювання складності є норма градієнта функції BRDF. Вона показує, наскільки швидко змінюється значення функції у параметричному просторі:

$$C(x) = \left|\left|\nabla f_r(x)\right|\right|,$$

де $x$ – вектор параметрів BRDF; $C(x)$ – локальна складність у точці $x$. Велике значення $C(x)$ означає різку зміну функції, що може потребувати більшої кількості операцій, точнішого семплювання або меншої апроксимації.

$$x = (\omega_i, \omega_o, \lambda, \alpha, \beta, \gamma),$$

де $x$ – координати точки у параметричному просторі BRDF. Частинні похідні характеризують чутливість BRDF до зміни окремих параметрів матеріалу та умов освітлення. Їх значення показують швидкість локальної зміни функції відбиття в заданому напрямку параметричного простору. У програмній реалізації похідні можуть визначатися аналітично або чисельно, зокрема методом скінченних різниць. Чисельний підхід є універсальнішим для складних BRDF-моделей і дає змогу оцінювати локальну зміну функції без виведення окремих аналітичних залежностей.

Частинні похідні характеризують чутливість BRDF до зміни відповідного параметра:

$$\left\|\nabla f_r(x)\right\| = \sqrt{\left(\frac{\partial f_r}{\partial \omega_i}\right)^2 + \left(\frac{\partial f_r}{\partial \omega_o}\right)^2 + \left(\frac{\partial f_r}{\partial \alpha}\right)^2 + \left(\frac{\partial f_r}{\partial \beta}\right)^2 + \left(\frac{\partial f_r}{\partial \gamma}\right)^2}.$$

У практичній реалізації похідні можуть обчислюватися аналітично, чисельно або наближено через скінченні різниці.

Проте сам градієнт не завжди достатній для оцінювання складності. Якщо функція має сталий нахил, її обчислення та апроксимація залишаються передбачуваними. Значно важливішою є швидкість зміни градієнта, тобто локальна кривина функції:

$$K_{BDRF} = \left|\left|\nabla^2 f_r\right|\right|,$$

де $K_{BDRF}$ – коефіцієнт локальної кривини BRDF; $\nabla^2 f_r$ – друга похідна або оператор Гессе для багатовимірної функції. Цей коефіцієнт є основним критерієм запропонованого методу:

$$H = \left[\frac{\partial^2 f_r}{\partial x_i \partial x_j}\right],$$

де $H$ – матриця Гессе, елементи якої є другими частинними похідними BRDF за параметрами $x_i$ і $x_j$.

$$K_{BDRF} = max(|\lambda_1|, |\lambda_2|, \ldots, |\lambda_n|),$$

де $\lambda_i$ – спектр матриці Гессе. Максимальне за модулем спектру матриці характеризує найбільшу локальну кривину функції у певному напрямку параметричного простору.$K_{BDRF} < K_1$ – це умова визначає гладку ділянку BRDF, яку можна обробляти великими групами або з активним кешуванням.$K_1 \le K_{BDRF} < K_2$ – це умова визначає ділянку середньої складності, для якої доцільно використовувати стандартне розбиття.

Умова $K_{BDRF} \ge K_2$визначає складну ділянку BRDF, яку потрібно деталізувати та розподіляти між більшою кількістю потоків.

Простір гладких ділянок:

$$P = P_s \cup P_m \cup P_c,$$

де $P_m$ – ділянок середньої складності, $P_c$ – ділянок високої складності. Об'єднання цих підпросторів утворює повний простір параметрів BRDF.

Під час рендерингу комп'ютер багаторазово визначає BRDF, тобто встановлює, як світло відбивається від поверхні залежно від матеріалу, напрямку світла та положення спостерігача. Важливо зазначити, що не всі такі обчислення мають однакову складність. Для матової поверхні BRDF змінюється плавно, а для блискучої або дзеркальної – дуже різко. Тому одна частина сцени може обчислюватися швидко, а інша – значно довше.

У звичайному паралельному рендерингу роботу часто ділять між потоками приблизно порівну: за кількістю пікселів, променів або фрагментів сцени. Але це не означає, що всі потоки отримають однакове навантаження. Один потік може отримати прості матові поверхні й швидко завершити роботу, а інший – складні блискучі матеріали й працювати значно довше. У результаті частина процесорних ресурсів простоює, тобто не досягається збалансованості.

