

МОДИФІКАЦІЯ ІЄРАРХІЧНОГО МЕТОДУ RBF ДЛЯ ОТРИМАННЯ 3D-МОДЕЛЕЙ ЗА РЕЗУЛЬТАТАМИ ЛАЗЕРНОГО СКАНУВАННЯ

В.С. Бабков

Babkov, V.S. (2008) “Modification of hierarchical RBF method for 3D-modelling based on laser scan result”. *Proc. Int. Conference “Modern problems and achievement of radio, communication and informatics”*, Zaporizhzhya National Technical University

In article problem of surface build in 3D based on 3D laser scan data is reviewed. Modified method based on hierarchical RBF but adapted for 3D and has decrease time cost is described. Method efficiency is experimentally accepted.

В работе рассматривается задача построения поверхности в пространстве 3D на основе данных трехмерного лазерного сканирования. Изложена сущность модифицированного метода, который базируется на методе иерархической RBF, но адаптирован для трехмерного пространства и имеет улучшенные временные характеристики. Эффективность метода подтверждена экспериментально.

У роботі розглядається задача побудови поверхні у просторі 3D на основі даних тривимірного лазерного сканування. Викладено сутність модифікованого алгоритму, який базується на методі ієрархічної RBF, але адаптований до тривимірного простору і має покращенні часові характеристики. Ефективність методу підтверджено експериментально.

Вступ

У наш час у багатьох галузях науки і техніки важливу роль відіграє використання тривимірних комп'ютерних моделей, побудованих за результатами дослідження реальних об'єктів складної форми. Подібна задача виникає, наприклад, у таких випадках: пошук дефектів у структурі об'єктів, вивчення внутрішньої структури об'єкта без його руйнування, відновлення об'єкту за неповними даними, побудова тривимірних моделей рухливих об'єктів у реальному часі; оцінка, реконструкція і проектування великих промислових об'єктів і ділянок місцевості.

Існує велика кількість методів одержання проєкційних даних для побудови тривимірних моделей. Загальна риса цих методів – видача результатів сканування у вигляді „хмари” точок, що описують поверхню об'єкта або системи взаємопов'язаних об'єктів. Зокрема при одержанні проєкційних даних про досить великі об'єкти складної форми

(інженерні споруди, будинки, ділянки місцевості) застосовується метод лазерного сканування зовнішньої поверхні [1].

Аналіз стану питання.

За результатами аналізу алгоритмів та методів побудови тривимірних поверхневих моделей реальних об'єктів за проекційними даними, який було зроблено у [2], можна виділити наступні етапи обробки, які здійснюють найбільший внесок у загальний час реконструкції:

- розрахунок функції RBF для всіх пар проекційних точок;
- розв'язання системи лінійних рівнянь великої розмірності;
- обчислення поверхні за допомогою інтерполянта.

На кожному з цих етапів обчислювальна складність та час виконання операцій визначаються значенням N - кількістю проекційних точок.

Проаналізувавши відомі методи побудови поверхневих моделей, можна зробити висновок, що зменшення обчислювальної та просторової складності досягається за рахунок зменшення кількості проекційних точок, що беруть участь у розрахунках (методи minRBF, csRBF, FMM), а також за рахунок декомпозиції точок на групи [3].

Подальше зменшення часових витрат на різних етапах побудови тривимірної моделі доцільно здійснювати за рахунок розпаралелювання алгоритмів та методів. Згідно з висновками, які були зроблені у [4], відомі алгоритми та структури даних не пристосовані до паралельної реалізації. Тому у роботі ставиться задача запропонувати та дослідити засоби підвищення ефективності алгоритмів, з точки зору зменшення часових витрат.

Для розв'язання поставленої задачі пропонується модифікувати відомі методи, засновані на декомпозиції точок, для паралельної реалізації. Схема подібних методів наведена на рис. 1.

