УДК 621.39

**О. М. Ткаченко,** канд. техн. наук, доц.;
**О. Ф. Грійо Тукало,** асп.

# МЕТОД ШВИДКОГО ПОШУКУ НАЙБЛИЖЧОГО СУСІДА З ОБЧИСЛЕННЯМ ВІДСТАНІ ЗА ЗВАЖЕНОЮ ЕВКЛІДОВОЮ МЕТРИКОЮ

*Поставлено і розв'язано задачу підвищення продуктивності комп'ютерних систем для обробки мультимедійної, зокрема, голосової інформації за рахунок зменшення часу пошуку найближчого вектора у словнику великого розміру. Розроблено підхід на основі kd-дерева, що поєднує переваги методів швидкого пошуку найближчого сусіда із застосуванням зваженої евклідової метрики. Досягнуте зниження обчислювальної складності робить можливою реалізацію запропонованого методу на процесорах з обмеженою продуктивністю.*

## Вступ

Завдяки бурхливому розвитку методів запису і зберігання даних обсяги інформації, що передаються та обробляються в комп'ютерних системах, останнім часом суттєво зросли. Наразі Інтернет об'єднує сотні мільйонів серверів, на яких розміщені мільярди різних сайтів і окремих файлів з різноманітною інформацією. Слід зазначити, що в сучасних мережах основний обсяг трафіка припадає на мультимедійну, зокрема, аудіо- та мовленнєву інформацію. Обсяги даних настільки значні, що людина не в змозі проаналізувати їх самостійно, тому необхідність автоматизації процесів аналізу, зокрема, пошуку даних, є цілком очевидною. У теперішній час пошук має велике значення, він є основою розв'язання широкого кола задач, а саме: інформаційний пошук, ущільнення даних, розпізнавання і класифікація образів, кодування зображень, звукових даних тощо. Необхідність забезпечення функціонування у реальному масштабі часу накладає жорсткі вимоги до швидкодії мікропроцесорних пристроїв у складі систем обробки мовленнєвих сигналів. Таким чином, існує проблема недостатньої продуктивності комп'ютерних систем, призначених для обробки мультимедійної інформації. Підвищення швидкості пошуку даних дозволяє істотно підвищити ефективність роботи таких систем.

Задача пошуку найближчого сусіда полягає у знаходженні серед множини елементів, розташованих в багатовимірному метричному просторі, елементів, близьких до заданого, згідно з деякою функцією близькості. Зменшення обчислювальної складності пошуку найближчого сусіда в словниках великого розміру розглядалося в [1—3]. Застосування наведених в цих роботах методів дозволяє скоротити кількість операцій та, відповідно, час пошуку в 20—30 разів. Проте практична цінність цих методів суттєво обмежується неможливістю застосування зваженої евклідової метрики (ЗЕМ) [4], оскільки всі вони базуються на попередній обробці даних в словниках, коли значення ваг ще невідомі.

*Метою роботи* є підвищення продуктивності функціонування комп'ютерної системи за рахунок зменшення обчислювальної складності пошуку даних, що досягається завдяки реалізації нового підходу, який поєднує переваги методів швидкого пошуку найближчого сусіда із застосуванням ЗЕМ. Суть цього підходу полягає у тому, що на першому етапі на основі швидкого пошуку по *kd*-дереву за евклідовою метрикою (ЕМ) відбираються кандидати, з яких на другому етапі із використанням ЗЕМ відбирається один, найближчий до вхідного із наперед заданою ймовірністю $p^*$.

## Розробка методу швидкого пошуку найближчого вектора за ЗЕМ на основі *kd*-дерева

Нехай $\mathbf{K} = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_k\}$ — множина векторів, які містяться в словнику. Суть розробленої двоетапної стратегії пошуку, схематично показаної на рис. 1, полягає в тому, що:

1. На першому етапі виконується так званий швидкий пошук в упорядкованому певним чином словнику, в процесі якого за ЕМ відбирається множина $\mathbf{C} \subset \mathbf{K}$ векторів (кандидатів), упорядкова-

них за зростанням відстані до вхідного вектора $\mathbf{x}$.