Запропонований метод спочатку оцінює складність BRDF. Для цього аналізується, наскільки швидко змінюється функція відбиття. Якщо вона змінюється плавно, така ділянка вважається простою. Якщо зміни різкі – складною. У роботі для цього використовується локальна кривина BRDF, яка визначається за матрицею Гессе та її власними значеннями.

Після цього ділянки BRDF поділяються на групи за складністю. Прості ділянки можна обробляти більшими ділянками, а складні – ділити на менші частини та розподіляти між більшою кількістю потоків. Кожній групі фактично надається своя «вага», яка показує, скільки обчислювальної роботи вона потребує.

Далі задачі розподіляються між потоками так, щоб сумарне навантаження на них було приблизно однаковим. Якщо під час роботи виявляється, що один потік перевантажений, частина його задач передається менш завантаженому потоку. Тобто система не лише один раз розподіляє роботу, а постійно підлаштовується під реальне навантаження.

Головна ідея методу: не ділити роботу між потоками лише за кількістю задач, а спочатку оцінювати складність кожного обчислення BRDF і вже відповідно до цієї складності розподіляти навантаження. Саме це має зменшити простої потоків і підвищити продуктивність паралельного рендерингу.

Запропонований метод складається з попереднього аналізу параметрів BRDF, побудови карти локальної кривини, кластеризації ділянок параметричного простору, вагового розподілу задач між потоками та динамічного перерозподілу навантаження під час виконання. Основна відмінність методу полягає у тому, що розподіл відбувається не за кількістю пікселів або променів, а за оцінкою реального обчислювального навантаження.

Вага $W_i$ і-ї ділянки або кластера дорівнює

$$W_i = N_i * K_{BRDF,i},$$

де $N_i$ – кількість звернень до BRDF у цій ділянці середній коефіцієнт кривини для ділянки.

Вага є оцінкою реального навантаження. Кількість потоків або частка обчислювального ресурсу, що виділяється $i$ −й ділянці,

$$T_i = M\frac{W_i}{\sum_{k=1}^{n} W_k},$$

де $M$ – загальна кількість потоків; $n$ – кількість кластерів. Формула забезпечує пропорційний розподіл ресурсів відповідно до складності.

$$D_{ij} = \left|K_i - K_j\right|,$$

де $D_{ij}$ – відстань між двома ділянками за складністю.

Якщо $D_{ij}$ менше порогу $\delta$, то ділянки можуть бути об'єднані в один кластер.

На першому рівні дерево кластерів групує матеріали за середнім значенням $K_{BRDF}$. На другому рівні виконується поділ за напрямками падаючого світла. На третьому рівні виконується поділ за напрямками спостереження. Така ієрархія дозволяє автоматично деталізувати лише ті ділянки, де BRDF має високу нелінійність.

Після кластеризації виникає задача розподілу кластерів між потоками. Нехай множина кластерів позначається як $P = \{P_1, P_2, \dots, P_N\}$. Кожен кластер має вагу $W_i$. Потрібно знайти такий розподіл кластерів між $M$ потоками, щоб навантаження було максимально рівномірним. Навантаження $j$-го потоку $L_j = \sum_{i\in S_j} W_i$, де $S_j$ – множина кластерів, призначених цьому потоку. Середнє навантаження потоку знайдемо за формулою $\bar{L} = \frac{1}{M}\sum_{j=1}^{M} L_j$, а функціонал дисбалансу $\Phi = \sum_{j=1}^{M}(L_j - \bar{L})^2$. Мінімізація $\Phi$ означає вирівнювання навантаження між потоками.

Умова $\Phi \to min$ задає оптимізаційну мету адаптивного диспетчера задач. Середньоквадратичне відхилення навантаження потоків дорівнює:

$$\sigma_L = \sqrt{\frac{1}{M}\sum_{j=1}^{M}(L_j - \bar{L})^2}.$$

Коефіцієнт ефективності розпаралелювання BRDF згайдемо за формулою $E_p = 1 - \frac{\sigma_L}{\bar{L}}$.