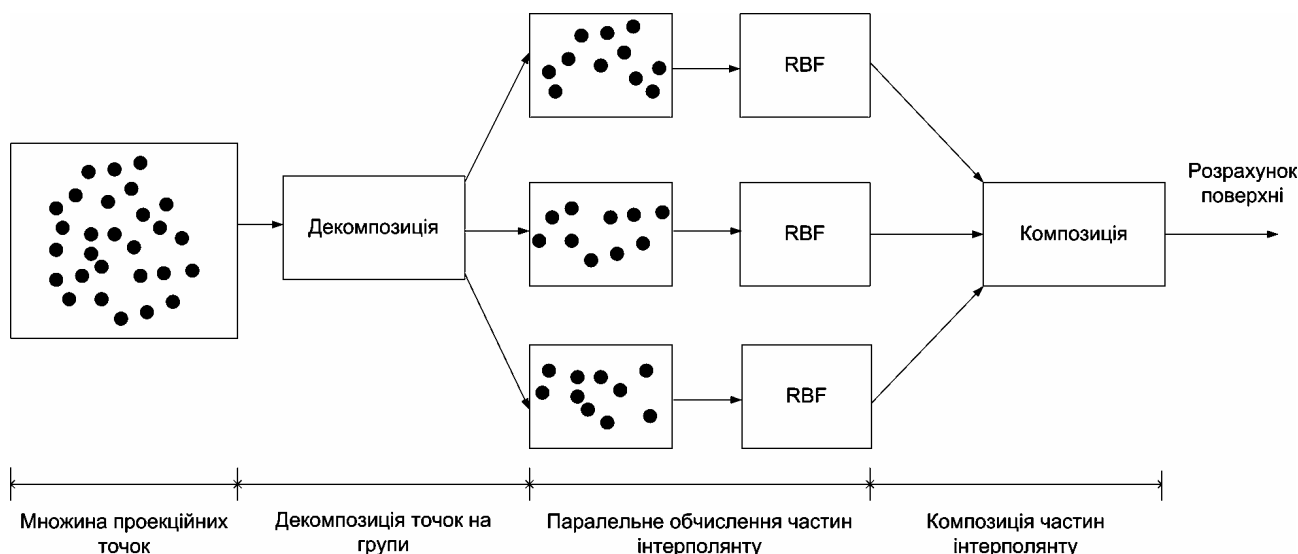


Рисунок 1 – Загальна схема паралельного методу побудови поверхневої моделі

Як прототип даного методу використовується метод ієрархічної RBF-інтерполяції, який було запропоновано в [3] для побудови тривимірних ландшафтів за контурними геодезичними даними .

Наведений метод для застосування у якості методу розрахунку поверхні за тривимірними проекційними даними має такі недоліки:

- метод передбачає використання двовимірних доменів (площин);
- на етапі декомпозиції використовується процедура впорядкування, яка при значній кількості проекційних точок ($10^6..10^9$) [6] призводить до великих часових витрат;
- часові характеристики визначені за експериментальними даними [3], показують, що для проекційних даних із кількістю точок $\approx 10^5$ час побудови поверхні складає у середньому 100..500 с, що не дозволяє виконувати реконструкцію поверхні у реальному часі, особливо для динамічних об'єктів та при значно більшій кількості проекційних точок;
- структура поділу простору на домени не є регулярною, тобто домени відрізняються один від одного і їх структура прив'язана до розташування точок у просторі.

У роботі [5] була здійснена спроба адаптувати даний метод до використання у тривимірному просторі, але при цьому до розташування центрів інтерполяції висувалися жорсткі вимоги. Декомпозиція на групи відбувалася за рахунок природного поділу точок на групи, розташовані у паралельних площинах (як це найчастіше буває при отриманні проекційних даних методами неруйнівного сканування – томографія і т.п.). Таким чином ієрархічний поділ на піддомени не використовувався, а глобальна функція інтерполяції обчислювалася як лінійна комбінація всіх локальних розв'язань з урахуванням відносного розташування паралельних площин одна від одної.

Опис модифікованого методу побудови поверхні за проекційними даними.

Для використання ієрархічного методу для реконструкції поверхні за проекційними точками у тривимірному просторі та для зменшення обчислювальної складності пропонується наступна модифікація.

Розглянемо послідовно етапи декомпозиції та композиції. Пропонується для тривимірному простору, у якому міститься множина проекційних точок P , така що $P = \{c_i = (x_i, y_i, z_i) \mid_{i=1}^N\} \subset R^3$, здійснювати рекурсивну декомпозицію на рівні частини, що не перехрещуються. Тобто нехай задана множина точок P , яка розташована у габаритному об'єкті (box) B^0 , із центром в точці c^0 та стороною a^0 . При декомпозиції на кожному рівні

домен B^l розбивається на вісім рівних піддоменів $b_1^{l+1}, \dots, b_8^{l+1}$, що не перехрещуються, тобто $\bigcap_{i=1}^8 b_i^{l+1} = \emptyset$. Це забезпечить регулярність структури даних, яка утворюється при декомпозиції. Параметри доменів обчислюються за наступними формулами:

$$a^{l+1} = \frac{a^l}{2}, \quad (1)$$

$$c_1^{l+1} = c^l(x - \frac{a^{l+1}}{2}, y - \frac{a^{l+1}}{2}, z - \frac{a^{l+1}}{2}), \dots, c_8^{l+1} = c^l(x - \frac{a^{l+1}}{2}, y + \frac{a^{l+1}}{2}, z + \frac{a^{l+1}}{2}).$$