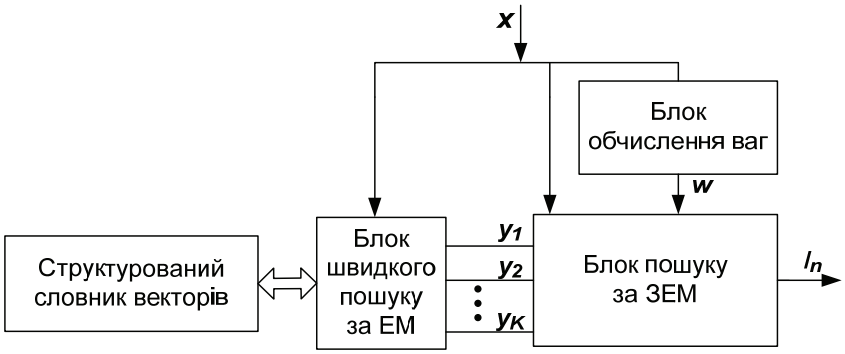



Рис. 1. Структурна схема двоетапної стратегії пошуку векторів у словнику

2. На другому етапі з використанням ЗЕМ з відібраної множини кандидатів обирається один вектор, найближчий до вхідного (із заданою ймовірністю $p^*$).

$$\mathbf{C} = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_t\}, |\mathbf{C}| = t, t \le k. \quad (1)$$

Оскільки за ЗЕМ розмірності мають різну вагу, вектор зі словника, який є найближчим за ЕМ, може виявитися не найкращим при врахуванні ваг. Таку ситуацію для двовимірного випадку показано на рис. 2.

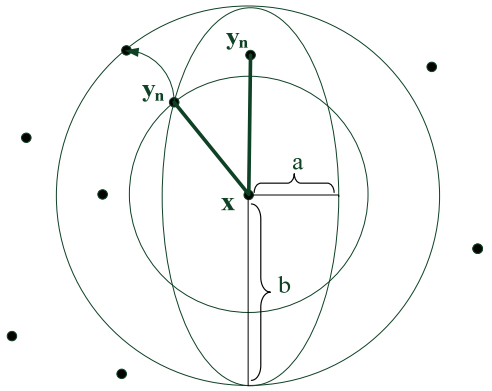



Рис. 2. Пошук найближчого сусіда за ЗЕМ в двовимірному просторі

Як видно з рис. 2, точка $\mathbf{y}_n$ (вектор в двовимірному просторі) є найближчою за ЕМ, але в результаті використання ЗЕМ найближчою виявляється точка $\mathbf{y}'_n$, оскільки для знаходження вектора, найближчого до вхідного за ЗЕМ, в процесі пошуку достатньо охопити множину кандидатів, обмежену колом радіуса $r_{\max} = \max(a, b)$, що детально розглянуто в [5].

Додаткове зменшення часу пошуку можна отримати, відмовившись від вимоги обов'язкового знаходження на другому етапі вектора, найближчого до вхідного за ЗЕМ. Передумови до цього такі:

1. Ймовірність того, що поточний вектор є найближчим за ЗЕМ, зменшується зі зростанням відстані за ЕМ від вектора-кандидата до вхідного. Таким чином, значна частина обчислень відстаней необхідна лише для того, щоб переконатися у відсутності кращого вектора.

2. Пропуск в деяких випадках найближчих векторів за ЗЕМ не приводить до помітного збільшення спектрального спотворення [4]. Пояснюється це тим, що замість найближчого вектора, як правило, вибирається вектор, досить близький до вхідного. Так, експериментальні дані [3] показують, що пропуск найближчих векторів в 5…10 % фреймів збільшують спектральне спотворення лише на 0,01…0,02 дБ.

Таким чином, немає необхідності обчислювати відстань за ЗЕМ до всіх векторів, що потрапляють в коло (гіперкулю) радіусом $r_{\max}$, а, задавши деяке значення ймовірності $p^*$, слід знайти величину $r^*$, що визначає радіус пошуку, в межах якого з ймовірністю $p \ge p^*$ знаходиться вектор,

найближчий до вхідного за ЗЕМ. Для цього необхідно знайти залежність $r = f(p, r_{\max})$ або $p = f\left(r/r_{\max}\right)$.

Ймовірність знаходження найближчого вектора на відстані *r* можна знайти як відношення частини об'єму *M*-вимірного еліпсоїда, обмеженого гіперкулею радіусу *r*, до всього об'єму *M*-вимірного еліпсоїда:

$$p(r) = \frac{V^{(M)}(r)}{V^{(M)}_{el}}; \quad p(r) = \begin{cases} 0, r = 0; \\ 1, r \ge r_{\max}. \end{cases} \tag{3}$$

Геометричну інтерпретацію наведеної залежності для двовимірного випадку показано на рис. 3.