При $E_p \to 1$ навантаження є наближається до збалансованого. Значення $E_p$ можна використовувати як критерій якості розподілу. Наприклад, $E_p > 0.9$ відповідає високоефективному балансуванню, тоді як $E_p < 0.7$ вказує на необхідність повторної кластеризації або міграції частини задач.

Початковий розподіл задач не завжди є оптимальним. Для цього введемо механізм динамічного балансування.

$$\bar{T} = \frac{1}{M}\sum_{j=1}^{M} T_j,$$

де $\bar{T}$ – середній фактичний час виконання потоків.

Коефіцієнт перевантаження $j$-го потоку визначаємо за формулою $R_j = \frac{T_j}{\bar{T}}$. Якщо $R_j$ перевищує критичне значення, то потік вважається перевантаженим. Умовою запуску перерозподілу задач є нерівність $R_j > R_{cr}$. Обсяг навантаження, який переноситься з перевантаженого потоку $\Delta W = \eta * (T_j - \bar{T})$, де $\eta$ – коефіцієнт адаптації, $0 < \eta < 1$.

Нове навантаження перевантаженого потоку після передачі частини задач можна знайти за формулою $L_k^{new} = L_k - \Delta W$. Нове навантаження менш завантаженого потоку, який отримав частину задач, дорівнює $L_k^{new} = L_k - \Delta W$.

Коефіцієнт вільних ресурсів $i$-го потоку $F_i = 1 - \frac{L_i}{max(L_j)}$. Якщо $F_i$ великий, потік може приймати додаткові задачі. Коефіцієнт зміни карти складності знайдемо за формулою

$$Q = \frac{\sum_{i=1}^{N} \left| K_i^{new} - K_i^{old} \right|}{\sum_{i=1}^{N} K_i^{old}}.$$

Якщо $Q$ перевищує критичний поріг, доцільно виконати повторну кластеризацію параметричного простору.

Додатковим механізмом підвищення продуктивності є кешування результатів у межах кластера. Якщо два звернення до BRDF мають близькі напрямки освітлення та спостереження, повторне повне обчислення може бути замінене локальною апроксимацією.

Нехай $\left|\left|\omega_o - \omega_o^{'}\right|\right| < \varepsilon$ – умова близькості напрямків спостереження, де $\varepsilon$ – малий поріг відстані у просторі напрямків. Апроксимоване значення BRDF у точці $f_r^{'} = f_r + \nabla f_r \cdot \Delta x$, де $\Delta x$ – зміщення у параметричному просторі. Формула відповідає першому члену розкладу Тейлора.

$$R_{cache} = \frac{N_{cache}}{N_{total}}$$

де $R_{cache}$ – коефіцієнт повторного використання кешу; $N_{cache}$ – кількість звернень, оброблених через кеш або апроксимацію; $N_{total}$ – загальна кількість звернень до BRDF.

Кешування не повинно погіршувати якість зображення. Тому для ділянок з високим $K_{BRDF}$ поріг $\varepsilon$ має зменшуватися, а для гладких ділянок – збільшуватися.