Множина точок P^l розподіляється між піддоменами за просторовою приналежністю кожному з них, тому на кожному рівні декомпозиції отримується вісім підмножин $P_1^{l+1}, \dots, P_8^{l+1}$. Таким чином зникає необхідність у попередньому впорядкуванні центрів інтерполяції у кожному домені. При такому методі розподілу точок між піддоменами можлива ситуація, за якої піддомен не міститиме жодної точки. Такі піддомени не можуть брати участі в обчисленні глобальної функції інтерполяції і виконувати для них подальшу декомпозицію не потрібно. Тому на кожному кроці формується домен наступного рівня $l+1$, який являє собою множину піддоменів, до якої слід застосовувати подальшу декомпозицію:

$$B^{l+1} = \{b_1^{l+1}, \dots, b_k^{l+1}\}, \quad (2)$$

де

$$k = \overline{1,8}$$

$$\forall b_i^{l+1} \in B^{l+1} \Rightarrow P_i^{l+1} \neq \emptyset, i = \overline{1,k}$$

У результаті отримується деревовидна структура.

Для довільної точки P глобальне розв'язання пропонується відшукувати за допомогою послідовного поєднання локальних розв'язань, починаючи з найнижчого рівня декомпозиції. Тобто для точки P глобальне розв'язання на рівні l буде обчислюватися за формулою:

$$f^l(p) = C(F^{l+1}(p)), \quad (3)$$

де

C - деяка функція, яка забезпечує безперервність розв'язання на межах піддоменів;

F^{l+1} - множина значень локальних розв'язань у піддоменах нижчого рівня.

$$F^{l+1}(p) = \{\alpha_i(p)f_i^{l+1}(p)\}, i \in B^{l+1}, \quad (4)$$

де

$\alpha_i^{l+1}(p)$ - коефіцієнт впливу локального розв'язання на глобальне;

$f_i^{l+1}(p)$ - значення локального інтерполянту;

B^{l+1} - множина піддоменів нижчого рівня, які беруть участь у розрахунку.

Вплив на глобальне розв'язання може здійснювати як локальний інтерполянт піддомену, у якому знаходиться точка P , так і локальні інтерполянти піддоменів, що є сусідніми (тобто мають того ж домена предка і розташовані на одному рівні з вказаним).

У зв'язку з цим, на відмінність від відомих методів, у даному випадку вводиться два варіанти функції відстані $\beta_{int}(p)$ та $\beta_{ext}(p)$. Для забезпечення умови $\sum_{i \in B^{l+1}} a_i(p) = 1$, коефіцієнти повинні бути нормованими, і обчислюватимуться за формулою:

$$\alpha_i(p) = \frac{\beta_{int_i}(p)}{\beta_{int_i}(p) + \sum_{j \in B^{l+1} - b_i} \beta_{ext_j}(p)}. \quad (5)$$

У загальному випадку функції відстані повинні задовольняти наступним умовам:

$$\beta_{int}(p) = \begin{cases} 0 & , P \text{ лежить на межі піддомену} \\ 1 & , P \text{ лежить у центрі піддомену} \end{cases} \quad (6)$$

$$\beta_{ext}(p) = \begin{cases} 1 & , P \text{ лежить у центрі сусіднього піддомену} \\ 0 & , P \text{ лежить на межі даного піддомену, що найбільше} \\ & \text{віддалена від сусіднього піддомену} \end{cases} \quad (7)$$

Функція безперервності C повинна забезпечувати безперервність розв'язання (тобто відсутність точок розриву першого роду) на межах доменів. Для цього у відомих алгоритмах використовується, в залежності від мірності простору, функція згасання [3, 5]. У

модифікованому алгоритмі пропонується поєднувати частини локальних інтерполянтів за допомогою наступної функції:

$$C(F^{l+1}(p)) = \frac{1 + \sum_{i \in B^{l+1}} f_i^{l+1}(p)}{|B^{l+1}|}. \quad (8)$$

Загальний вид модифікованих алгоритмів декомпозиції та обчислення глобального інтерполянта можна записати наступним чином:

Декомпозиція (P, B^0)

{ $n = |P|$

якщо $n > M$ то

{

розрахунок параметрів $a^{l+1}, c_1^{l+1}, \dots, c_8^{l+1}$

визначення підмножин $P_1^{l+1}, \dots, P_8^{l+1}$

формування множини B^{l+1}

для $b_i \in B^{l+1}$

Декомпозиція (P_i^{l+1}, b_i)

}

інакше

Локальне розв'язання RBF-системи для множини P

}

Обчислення (P, B^l)

