

СТРУКТУРА ТА ПРИНЦИПИ ФУНКЦІОНУВАННЯ СПЕЦІАЛІЗОВАНИХ ПРИСТРОЇВ ЗМЕНШЕННЯ ОБСЯГІВ ДАНИХ ТРИАНГУЛЯЦІЙНОГО ОПИСУ ТРИВИМІРНИХ ОБ’ЄКТІВ

© Акимішин О.І., 2008

Запропоновано алгоритм розбиття триангуляції для її паралельної обробки та на його основі розроблено базову структуру спеціалізованих пристроїв зменшення обсягів даних триангуляційного опису тривимірних об’єктів.

The method of triangulation partitioning into separated processing elements to their parallel processing is developed. The basic structures and functioning principles of the hardware accelerators of data reduction of 3D object triangular representation based on proposed method have been developed.

Вступ. Системи візуального подання інформації мають важливе значення в різноманітних галузях людської діяльності, особливо там де безпосередній контроль процесу людиною є або надважкий або неможливий. У таких випадках використовуються комплекси апаратно-програмних засобів для збору інформації з подальшим її опрацюванням та візуалізацією засобами ПЕОМ. Використання новітніх алгоритмів візуалізації та методів опрацювання моделей тривимірних об’єктів сприяє появі нових завдань у цій галузі та пошуку шляхів ефективного їх вирішення.

Найпоширенішим способом подання тривимірних об’єктів в комп’ютерній графіці вважається опис їхніх поверхонь триангуляційними сітками. Як правило, такі описи забезпечують заданий рівень деталізації, проте часто містять мільйони трикутників, що для подання багатьох об’єктів є надлишковими. Тому актуальним є завдання зменшення обсягів даних триангуляційного опису об’єктів перед їх обробкою та візуалізацією.

Крім того, час, за який необхідно забезпечити зменшення обсягів даних триангуляційного опису об’єктів, є обмежений специфікою прикладної галузі, для якої виконується візуалізація зображень. Незважаючи на стрімке зростання характеристик універсальної обчислювальної техніки, їх можливостей замало для виконання цього завдання за прийнятний час. Тому актуальним є розроблення спеціалізованих прискорювачів зменшення обсягів даних триангуляційного опису тривимірних об’єктів.

Огляд літературних джерел. Завдання обробки обсягів даних, що сягають гігабайт інформації та не поміщаються в основній пам’яті комп’ютера, є актуальним у системах реконструкції тривимірних зображень за даними пристроїв об’ємного сканування.

Зокрема в [1, 2] описано проект фірми ІВМ сумісно з командою Стенфордського університету, що полягав у деталізованому скануванні скульптури Мікеланджело – Давида. Використовуючи лазерний сканер, виконано сканування скульптури з роздільною здатністю 1 мм. Після реконструкції відсканованих даних отримано тривимірну модель, що містить 28 мільйонів вершин та 56 мільйонів трикутників. Лише для збереження моделі такого розміру потрібно близько 1 Gb дискового простору. Аналогічними є розміри триангуляційних сіток, отриманих під час виділення об’єктів за об’ємними воксельними даними, зокрема за даними промислової комп’ютерної томографії. Вказані моделі не тільки не поміщаються в основній пам’яті сучасних комп’ютерів, але і сягають граничних розмірів чотиригігабайтного адресного простору 32-бітних комп’ютерів [3]. Суміжні приклади деяких моделей та їх характеристики наведено в табл. 1. При візуалізації та опрацюванні вказаних обсягів даних виникають дві основні проблеми:

- опрацювання моделей, об'єми даних, для представлення яких не поміщаються в основній пам'яті комп'ютера, що істотно сповільнює їх обробку;
- забезпечення швидкого відображення моделі тривимірного об'єкта на екрані комп'ютера;

Таблиця 1

Розміри тривимірних моделей в тисячах (К) та мільйонах (М) елементів

Назва моделі	Кількість вершин	Кількість трикутників
Buddha	544 К	1.1 М
David (2 mm)	4.1 М	8.3 М
David (1 mm)	28 М	56 М
St. Matthew	187 М	373 М
isosurface	235 М	469 М

Відомі методи для опрацювання згаданих сіток, головною характеристикою яких є опрацювання даних поза ядром (out-of-core processing) [4-6]. Ідея полягає в тому, що в основній пам'яті комп'ютера по черзі розміщуються неперервні фрагменти сітки, над якими виконується обробка, в той час коли значно більша частина даних перебуває на диску.