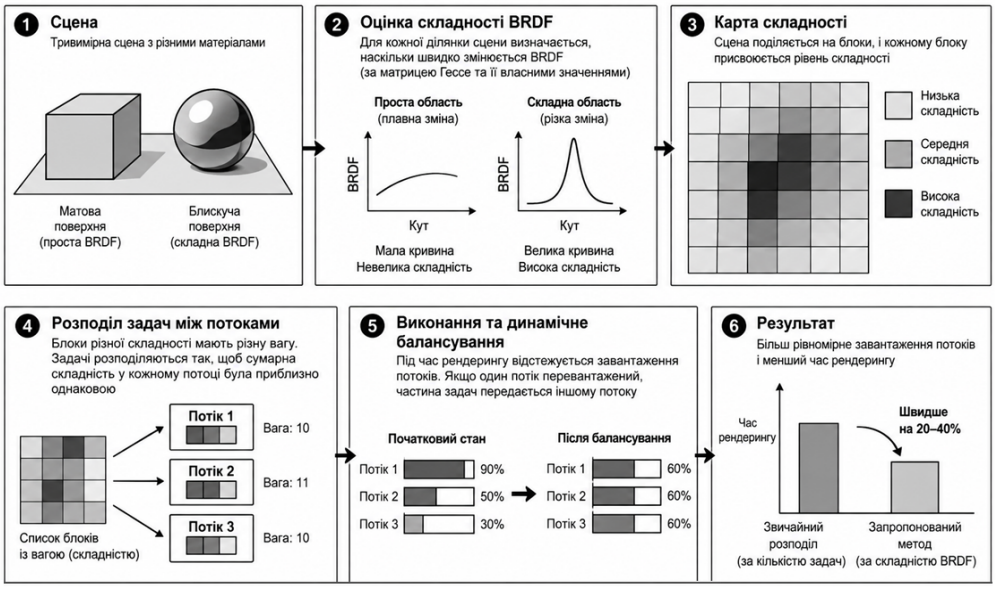


Рисунок 1 – Структурна схема адаптивного розпаралелювання обчислення BRDF

Локальний поріг кешування для і-ї ділянки

$$\varepsilon_i = \frac{\varepsilon_o}{(1 + K_{BRDF,i})},$$

де $\varepsilon_i - \varepsilon_o$ – базовий поріг. Чим більша кривина BRDF, тим меншою має бути область апроксимації.

Рисунок 1 відображає послідовність переходу від аналізу вихідної тривимірної сцени до формування карти локальної складності, розподілу обчислювальних задач між паралельними потоками, динамічного балансування навантаження та отримання кінцевого результату рендерингу. Основною відмінністю запропонованого підходу є те, що під час планування паралельних обчислень враховується не лише кількість пікселів, променів або фрагментів сцени, а й оцінена складність обчислення BRDF для відповідних ділянок параметричного простору.

На першому етапі виконується аналіз тривимірної сцени, її геометрії, матеріалів, параметрів освітлення та умов спостереження. Наявність у сцені поверхонь із різними оптичними властивостями зумовлює неоднорідність обчислювального навантаження. Для дифузних і матових матеріалів значення BRDF, як правило, змінюються більш плавно, тоді як для дзеркальних, металевих або поверхонь із малою шорсткістю функція відбиття може характеризуватися значними локальними змінами. Отже, навіть однакова кількість звернень до BRDF для різних матеріалів не обов'язково відповідає однаковій обчислювальній трудомісткості.

Другий етап передбачає оцінювання локальної складності функції відбиття. Для цього аналізується характер зміни BRDF у

параметричному просторі, який може включати напрямок падаючого світла, напрямок спостереження, шорсткість, металевість, параметри мікрогеометрії та інші характеристики матеріалу. Плавна зміна функції відповідає ділянки з невисокою локальною складністю, тоді як різка зміна BRDF свідчить про підвищену нелінійність і більшу потенційну складність її обчислення. У запропонованому методі така оцінка пов'язується з локальною кривиною функції відбиття, для визначення якої можуть використовуватися градієнт BRDF, матриця Гессе та її власні значення. Максимальне за модулем власне значення матриці Гессе характеризує найбільшу локальну кривину функції у відповідному напрямку параметричного простору.

На третьому етапі формується карта складності. Для цього параметричний простір або множина звернень до BRDF поділяється на окремі блоки, кожному з яких присвоюється певний рівень складності. На рисунку умовно виділено ділянки низької, середньої та високої складності. Така класифікація дає змогу перейти від рівномірного розподілу задач за їх кількістю до вагового розподілу, у якому кожній задачі або кластеру відповідає показник, що характеризує очікувані витрати на його оброблення. Для гладких ділянок можуть формуватися більші обчислювальні блоки, тоді як ділянки з високою локальною кривиною доцільно деталізувати та розподіляти між більшою кількістю виконавчих потоків.

На четвертому етапі здійснюється безпосередній розподіл задач між потоками CPU або GPU. У традиційній схемі кожному потоку може бути призначена приблизно однакова кількість задач, однак їх фактична складність може істотно відрізнятися. У запропонованому підході враховується вагова характеристика кожного обчислювального блока. Задачі розподіляються таким чином, щоб сумарна вага навантаження кожного потоку була максимально наближеною до середнього значення. У результаті потік, який отримує декілька складних ділянок BRDF, може одержати меншу кількість задач, тоді як потік, що обробляє переважно гладкі ділянки, – більшу кількість простіших задач. Такий принцип забезпечує більш адекватне вирівнювання навантаження порівняно з розподілом лише за кількістю операційних блоків.