{ якщо $p \notin B^l$ то

повернути 0

якщо l - це лист дерева, то

повернути $f_{rbf}(p)$

інакше

{

визначення $\beta_{int}(p), \beta_{ext_1}(p), \dots, \beta_{ext_k}(p)$

для $b_i \in B^{l+1}$

```

    {
    визначення  $\alpha_i(p)$ 
     $f_i^{l+1}(p) = \alpha_i * \text{Обчислення}(P, b_i)$ 
    }
    формування множини  $F^{l+1}$ 
    повернути  $C(F^{l+1})$ 
    }
}

```

Вхідними даними алгоритму декомпозиції є множина точок P та поточний домен B , що містить дану множину. M - це максимальна кількість точок, яка може бути у довільному домені після декомпозиції. Даний параметр може визначатися розміром системи, яка може бути локально розв'язана при послідовному або паралельному способі обчислення.

Дослідження впливу виду функції відстані на похибку методу.

При розробці модифікованого алгоритму виникає проблема вибору виду функції відстані β . Враховуючи умови (6), (7), схематично функцію відстані можна зобразити так: див. рис. 2.

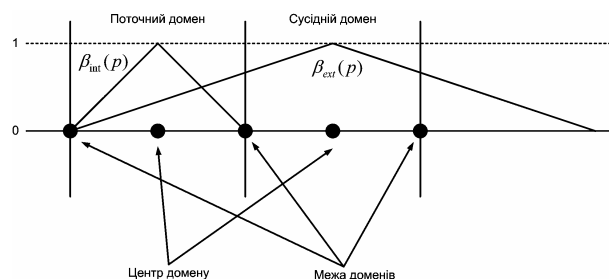


Рисунок 2 – Схематичне зображення функцій відстані

Як зрозуміло з умов, які висуваються до функції β_{int} , функція повинна характеризувати вплив локального інтерполянту на значення у точці P в залежності від її розташування у домені. Чим більше точка віддалена від центру інтерполяції (центру домену), тим меншим є вплив локального інтерполянта. Межевими випадками є розташування точки P в центрі інтерполяції (центрі домену) та розташування на межі домену.

У першому випадку вплив максимальний $\beta_{int}(p) = 1$, у другому – мінімальний $\beta_{int}(p) = 0$. Аналогічні доводи можна навести і для функції β_{ext} . Відмінність полягає у тому, що інтерполянт сусіднього домену найменше впливає у випадку, коли точка P найбільше віддалена від його центру інтерполяції (центру домену), тобто знаходиться на

протилежній межі поточного домену. Найбільший вплив спостерігається при розташуванні точки P у центрі інтерполяції сусіднього домену (центрі домену). У попередніх твердженнях поняття „центр інтерполяції” та „центр домену” ототожнюються. У реальних випадках це не так, оскільки кожен домен може містити кілька центрів інтерполяції (не більше за $Thresh$) і ці центри можуть довільно скупчуватися у різних місцях домену. У запропонованому методі відправну точку для обчислення функції відстані пропонується обирати як центр мас центрів інтерполяції домену.

Розглянемо можливі варіанти вибору функції відстані з точки зору зменшення похибки при визначенні глобального розв’язання на основі локальних. У якості функції відстані будемо розглядати наступні варіанти: див. рис. 3 (на прикладі функції β_{int}).

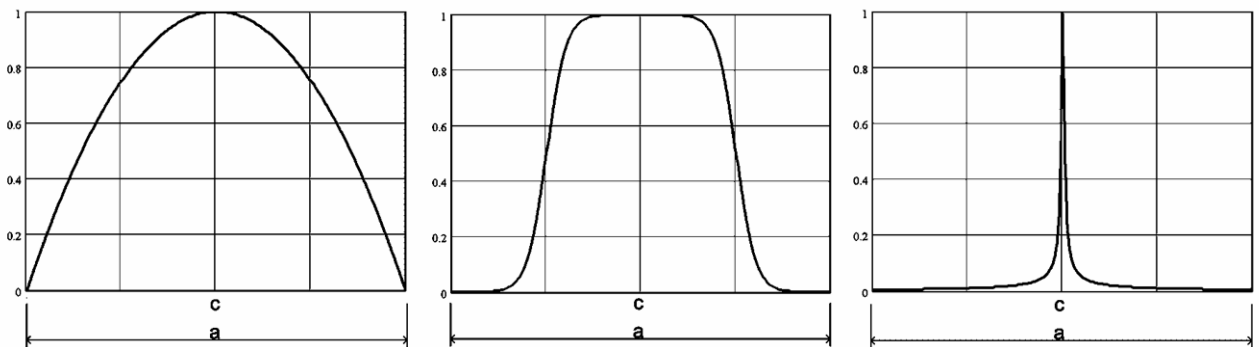


Рисунок 3 – Функції відстані

а) квадратична, б) експоненційна, в) ІМQ

Залежності, вказані на рис. 3, математично можуть бути виражені, наприклад, наступним чином:

$$f1(p) = \frac{4(p - c + \frac{a}{2})(c + \frac{a}{2} - p)}{a^2}, \quad (9)$$

де

c - центр домену;

a - сторона домену.