Для опрацювання послідовностей даних таких розмірів доцільно було б реалізувати апаратний пристрій. Апаратна реалізація пристроїв такого класу є доволі складним завданням, яке не є успішно вирішеним сьогодні. Ця проблема зумовлена тим, що методи зменшення обсягів даних триангуляційного опису є дуже часто залежними від конкретного застосування та формату подання вхідних/вихідних даних [7]. У роботі [8] авторами вказано, що вони виконали першу імплементацію на ПЛІС алгоритму компресії/декомпресії триангуляційних сіток, однак основним застосуванням цього пристрою є підвищення пропускної здатності під час пересилання даних по шині відеоадаптера. Крім того стиснені дані вимагають етап декомпресії перед опрацюванням та візуалізацією.

Мета і завдання дослідження. Метою досліджень є розроблення спеціалізованих обчислювальних пристроїв зменшення обсягів даних триангуляційного опису об'єктів. Реалізація мети передбачає виконання таких завдань:

- розробити апаратно-орієнтований метод зменшення обсягів даних триангуляційного опису об'єктів, а також для пришвидшення обробки даних розробити метод розбиття вхідних даних на окремі елементи опрацювання;
- розробити базову структуру та моделі спеціалізованих обчислювальних пристроїв для зменшення обсягів даних та виконати їх синтез, використовуючи засоби автоматизованого проектування комп'ютерних систем.

Розбиття даних для паралельного оброблення. Ключові підходи для ефективного опрацювання надвеликих обсягів даних можна визначити такими пунктами [9]:

1. Роздільність даних. Дані повинні бути роздільними, тобто вхідна послідовність може бути розділена на фрагменти. Алгоритм розділення повинен бути простим та ефективним. Повинно бути забезпечено коректне опрацювання фрагментів даних.
2. Відображення вхідних даних. Тобто потрібно забезпечити можливість визначати, який фрагмент повинен бути взятий на вході алгоритму для отримання відповідного результату на його виході. Це дозволяє контролювати розмір фрагментів опрацювання даних.
3. Інваріантність (стабільність) результатів. Результати виконання алгоритму не повинні залежати від кількості фрагментів вхідних даних та від типу виконання (одно- та багатопотокове опрацювання).

Для забезпечення незалежного опрацювання елементів триангуляції використано метод розбиття вхідних даних на окремі елементи [10]. Він ґрунтується на розбитті даних, що виконується при паралельній обробці. Одиницею розбиття прийнято ребро триангуляції із суміжними йому трикутниками. Суть методу полягає у розділенні геометрично незалежних елементів вхідної сітки для їх подальшого опрацювання.

Виділення окремих елементів опрацювання передбачає виконання такої послідовності кроків (рис. 1):

Крок 1. Вчитати вхідну сітку або її фрагмент у пам'ять комп'ютера.

Крок 2. Помітити всі вершини вхідної ТС, як невикористані.

Крок 3. Для кожного ребра ТС виконувати:

3.1. Перевірити вершини, які утворюють це ребро. Якщо вони помічені як невикористані, то перевірити вершини в їх околі.

3.1.1. Якщо всі вершини в околі ребра є помічені як невикористані, то утворити незалежний елемент опрацювання триангуляції та помітити вказані вершини як використані.

3.1.2. Вершини, помічені як використані, пропускаються.

Крок 4. Вивести список незалежних елементів опрацювання.