П'ятий етап відповідає виконанню рендерингу та динамічному контролю стану паралельних потоків. Початковий розподіл задач не завжди залишається оптимальним упродовж усього процесу формування зображення, оскільки під час трасування можуть виникати вторинні промені, відбиття, заломлення та різна глибина рекурсії. Тому в процесі виконання оцінюється фактичний час роботи окремих потоків і ступінь їх завантаження. Якщо навантаження певного потоку перевищує встановлений поріг, частина його задач може бути передана менш завантаженому потоку. Таким чином реалізується механізм динамічного балансування, який дає змогу коригувати початковий розподіл відповідно до реального перебігу обчислень.

Шостий етап характеризує кінцевий результат застосування методу. Очікуваним наслідком більш рівномірного розподілу навантаження є зменшення часу простою окремих потоків, підвищення ступеня використання процесорних ресурсів і скорочення загального часу виконання BRDF-обчислень. Ефективність запропонованого підходу доцільно оцінювати за такими показниками, як час рендерингу, прискорення, ефективність використання потоків, коефіцієнт завантаження ресурсів і коефіцієнт дисбалансу. Саме сукупність цих характеристик дає можливість визначити, наскільки адаптивний розподіл за складністю BRDF є ефективнішим за статичний розподіл за пікселями, тайлами або променями.

Рисунок 1 відображає повний цикл роботи запропонованого методу: від аналізу властивостей матеріалів і визначення локальної складності BRDF до формування вагових характеристик задач, їх розподілу між паралельними потоками та подальшого динамічного балансування. Виконується перехід від формально рівномірного розподілу однакової кількості задач до розподілу за реальною оцінкою їх обчислювальної трудомісткості. Це створює передумови для більш рівномірного завантаження CPU/GPU та підвищення продуктивності фотореалістичного рендерингу.

Для кількісного оцінювання ефективності розпаралелювання використаємо показники прискорення, ефективності паралельного виконання, використання обчислювальних ресурсів і дисбалансу навантаження [3].

Коефіцієнт прискорення визначимо як

$$S = \frac{T_{seq}}{T_{par}},$$

де $T_{seq}$ – час послідовного виконання; $T_{par}$ – час паралельного виконання.

Ефективність використання $M$ потоків визначимо за формулою $E = \frac{S}{M}$. При ідеальному масштабуванні $E$ наближається до 1.

Ступінь використання процесорних $E = \frac{S}{M}$ ресурсів визначимо як

$$U = \frac{\sum_{i=1}^{M} L_i}{M * max(L_i)}.$$

За рівномірного навантаження потоків значення $U$ наближається до одиниці. Нерівномірність розподілу навантаження оцінимо

коефіцієнтом $B = \frac{\sigma(L)}{L_{bar}}$, де $\sigma(L)$ – середньоквадратичне відхилення навантаження потоків. Менше значення $B$ відповідає більш рівномірному розподілу навантаження.

Для запропонованого методу час паралельного виконання подамо у вигляді $T_{par} = max(L_j) + T_s$ де $T_s$ – витрати на синхронізацію та диспетчеризацію задач.

З урахуванням введеного коефіцієнта ефективності розпаралелювання $E_p$ синхронізаційні витрати оцінимо співвідношенням $T_s = T_0 * (1 - E_p)$, де $T_0$ – умовна максимальна вартість синхронізації. Чим вищий $E_p$, тим менші втрати на очікування потоків.

Тоді оцінка прискорення обчислення BRDF набуває такого вигляду:

$$S = \frac{N_c t_{BRDF}}{max(L_j) + T_0(1 - E_p)},$$

де $t_{BRDF}$ – середній час одного обчислення BRDF.

Формула показує зв'язок між балансуванням навантаження та загальним прискоренням.