$$f2(p) = \frac{1}{1 + e^{2(|p-c|-\frac{a}{4})}} \quad (10)$$

$$f_3(p) = \frac{1}{\sqrt{1+k^2(p-c)^2}}, \quad (11)$$

де

k - коефіцієнт згладжування.

Слід зазначити, що функція (9) – це квадратична функція відстані, запропонована в [3]. Функція (11) – це інверсний мультіквадрик (ІМҚ). При використанні функції (11) параметр k обчислюється за формулою:

$$k = c - ce, \quad (12)$$

де

c - центр поточного домену;

ce - центр сусіднього домену.

Такий вибір коефіцієнту згладжування забезпечує умову: $p \rightarrow c \pm \frac{a}{2} \Rightarrow f_3(p) \rightarrow 0$.

Розглянемо, як впливає вибір функції відстані на похибку при розрахунку глобального інтерполянта. Для цього розглянемо узагальнену ситуацію, коли множина центрів інтерполяції розбивається на два домени: $D_1 : c_1, c_2, \dots, c_m$ та $D_2 : c_{m+1}, c_{m+2}, \dots, c_N$, де N - загальна кількість центрів інтерполяції. Запишемо глобальний інтерполянт, який можна отримати прямим методом RBF (F_1) та глобальний інтерполянт, отриманий за допомогою модифікованого алгоритму (F_2):

$$F_1(p) = \sum_{i=1}^N \lambda_i \varphi(p, c_i)$$

$$F_2(p) = \frac{F'(p) + F''(p) + 1}{2}, \quad (13)$$

де

$F'(p), F''(p)$ - локальні інтерполянти з урахуванням вагових коефіцієнтів. Запишемо вирази для локальних інтерполянтів:

$$F'(p) = \alpha_1(p) f'(p) = \alpha_1(p) \sum_{i=1}^m \lambda'_i \varphi(p, c'_i),$$

$$F''(p) = \alpha_2(p) f''(p) = \alpha_2(p) \sum_{i=m+1}^N \lambda''_i \varphi(p, c''_i). \quad (14)$$

Враховуючи, що при розрахунку глобального інтерполянта $F_1(p)$ порядку, у якому впорядковані точки c , значення не має, можна вважати, що всі центри інтерполяції згруповані у вигляді $c_1, \dots, c_m, c_{m+1}, \dots, c_N$. Завдяки такій впорядкованості можна стверджувати, що:

$$\begin{aligned}\varphi(p, c'_i) &= \phi(p, c_i), i \in [1, m], \\ \varphi(p, c''_i) &= \phi(p, c_i), i \in [m+1, N].\end{aligned}\tag{15}$$

Враховуючи вирази (13-15), запишемо значення абсолютної похибки при обчисленні глобального інтерполянта:

$$\Delta F = F_1 - F_2 = \frac{2 \sum_{i=1}^N \lambda_i \varphi(p, c_i) - \alpha_1(p) \sum_{i=1}^m \lambda'_i \varphi(p, c_i) - \alpha_2(p) \sum_{i=m+1}^N \lambda''_i \varphi(p, c_i) - 1}{2}$$

Оскільки мета вибору функції відстані – зменшення похибки, то $\Delta F \rightarrow 0$. Для цього верхня частина дробу повинна спрямовуватися до 0, а відповідно, його ліва частина до 1. Шляхом простих перетворень визначаємо, що для зменшення похибки повинна виконуватися умова:

$$\sum_{i=1}^m \varphi(p, c_i) (2\lambda_i - \alpha_1(p)\lambda'_i) + \sum_{i=m+1}^N \varphi(p, c_i) (2\lambda_i - \alpha_2(p)\lambda''_i) \rightarrow 1.\tag{16}$$

Можна бачити, що ліва частина виразу - це глобальний інтерполянт, отриманий за прямим методом RBF, і умова (16) буде виконуватися у наступних випадках:

$$\begin{aligned}2\lambda_i - \alpha_1(p)\lambda'_i &= \lambda_i, i \in [1, m], \\ 2\lambda_i - \alpha_2(p)\lambda''_i &= \lambda_i, i \in [m+1, N].\end{aligned}\tag{17}$$