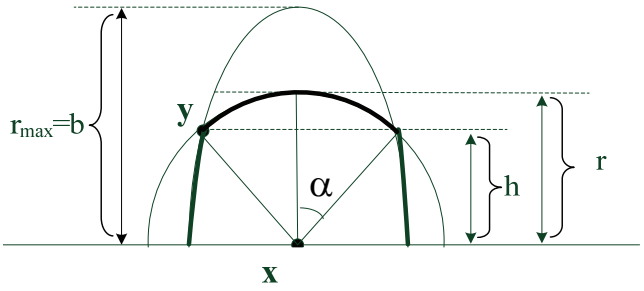



Рис. 3. Геометрична інтерпретація залежності ймовірності від відстані

Частина об'єму *M*-вимірного еліпсоїда, обмеженого гіперкулею радіусу *r* можна знайти як

$$V^{(M)}(h) = V^{(M)}_{el}(h) + \left(V_s^{(M)}(h) - V_c^{(M)}(h)\right), \tag{4}$$

де $V^{(M)}_{el}(h)$ — об'єм частини напівеліпса в *M*-вимірному просторі, обмеженого гіперплощиною, що проходить на відстані $h$;

$$V^{(M)}_{el}(h) = \int_0^h V^{(M-1)}_{el}(y)dy, \tag{5}$$

Коли $h = b$, де $b$ — максимальна піввісь еліпса, формула (5) дозволить отримати повний об'єм напівеліпса.

$V_s^{(M)}(h)$ — об'єм *M*-вимірного сектора радіусом *r* і кутом $2\alpha$;

$$V_s^{(M)}(h) = \frac{2\pi r^2}{M} V_s^{(M-2)} - \frac{(M-1) r \sin^{M-3}\alpha \cdot \cos\alpha}{M(M-2)} V_b^{(M-1)}, \tag{6}$$

де $V_b^{(M-1)} = \frac{2\pi r^2}{M} \cdot V_b^{(M-3)}$ — об'єм гіперкулі розмірності (*M* – 1); $V_c^{(M)}(h)$ — об'єм *M*-вимірного конуса висотою *h*.

$$V_c^{(M)}(h) = \frac{V_b^{(M-1)} \cdot h}{M}. \tag{7}$$

Для випадку *M* = 5 (розбиття 10-мірного вектора LSF на два підвектора) формули (5)—(7) набувають вигляду

$$V^{(5)}_{el}(h) = \int_0^h \pi^2 abcd\left(1 - \frac{y^2}{e^2}\right)dy = \frac{\pi^2 abcd}{2}\left(h - \frac{2}{3e^2}h^3 + \frac{1}{5e^4}h^5\right); \tag{8}$$

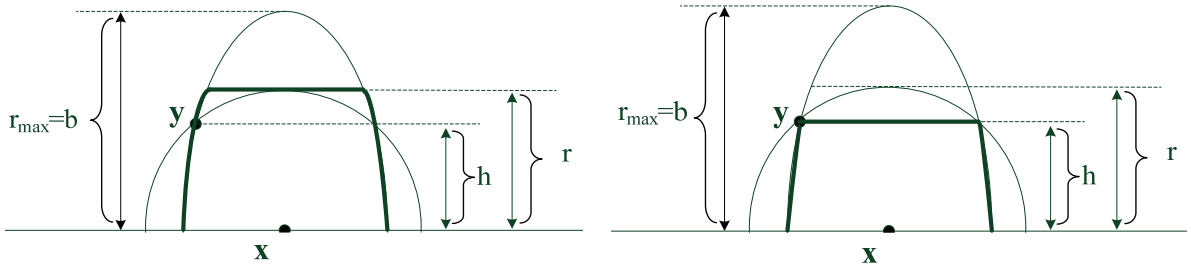
$$V^{(5)}_{el} = \frac{4}{15}\pi^2 abcde; \tag{9}$$

$$V_s^{(5)}(h)=\frac{2\pi^2 r^5}{15}\left(2-3\frac{h}{r}+\left(\frac{h}{r}\right)^3\right)4\,; \tag{10}$$

$$V_c^{(5)}(h)=\frac{\pi^2\left(r^2-h^2\right)^2 h}{10}, \tag{11}$$

де $h=\sqrt{\frac{e^2\left(r^2-a^2\right)}{e^2-a^2}}=r\sqrt{\frac{1-\left(\frac{a}{r}\right)^2}{1-\left(\frac{a}{e}\right)^2}}$; $a<b<c<d<e$ — напівосі еліпсоїда.