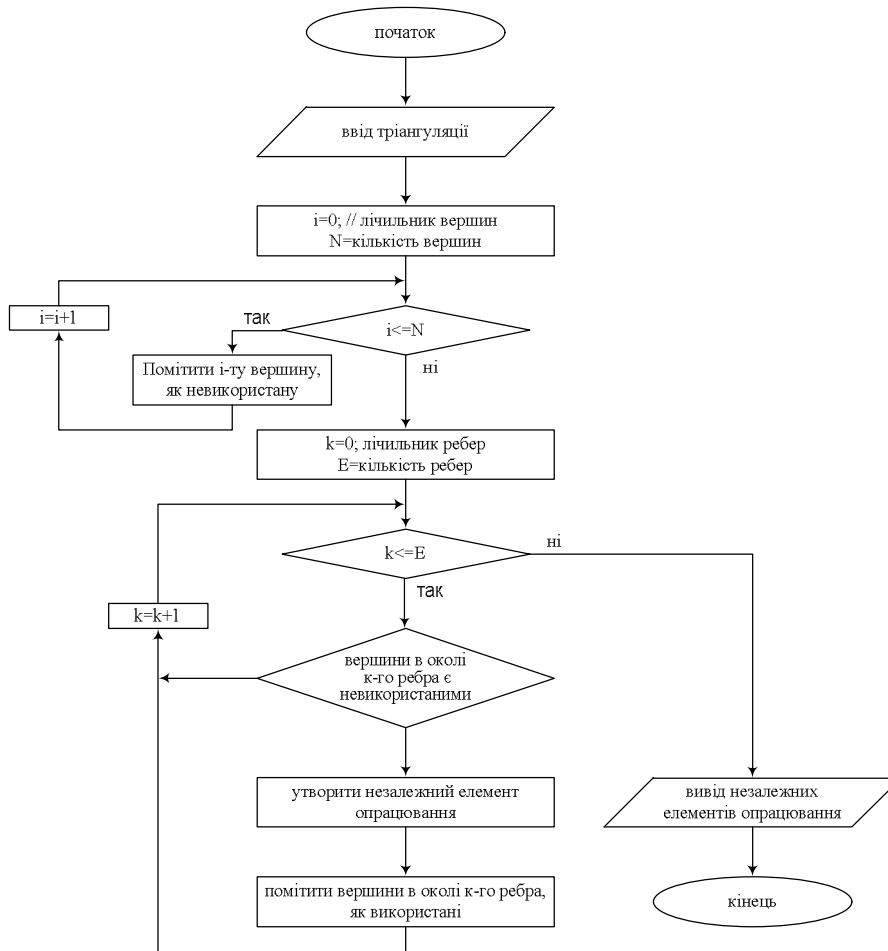


Рис. 1. Граф-схема алгоритму виділення окремих елементів опрацювання

Використання описаного алгоритму розбиття вхідних даних дає можливість отримати на вході спеціалізованого пристрою послідовність окремих блоків даних, що можуть надходити на обробку одною послідовністю або у багатопотоковому режимі. Саме ця особливість даних дозволяє створювати різноманітні структури спеціалізованих пристроїв для зменшення обсягів даних триангуляційного опису об'єктів з різними характеристиками затрат часу та обладнання.

Розробка базової структури пристроїв зменшення обсягів даних. Розбивши вхідні дані на окремі елементи опрацювання, отримуємо послідовність блоків даних, що можуть бути опрацьовані незалежно один від одного. Потокове опрацювання отриманих блоків запропоновано виконувати так:

Крок 1. Вчитати блок даних та розмістити його у вхідній пам'яті;

Крок 2. Обчислити відхилення, що виникає внаслідок виконання локальної модифікації над поточним блоком даних;

Крок 3. Порівняти обчислене відхилення із заданою величиною допустимого відхилення;

Крок 4. Якщо обчислене відхилення менше або дорівнює величині заданого відхилення, то виконати локальну модифікацію над поточним блоком даних;

Крок 5. Вивести опрацьований блок даних у вихідну пам'ять.

Базову структуру апаратних прискорювачів зменшення обсягів даних, що забезпечує потокову обробку даних, з можливістю розділення кроків алгоритму конвеєрними регістрами наведено на рис. 2.