Розглянемо на прикладі першої умови. Виразимо $\alpha_1(p) = \frac{\lambda_i}{\lambda'_i}$. З іншого боку, наприклад, для випадку двох центрів інтерполяції у двох доменах за формулою (5) можна записати:

$$\alpha_1(p) = \frac{\beta_{int_1}(p)}{\beta_{int_1}(p) + \beta_{ext_1}(p)}.\tag{18}$$

Права частина виразу (18) для випадку двох центрів інтерполяції у двох доменах може бути визначена наступним чином. λ_i - це коефіцієнти, що у даному випадку є розв'язанням системи рівнянь виду:

$$\begin{bmatrix} 1 & \varphi(c, c_e) \\ \varphi(c_e, c) & 1 \end{bmatrix} * \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}. \quad (19)$$

Розв'язуючи систему (19) отримуємо: $\lambda_1 = \lambda_2 = \frac{1 - \varphi(c, c_e)}{1 - \varphi^2(c, c_e)}$. Враховуючи, що $\varphi(c, c_e) = \sqrt{1 + (c - c_e)^2}$ і покладаючи $c - c_e = d$ отримуємо:

$$\lambda_1 = \lambda_2 = \frac{\sqrt{1 + d^2} - 1}{d^2}. \quad (20)$$

Розв'язуючи локальні системи, отримуємо: $\lambda'_1 = \lambda'_2 = 1$. Відповідно, вираз (20) буде правою частиною. Розглянемо, як буде визначатися похибка у визначенні виразу для $\alpha_1(p)$ в залежності від виду функції відстані.

Для квадратичної функції вираз (9) підставляємо у формулу (18) і зрівнюємо з виразом (20). У результаті отримуємо:

$$\Delta\alpha = \frac{(p - c + \frac{a}{2})(c + \frac{a}{2} - p)}{2cp - 2p^2 - c^2 + \frac{a^2}{2} + 2c_e p - c_e^2} - \frac{\sqrt{1 + d^2} - 1}{d^2}. \quad (21)$$

Аналогічно для експоненційної функції отримуємо:

$$\Delta\alpha = \frac{1 + e^{2(|p - c_e| - \frac{a}{4})}}{2 + e^{-\frac{a}{2}}(e^{2|p - c_e|} + e^{2|p - c|})} - \frac{\sqrt{1 + d^2} - 1}{d^2} \quad (22)$$

і для функції ІМQ отримуємо:

$$\Delta\alpha = \frac{\sqrt{1 + (p - c_e)^2}}{\sqrt{1 + (p - c_e)^2} + \sqrt{1 + (c - c_e)^2}(p - c)} - \frac{\sqrt{1 + d^2} - 1}{d^2}. \quad (23)$$

Проаналізуємо вирази (21-23). У даних виразах: P - поточна точка домену, c - центр інтерполяції поточного домену, c_e - центр інтерполяції сусіднього домену, a - сторона домену, d - відстань між центрами інтерполяції ($c - c_e$).

Якщо домен має сторону a , то точка P може лежати у межах $[c..c_e]$, а параметр d може змінюватися у межах $[0..2a]$. Побудуємо графіки залежності похибки визначення α від P та d , зафіксувавши c та a (див. рис. 4).

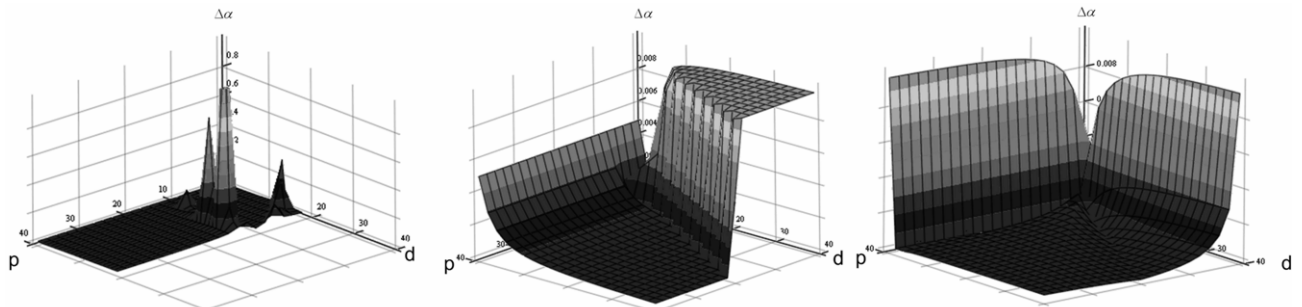


Рисунок 4 – Залежності абсолютної похибки $\Delta\alpha$ від розташування шуканої точки та відстані між центрами інтерполяції для функції відстані
а) квадратичної, б) експоненційної, в) ІМQ