Підстановка (8)—(11) у формули (3) і (4) дає можливість отримати $p=f(r,a,b,c,d,e)$, однак через високу обчислювальну складність ця залежність має лише теоретичне значення. На практиці граничні радіуси пошуку найближчого вектора за заданим значенням ймовірності $p^*$ зручно визначити для областей, показаних на рис. 4.




а б

Рис. 4. Геометрична інтерпретація залежності ймовірності від відстані згідно з формулами (12) і (13)

В цьому випадку отримуємо явну залежність $p=f(r,r_{\max})$, задану формулами (12), (13):

$$p(r)=\frac{V^{(5)}(r)}{V_{el}^{(5)}}=\frac{15}{8}\cdot\frac{r}{r_{\max}}-\frac{5}{4}\cdot\left(\frac{r}{r_{\max}}\right)^3+\frac{3}{8}\cdot\left(\frac{r}{r_{\max}}\right)^5; \tag{12}$$

$$p(h)=\frac{V^{(5)}(h)}{V_{el}^{(5)}}=\frac{15}{8}\cdot\frac{h}{r_{\max}}-\frac{5}{4}\cdot\left(\frac{h}{r_{\max}}\right)^3+\frac{3}{8}\cdot\left(\frac{h}{r_{\max}}\right)^5. \tag{13}$$

Так, використовуючи співвідношення (12), можна знайти:

$$\left.\frac{r}{r_{\max}}\right|_{p^*=0,95}=0,707;\quad \left.\frac{r}{r_{\max}}\right|_{p^*=0,9}=0,621;\quad \left.\frac{r}{r_{\max}}\right|_{p^*=0,85}=0,506.$$

Це означає, що, наприклад, для визначення найближчого вектора в словнику за ЗЕМ з ймовірністю $p^*=0,95$ достатньо пройти лише 70 % відстані і т. д. Формули (12) і (13) визначають, відповідно, верхню і нижню межі ймовірності $p^*$, що випливає з їхньої геометричної інтерпретації (див. рис. 4).

Метод пошуку вектора зі словника, найближчого до вхідного за ЗЕМ із кандидатів $\mathbf{C}\subset\mathbf{K}$, впорядкованих за зростанням, відстані $r$ на другому етапі з заданою ймовірністю $p^*<1$ реалізується таким чином:

1. Визначаються вагові коефіцієнти вхідного вектора $w_k\geq 1,\ k=\overline{1,M}$.

2. Ініціалізація: $i=1,\ r_{max}=INF$.

3. За формулами обчислення ймовірності (12) або (13) обчислюється значення $\frac{r}{r_{\max}}$ для заданого значення ймовірності $p^*$.

4. Для *i*-го вектора зі списку кандидатів $i = \overline{1,|\mathbf{K}|}$ обчислюється відстань за ЗЕМ $r_i^{(WE)}$:

$$r_i^{(WE)} = \sum_{j=1}^{M} \left[ w_j \left( x_j - y_j \right) \right]^2 ; \tag{14}$$

5. Якщо $\frac{r_i^{(E)}}{r_{\max}} \geq p^*$, де $r_i^{(E)} = \sum_{j=1}^{M} r_j^{(E)}$, $r_j^{(E)} = \left( x_j - y_j \right)^2$; пошук завершується, $|C| = i$. Якщо $r_i^{(WE)} < r_{\max}$, присвоюється $r_{\max} = r_i^{(WE)}$.

6. $i = i + 1$ перехід до п. 4.

Для виконання швидкого пошуку векторів у словнику на першому етапі вектори було упорядковано на основі *kd*-дерева (*k*-вимірне дерево). *kd*-дерево — це бінарне дерево (БД), в якому кожна вершина задає розбиття простору на два підпростори деякою площиною, що проходить через неї [6]. У *kd*-дереві, крім кореневої, присутні два типи вершин: термінальні та нетермінальні (вузли).