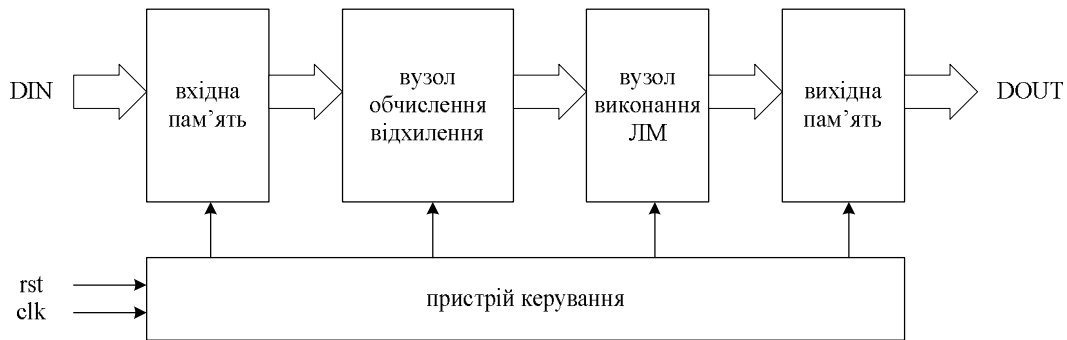


Рис. 2. Загальна структура спеціалізованих пристроїв зменшення обсягів даних триангуляційного опису тривимірних об'єктів

Після надходження даних по вхідних портах виконується обчислення відхилення (ОВ), що виникне внаслідок локальної модифікації (ЛМ) над елементом триангуляційної сітки. Залежно від результату порівняння обчисленого відхилення із заданим значенням допустимого відхилення виконується ЛМ над елементами триангуляції. Видача результатів здійснюється по вихідному порту (DOUT).

Наступними етапами розробки пристрою є розробка вхідного та вихідного блоків пам'яті, вузла ОВ, вузла виконання ЛМ та пристрою керування.

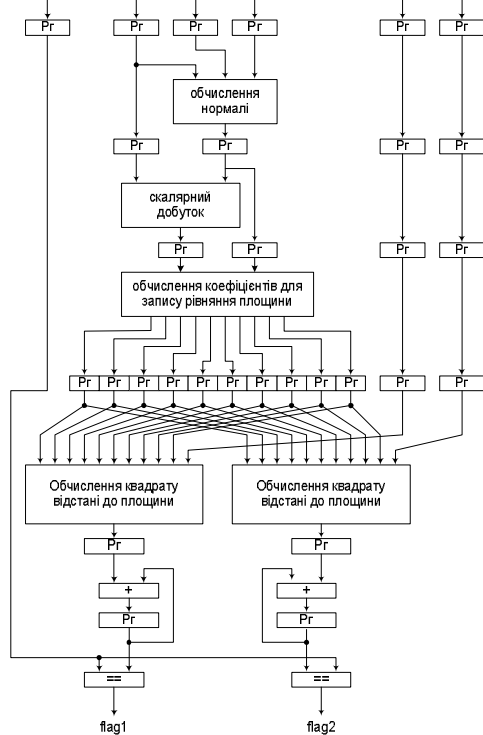


Рис. 3. Структурна схема вузла обчислення відхилення

Завданнями вузла ОВ є обчислення відхилення, яке виникне внаслідок виконання ЛМ, порівняння його із значенням заданого відхилення та формування ознак про можливість виконання ЛМ над біжучим фрагментом триангуляції. Основними елементами вузла ОВ (рис. 3) є обчислення нормалі до площини, обчислення коефіцієнтів для знаходження квадрата відстані від вершини до площини та обчислення відстані від вершини до площини.

При такій реалізації вузол ОВ має 6 входів та 2 виходи. На входи вузла подаються координати вершин, що задають суміжними ребру, над яким виконується ЛМ, трикутники, координати вершин, що утворюють це ребро, та задане значення допустимого відхилення. На виході формуються ознаки можливості виконання ЛМ над опрацьованим блоком даних.

Вузол ОВ є конвеєрним, оскільки потік даних розбитий регістрами на 5 ярусів. Частота роботи вузла ОВ визначається, як

$$f=1/T_{max}, \quad (1)$$

де T_{max} – максимальна затримка спрацювання внутрішніх елементів вузла ОВ.

Аналіз розроблених графів алгоритмів виконання базових операцій свідчить, що максимальну затримку обробки даних має вузол обчислення квадрата відстані від вершини до площини. На основі (1) частота роботи вузла ОВ буде визначатися часом обчислення квадрата відстані від вершини до площини.