### *Експериментальна перевірка*

Метою експериментів є перевірка того, чи забезпечує запропонований метод кращий баланс потоків і менший час рендерингу порівняно зі статичним розподілом за тайлами або променями. Рекомендовані тестові сцени (табл. 1) вибрано для різної складності.

Таблиця 1 – Тестові сцени

| № | Сцена | Орієнтовна складність |
|---|---|---|
| 1 | Cornell Box | низька |
| 2 | Sponza | середня |
| 3 | Bistro | висока |
| 4 | San Miguel | дуже висока |
| 5 | Синтетична BRDF-сцена | керована |




Рисунок 2 – Фрагменти тестових фігур

На рисунку 2 наведено зображення тестових фігур. Основні метрики вимірювання зведено в таблицю 2.

Таблиця 2 – Метрики вимірювання

| Метрика | Формула | Пояснення |
|---|---|---|
| Час виконання | $T = t_{finish} - t_{start}$ | Повний час рендерингу або час BRDF-етапу |
| Прискорення | $S = \frac{T_{seq}}{T_{par}}$ | Виграш від паралельного виконання |
| Ефективність | $E = \frac{S}{M}$ | Ефективність використання потоків |
| Використання ресурсів | $U = \frac{\sum L_i}{P\, max(L_i)}$ | Ступінь рівномірності завантаження |
| Дисбаланс | $B = \frac{\sigma(L)}{\bar{L}}$ | Відносне відхилення навантаження потоків |
| Кешування | $R_{cache} = \frac{N_{cache}}{N_{total}}$ | Частка повторно використаних значень BRDF |

Результати вимірювання зведено в таблицю 3.

Таблиця 3 – Результати тестування

| Сцена | Метод | Потоків | Середній час, с | σ часу | $S$ | $E$ | $U$ | $B$ | $R_{cache}$ |
|---|---|---|---|---|---|---|---|---|---|
| Cornell Box | статичний | 1 | 5.840 | 0.083 | 1.00 | 1.00 | 1.00 | 0.00 | 0.00 |
| Cornell Box | статичний | 8 | 0.920 | 0.026 | 6.35 | 0.79 | 0.82 | 0.18 | 0.00 |
| Cornell Box | запропонований | 8 | 0.790 | 0.019 | 7.39 | 0.92 | 0.95 | 0.07 | 0.28 |
| Sponza | статичний | 8 | 3.150 | 0.091 | 5.94 | 0.74 | 0.78 | 0.22 | 0.00 |
| Sponza | запропонований | 8 | 2.550 | 0.061 | 7.33 | 0.92 | 0.94 | 0.08 | 0.34 |
| Bistro | статичний | 16 | 4.820 | 0.167 | 10.08 | 0.63 | 0.71 | 0.31 | 0.00 |
| Bistro | запропонований | 16 | 3.490 | 0.084 | 13.93 | 0.87 | 0.91 | 0.10 | 0.38 |

Запропонований метод найбільш ефективний для сцен із великою різницею складності матеріалів. Якщо всі матеріали майже однакові, перевага над статичним розподілом буде малою. Якщо ж сцена містить одночасно дифузні, дзеркальні, металеві, шорсткі й анізотропні матеріали, параметричне балансування BRDF має зменшити розкид часу виконання потоків.

### *Висновки*

У роботі запропоновано метод адаптивного багаторівневого розпаралелювання обчислення BRDF з урахуванням локальної складності функції відбиття. Складність окремих ділянок параметричного простору оцінюється за локальною кривиною BRDF, що дає змогу формувати вагові характеристики обчислювальних задач. Це особливо важливо для сцен, що містять матеріали з різними відбивними властивостями та, відповідно, різною складністю обчислення. Метод може бути реалізований засобами C++/OpenMP або CUDA та порівнюватися зі статичним тайловим і променевим розподілом. Ефективність доцільно оцінювати за часом рендерингу, прискоренням, рівнем використання обчислювальних ресурсів і коефіцієнтом дисбалансу. Подальшого дослідження потребує вплив локальної апроксимації BRDF на точність формування зображення та загальну продуктивність рендерингу. Попередня оцінка ефективності показала, що використання запропонованого методу може зменшити час рендерингу приблизно на 14–28 % порівняно зі статичним розподілом задач. Для тестових сцен Cornell Box, Sponza та Bistro ефективність використання потоків становила відповідно 0,92, 0,92 та 0,87, тоді як для статичного розподілу – 0,79, 0,74 та 0,63. Найбільший виграш спостерігається для складніших сцен із неоднорідними матеріалами.