Проаналізувавши рис. 4, можна зробити наступні висновки. При використанні квадратичної функції відстані абсолютне значення похибки при $p \in [c, c_e]$ і $d \in [0, 2a]$ не перевищує 0.08. Оскільки максимальне значення $\alpha=1$, то це відповідає відносній похибці 8%. Крім того, для певних комбінацій параметрів P та d визначення значення функції унеможлиблюється, тому що доводиться виконувати ділення на нуль. При використанні експоненційної функції відстані абсолютне значення похибки при $p \in [c, c_e]$ і $d \in [0, 2a]$ не перевищує 0.01, це відповідає відносній похибці 1%. При цьому найгіршим випадком є ситуація, коли $d \rightarrow 2a$ і $p \rightarrow c$. При використанні функції відстані ІМQ абсолютне значення похибки при $p \in [c, c_e]$ і $d \in [0, 2a]$ не перевищує 0.01, це відповідає відносній похибці 1%. При цьому існує два найгірші випадки: коли $d \rightarrow 2a$ і $p \rightarrow c$ та $d \rightarrow 0$ і $p \rightarrow c_e$. Такими чином для модифікованого методу використання квадратичної функції відстані неможливе. Якщо порівнювати експоненційну та функцію ІМQ, то можна спостерігати, що діапазон значень P та d у ситуації, коли $d \rightarrow 2a$ і $p \rightarrow c$, дає менші значення помилки у функції ІМQ ніж у експоненційної (див. рис. 4б та 4в). Порівняємо мінімальне значення помилки, максимальне та середнє для експоненційної функції та ІМQ (див. табл. 1). З таблиці 1 видно, що з точки зору мінімальної похибки, більш придатною є функція ІМQ.

Таблиця 1 – Порівняння похибок для експоненційної функції та ІМQ

	Експоненційна, %	ІМQ, %
$\Delta\alpha_{\min}$	0.00361	0
$\Delta\alpha_{\max}$	0.98	0.98
$\Delta\alpha_{avg}$	0.31	0.09163

Експериментальне дослідження модифікованого методу.

Для визначення ефективності модифікованого методу був виконаний обчислювальний експеримент, у якому реконструкція одного й того ж об'єкту (Stanford Bunny [7]) здійснювалась для різної кількості проєкційних точок трьома методами: прямий RBF з використанням компактних функцій [8] (який має кращі часові характеристики серед послідовних методів, як показано у [2]), ієрархічний RBF [3], модифікований алгоритм. Слід зазначити, що прямий RBF є послідовним алгоритмом, що не передбачає розпаралелювання. Можливість паралельного розв'язання систем у методах, заснованих на декомпозиції, імітувалась їх послідовним розв'язанням з наступним визначенням часу за формулою:

$$t_{solve} = \max(t_{solve_i}), \quad (24)$$

де

$$i \in [1..M];$$

t_{solve_i} - час розв'язання i -ої системи;

M - кількість систем, що розв'язується одночасно.

В експерименті порівнювалися часові витрати на трьох етапах: побудова матриці (декомпозиція), розв'язання систем, обчислення поверхні. Результати експерименту наведено на рис. 5.

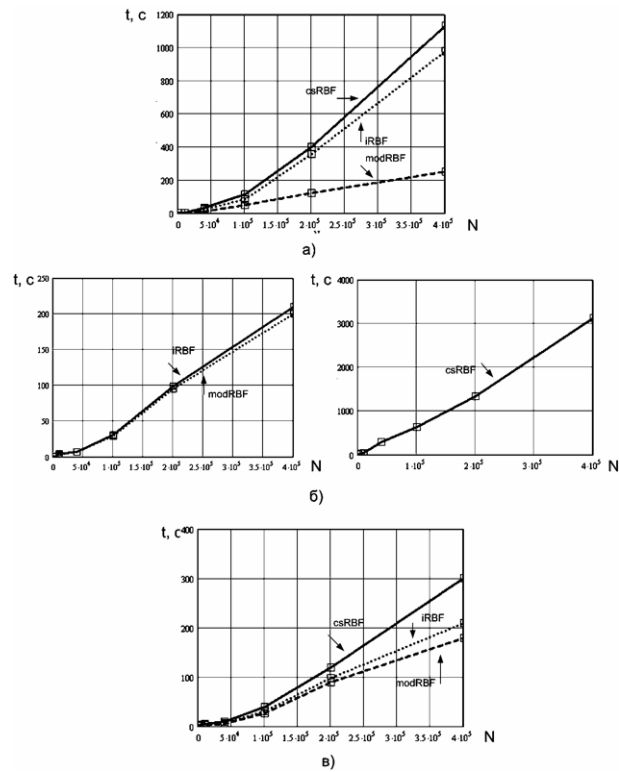


Рисунок 5 – Результати експериментальних досліджень

а) t_{decomp} - час декомпозиції та побудови матриць, б) t_{solve} - час розв'язання систем,