Вимогами до пам'яті є забезпечення швидкого надходження даних на всі вхідні порти вузла ОВ. Тому доцільно використати багатопортову пам'ять з можливістю її постійного завантаження вхідними даними. Огляд відомих типів пам'яті дозволив взяти за основу багатопортову сортувальну пам'ять та на її основі розробити структури, що є ефективними для виконання поставленого завдання. Інтерфейс та діаграми функціонування розробленої пам'яті наведено на рис. 4. Вона має множини вхідних та вихідних портів, входи для задання із блоком даних номера вихідного порту, по якому потрібно здійснити видачу цього блока даних та номер даних у вихідному масиві. На діаграмі виділено послідовне потактове завантаження блоків даних та їх одночасну видачу по трьох вихідних портах.

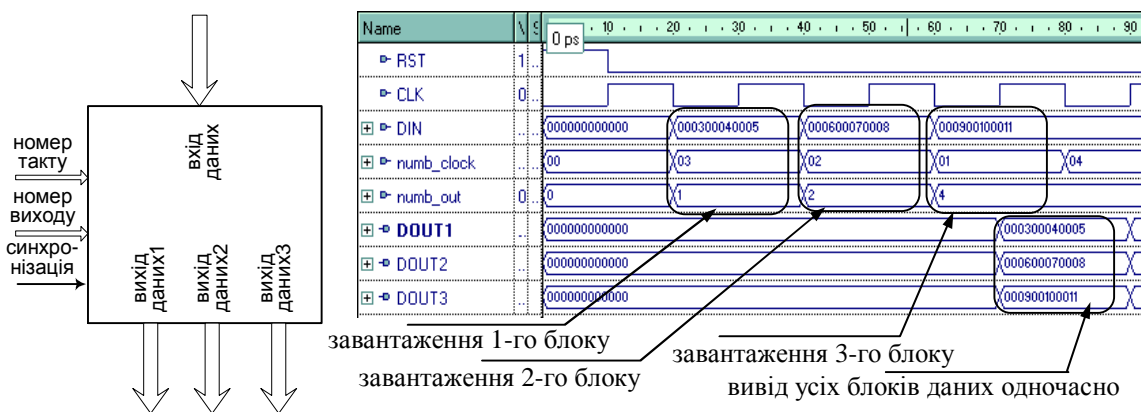


Рис. 4. Інтерфейс та результати моделювання роботи вхідної пам'яті

Блоком даних, над яким виконується локальна модифікація, є ребро та суміжні до нього трикутники. Залежно від відхилення, що виникає внаслідок виконання ЛМ, можливими є три варіанти виконання ЛМ, що графічно зображені на рис. 5, зокрема виконання колапсу ребра (e_1, e_2) у вершину e_1 (рис. 5.а); виконання колапсу ребра (e_1, e_2) у вершину e_2 (рис. 5.в) та залишення початкового блока даних без змін, якщо відхилення, що виникає внаслідок виконання ЛМ, перевищує задане значення (рис. 5.б).