### *Використання штучного інтелекту та технологій на основі ШІ*

У процесі підготовки рукопису для перевірки граматики, стилістики та орфографії було використано інструмент штучного інтелекту GPT-5.6 Sol. Штучний інтелект не використовувався для формування наукової гіпотези, розробки математичної моделі, інтерпретації експериментальних результатів або формулювання наукових висновків. Автори несуть повну відповідальність за зміст статті, достовірність наведених даних та їх наукову інтерпретацію.

### *Література*

1. Salesin K., Knobelspiesse K. D., Chowdhary J., Zhai P.-W., Jarosz W. Unifying Radiative Transfer Models in Computer Graphics and Remote Sensing, Part I: A Survey. *Journal of Quantitative Spectroscopy and Radiative Transfer*. 2024. Vol. 314. Art. 108847. DOI: https://doi.org/10.1016/j.jqsrt.2023.108847.
2. Romanyuk O. N., Zavalniuk Y. K., Chekhmestruk R. Y., Mykhaylov P. I., Hamza A. Combined Bidirectional Reflectance Distribution Functions Usage for Increasing Images Creation Productivity. *Applied Aspects of Information Technology*. 2023. Vol. 6, No. 2. P. 130–138. DOI: https://doi.org/10.15276/aait.06.2023.9.
3. Романюк О. Н., Бобко О. Л., Снігур А. В. Метричне оцінювання розпаралелення рендерингу. *Наукові праці Вінницького національного технічного університету*. 2025. № 2. С. 125–136. DOI: https://doi.org/10.31649/2307-5376-2025-2-125-136.
4. Romanyuk O., Zavalniuk Y., Romanyuk O., Kotlyk S., Bobko O., Melnyk O. New Illumination Model for Rendering. *2025 15th International Conference on Advanced Computer Information Technologies (ACIT)*. 2025. P. 723–727. DOI: https://doi.org/10.1109/ACIT65614.2025.11185657.
5. Sforza D., Pellacini F. Numerical Approximations for Energy Preserving Microfacet Models. *Computers & Graphics*. 2023. Vol. 114. P. 36–44. DOI: https://doi.org/10.1016/j.cag.2023.05.012.
6. Huang B., Peng B., Ren Q., Liao S. Modeling and Analysis of Spectral Polarization BRDF Based on Microfacet Theory. *Journal of Quantitative Spectroscopy and Radiative Transfer*. 2024. Vol. 313. Art. 108830. DOI: https://doi.org/10.1016/j.jqsrt.2023.108830.

7. Bitterli B., d'Eon E. A Position-Free Path Integral for Homogeneous Slabs and Multiple Scattering on Smith Microfacets. *Computer Graphics Forum*. 2022. Vol. 41, No. 4. P. 93–104. DOI: https://doi.org/10.1111/cgf.14589.
8. Lin D., Kettunen M., Bitterli B., Pantaleoni J., Yuksel C., Wyman C. Generalized Resampled Importance Sampling: Foundations of ReSTIR. *ACM Transactions on Graphics*. 2022. Vol. 41, No. 4. Art. 75. P. 75:1–75:23. DOI: https://doi.org/10.1145/3528223.3530158.