в) t_{eval} - час обчислення поверхні

Як видно з рисунку 5, на етапі декомпозиції модифікований метод показує значний вигравш у часі у порівнянні з методами компактної та ієрархічної RBF. Це можна пояснити відсутністю при декомпозиції процедури впорядкування точок. На рис. 5б час розв'язання послідовного методу компактної RBF винесено окремо, оскільки послідовний метод має значно гірші характеристики, у порівнянні з паралельним розв'язанням систем в інших методах. Як можна бачити, при приблизно однаковій кількості точок у доменах ієрархічний та модифікований метод мають приблизно однакові характеристики. На етапі обчислення поверхні (рис. 5в) модифікований метод має менші часові витрати. Це можна пояснити регулярною структурою дерева, яке утворюється при декомпозиції за модифікованим методом.

Висновки.

В результаті проведених досліджень розроблено модифікований метод побудови поверхонь у тривимірному просторі, який може бути використаний для створення тривимірних моделей реальних об'єктів на основі проєкційних даних. Дослідження показали, що модифікований метод має кращі часові характеристики, ніж метод-прототип завдяки відсутності процедури впорядкування точок, регулярній структурі дерева та

паралельній реалізації. На практиці запропонований метод використано у програмній системі для 3D-реконструкції у медичній практиці [9].

Бачиться, що більшого ефекту, з точки зору зменшення часових витрат, можна досягти при паралельній реалізації етапів декомпозиції та обчислення поверхні, що є подальшим напрямком дослідження.

Перелік посилань

1. Мельников С.Р. Лазерное сканирование: новый метод создания трехмерных моделей местности и инженерных объектов / С.Р. Мельников // Информационный бюллетень „ГИС-Ассоциации”, – СПб: Санкт-Петербургское общество геодезии и картографии, №2(29)–3(30), 2001. – С. 35–37.
2. Башков Е.А. Исследование возможностей применения RBF-алгоритма и его модификаций для построения поверхностных компьютерных моделей в медицинской практике / Е.А. Башков, В.С. Бабков // Сборник трудов международной конференции "Моделирование-2008", 14-16 мая 2008 г. – Киев: Институт проблем моделирования в энергетике им. Г.Е. Пухова, т. 1, 2008. - С. 166–171.
3. Pouderox J. Adaptive hierarchical RBF interpolation for creating smooth digital elevation models / J. Pouderox et al. // Proc. 12-th ACM Int. Symp. Advances in Geographical information Systems 2004. – ACP Press, 2004. – P. 232–240.
4. Бабков В.С. Реконструкція 3D-моделей реальних об'єктів методом RBF з використанням GPU / В.С. Бабков // Наукові праці Донецького національного технічного університету. Серія: “Інформатика, кібернетика та обчислювальна техніка”, випуск 9 (132). – Донецьк: ДонНТУ. 2008. – С. 132-136.
5. Qiang W. Surface rendering for parallel slice of contours from medical imaging / [W. Qiang, Z. Pan, C. Chun, B. Jiajun] // Computing in science & engineering, January-February 2007, V. 9, №1, 2007. – P. 32–37.
6. Tobor I. Multi-scale reconstruction of implicit surfaces with attributes from large unrecognized point set / I. Tobor, P. Reuter, C. Schlick // Proc. Shape Modelling Int. 2004. – IEEE CS press, 2004. – P. 67–76.
7. Level of Detail for 3D Graphics [Електронний ресурс] / Martin Reddy // 2008. – Режим доступу до ресурсу: <http://lodbook.com/models/>.
8. Kojekine N. Software Tools Using CSRBFs for Processing Scattered Data / N. Kojekine, I. Hagiwara, V. Savchenko // Computers & Graphics, April 2003, V. 27, №2, 2003. – P. 311–319.
9. Бабков В.С. Проектирование многофункциональной программной системы для реконструкции трехмерных объектов в медицинской практике / В.С. Бабков, Е.В. Ивашковец

// Сборник трудов третьей международной научно-технической конференции молодых ученых и студентов «Информатика и компьютерные технологии», 11-13 декабря 2007 г. – Донецк: Донецкий национальный технический университет, 2007. – С. 285–287.