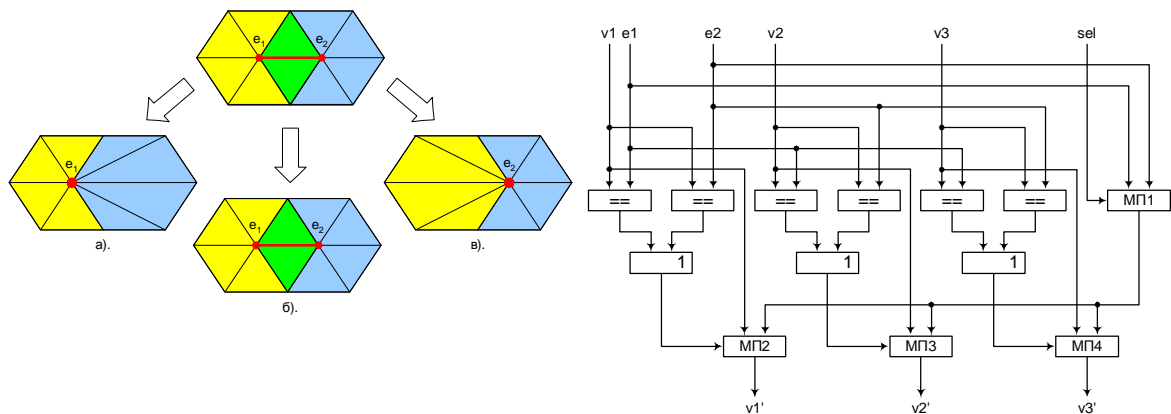


Рис. 5. Структура та принцип функціонування вузла виконання ЛМ

Найпростішим є випадок, зображений на рис. 5.б. Для його реалізації достатньо отримати весь фрагмент даних із входу та вивести його у тій самій послідовності. Для реалізації випадків рис. 5.а та рис. 5.б необхідно виконати такі дії: на основі ознак, що генеруються вузлом обчислення відхилення, визначити вершину, що буде видалена з моделі, та замінити її в описі усіх суміжних з нею трикутників на протилежну. Трикутники, що є спільними для обох вершин, видалити з моделі.

Розроблені внутрішні вузли та пристрої зменшення обсягів даних тріангуляційного опису тривимірних об'єктів реалізовано їхнім описанням мовою VHDL, виконано їх поведінкове моделювання та досліджено характеристики.

Висновки: 1. Вдосконалено метод розбиття тріангуляційних сіток на окремі елементи опрацювання, що дає можливість прискорення обробки даних їх конвеєрною чи паралельною обробкою. Виконано програмну реалізацію розробленого методу та перевірено його працездатність на тестових даних.

2. Розроблено базову структуру, принципи функціонування та VHDL-модель спеціалізованих пристроїв зменшення обсягів даних тріангуляційного опису об'єктів, засновану на розроблених методах, а також виконано її функціональне моделювання. Розроблена структура дає можливість синтезу комп'ютерних пристроїв для зменшення обсягів даних, використовуючи засоби сучасних технологій.

1. Bernardini, F., Martin, I., Mittleman, J., Rushmeier, H., and Taubin, G. *Building a digital model of Michelangelo's Florentine Pieta* // *IEEE Computer Graphics and Applications*. – 2000. – Vol. 22(1). – P. 59–67. 2. Levoy M., Pulli K., Curless B., Rusinkiewicz S., Koller D., Pereira L., Ginzton M., Anderson S., Davis J., Ginsberg J., Shade J., and Fulk D. *The Digital Michelangelo Project* // *In SIGGRAPH Proceedings*. – 2000. – P. 131–144. 3. M. Isenburg, P. Lindstrom, S. Gumhold, and J. Snoeyinklarge. *Mesh Simplification Using Processing Sequences* // *Proceedings of IEEE Visualization*. – 2003. – Seattle, Washington. – P. 19–24. 4. Silva C., Chiang, Y., El-Sana, J., and Lindstrom, P. *Out-of-core algorithms for scientific visualization and computer graphics* // *In visualization'02 course notes*. 5. Pavel Borodin, Michael Guthe, Reinhard Klein. *Out-of-Core Simplification with Guaranteed Error Tolerance* // *VMV*. – 2003. Munich, Germany. 6. Dirk Bartz *Large Model Visualization: Techniques and Applications* // *Journal of WSCG*. – 2003. – Vol.11, No.1. – Plzen, Czech Republic. 7. Joanna Leng. *The IEEE Viz 2001 Conference. Trip Report. San Diego, California, USA, 2001*. 8. Tulika Mitra, Tzi-cker Chiueh. *An FPGA Implementation of Triangle Mesh Decompression*, New York, 2004. 9. C. Charles Law, William J. Schroeder, Kenneth M. Martin, Joshua. Temkin. *A Multi-Threaded Streaming Pipeline Architecture for Large Structured Data Sets*. 10. Акимішин О.І. *Виділення незалежних елементів опрацювання тріангуляційних сіток в тривимірному просторі* // *Мат. III Міжнар. наук.-техн. конф. "Сучасні проблеми радіоелектроніки, телекомунікацій та приладобудування" СПРТП-2007*. – Вінниця, 2007. – С. 117–118.