### *References*

1. Salesin, K., Knobelspiesse, K.D., Chowdhary, J., Zhai, P.-W., Jarosz, W. (2024), "Unifying Radiative Transfer Models in Computer Graphics and Remote Sensing, Part I: A Survey", *Journal of Quantitative Spectroscopy and Radiative Transfer*, Vol. 314, Art. 108847. DOI: https://doi.org/10.1016/j.jqsrt.2023.108847.
2. Romanyuk, O.N., Zavalniuk, Y.K., Chekhmestruk, R.Y., Mykhaylov, P.I., Hamza, A. (2023), "Combined Bidirectional Reflectance Distribution Functions Usage for Increasing Images Creation Productivity", *Applied Aspects of Information Technology*, Vol. 6, No. 2, pp. 130–138. DOI: https://doi.org/10.15276/aait.06.2023.9.
3. Romaniuk, O.N., Bobko, O.L., Snihur, A.V. (2025), "Metric evaluation of rendering parallelization" ["Metrychne otsiniuvannia rozparaleliuvannia renderynhu"], *Scientific Works of Vinnytsia National Technical University*, No. 2, pp. 125–136. DOI: https://doi.org/10.31649/2307-5376-2025-2-125-136.
4. Romanyuk, O., Zavalniuk, Y., Romanyuk, O., Kotlyk, S., Bobko, O., Melnyk, O. (2025), "New Illumination Model for Rendering", *2025 15th International Conference on Advanced Computer Information Technologies (ACIT)*, pp. 723–727. DOI: https://doi.org/10.1109/ACIT65614.2025.11185657.
5. Sforza, D., Pellacini, F. (2023), "Numerical Approximations for Energy Preserving Microfacet Models", *Computers & Graphics*, Vol. 114, pp. 36–44. DOI: https://doi.org/10.1016/j.cag.2023.05.012.
6. Huang, B., Peng, B., Ren, Q., Liao, S. (2024), "Modeling and Analysis of Spectral Polarization BRDF Based on Microfacet Theory", *Journal of Quantitative Spectroscopy and Radiative Transfer*, Vol. 313, Art. 108830. DOI: https://doi.org/10.1016/j.jqsrt.2023.108830.
7. Bitterli, B., d'Eon, E. (2022), "A Position-Free Path Integral for Homogeneous Slabs and Multiple Scattering on Smith Microfacets", *Computer Graphics Forum*, Vol. 41, No. 4, pp. 93–104. DOI: https://doi.org/10.1111/cgf.14589.
8. Lin, D., Kettunen, M., Bitterli, B., Pantaleoni, J., Yuksel, C., Wyman, C. (2022), "Generalized Resampled Importance Sampling: Foundations of ReSTIR", *ACM Transactions on Graphics*, Vol. 41, No. 4, Art. 75, pp. 75:1–75:23. DOI: https://doi.org/10.1145/3528223.3530158.

*Надійшла до редакції 14.09.2026*

**O. N. ROMANIUK**, **O. L. BOBKO**
[1,2]Vinnytsia National Technical University, Vinnytsia, Ukraine
[1]rom8591@gmail.com, [2]Oleksii.bobko@gmail.com

**A METHOD FOR BALANCING COMPUTATIONAL LOAD IN MULTI-THREADED 3D RENDERING SOFTWARE SYSTEMS**

This paper addresses the issue of improving photorealistic rendering performance through a more uniform distribution of computations across parallel threads. Attention is paid to the Bidirectional Reflectance Distribution Function (BRDF), the computational complexity of which varies depending on the material. While the BRDF for matte surfaces changes relatively smoothly, the function's behaviour for specular, metallic, and rough materials can be significantly more complex. Consequently, traditional task distribution based on the number of pixels, rays, or tiles does not always ensure balanced CPU/GPU utilization. An adaptive multi-level parallelization method is proposed, in which BRDF computational complexity is estimated based on the local curvature of the reflectance function. This involves the use of the gradient, the Hessian matrix, and its eigenvalues. Based on this estimate, regions of the parameter space are classified as smooth, moderately complex, or complex. A weight characteristic is generated for each region and factored into the task distribution across threads. This enables load balancing based not only on the number of tasks but also on their actual computational cost. The method incorporates dynamic load balancing; if a thread becomes overloaded, a portion of its tasks is transferred to a less loaded thread. Additionally, caching and local BRDF approximation may be employed for similar lighting and viewing parameters. Rendering time, speedup, thread utilization efficiency, resource utilization ratio, and load imbalance ratio are used to evaluate performance. The Cornell Box, Sponza, Bistro, and San Miguel scenes, along with a synthetic BRDF scene, are used to validate the method. Preliminary evaluations indicate a potential reduction in rendering time of approximately 14–28% for scenes with heterogeneous materials.

***Keywords: BRDF, rendering, parallelization, GPU, load balancing, rendering, computational process, software optimization, parallel programming.***