Математично задача пошуку найближчого сусіда по *kd*-дереву формулюється так: дано деякий *M*-вимірний вектор; необхідно знайти вершину *kd*-дерева $v'$, щоб виконувалась умова

$$r^E \left( v', v \right) = \min \left\{ r^E \left( v_i, v \right) \right\}, \; i = \left( \overline{1,n} \right), \; v_i \in \mathbf{V} , \tag{15}$$

де $v$ — вершина, для якої виконується пошук найближчого вектора; $v'$ — вершина, відстань до якої найменша.

Використання *kd*-дерев дозволяє суттєво зменшити кількість вимірювань, необхідних для пошуку найближчого сусіднього вектора (в середньому $\log_2 n$ замість $n$ при повному пошуку). Разом з тим зростає похибка квантування і, відповідно, спектральне спотворення, оскільки пошук по *kd*-дереву не гарантує знаходження дійсно найближчого вектора згідно з формулою (15) [6]. В роботі [7] запропоновано удосконалену процедуру пошуку по *kd*-дереву, вперше введену для кодування зображень, яку пропонується застосувати для пошуку найближчого вектора у словнику в процесі кодування мовлення з урахуванням специфіки мовленнєвих сигналів. Ілюстрацію пошуку найближчого вектора у словнику, впорядкованої на основі *kd*-дерева, для двовимірного випадку показано на рис. 5.

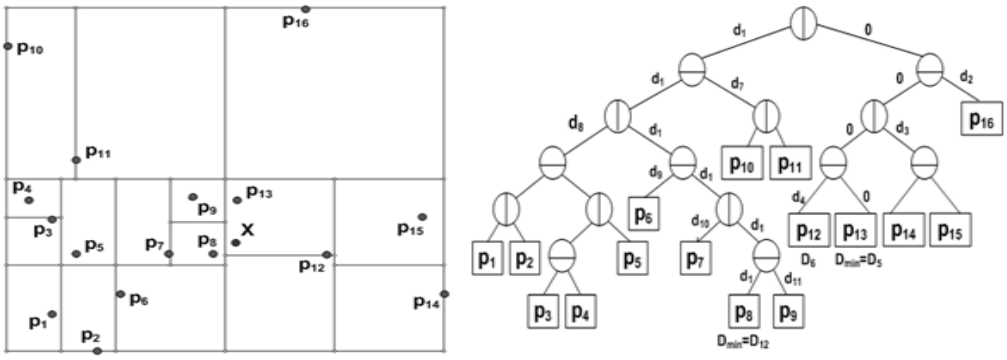



Рис. 5. Ілюстрація пошуку найближчого вектора по *kd*-дереву для двовимірного випадку

Щоб забезпечити знаходження найближчого вектора, пошук, крім прямої фази пошуку (спуску по дереву), повинен мати також обернену. Під час прямого пошуку фіксуються всі відстані до вузлів $d_i$. Пряма фаза завершується обчисленням відстані до відповідної термінальної вершини $D_k = D_{\min}$. Після цього починається обернена фаза пошуку, при цьому обчислюються відстані $D_k$ лише до тих вершин дерева, які можуть забезпечити виконання $D_k < D_{\min}$. Якщо ця умова виконується, вважають, що $D_{\min} = D_k$ [6].

Наведена процедура пошуку забезпечує отримання не одного найближчого вектора, а деякої

множини, упорядкованих за зростанням відстані згідно з формулою (1). Це можливо за рахунок того, що дані про відстань до вже пройдених термінальних вершин зберігаються, і одночасно здійснюється їх упорядкування за зростанням відстані до вхідного вектора. Завдяки цьому додаткове знаходження декількох найближчих векторів не вимагає багато часу.

## Експериментальні результати

Для експериментального дослідження двоетапної стратегії пошуку було використано загальнодоступну частину англомовного акустичного корпусу TIMIT. Тренувальна послідовність складалася з 90000 векторів LSF, отриманих на основі моделі лінійного прогнозування десятого порядку [8]. Тестова послідовність складалася з 15000 векторів LSF, відмінних від векторів тренувальної послідовності. Довжина фрейму становила 20 мс. Відстань вимірювалась за ЗЕМ з використанням ваг розрядів, обчислених за спектральною чутливістю [9]. Розмір словника становив 4096 векторів розмірністю $M = 5$ .

На рис. 6 наведено результати перевірки відповідності експериментальним результатам запропонованих математичних співвідношень (4)—(6) для оцінювання ймовірності знаходження найближчого вектора в словнику за ЗЕМ на відстані $r$. Результати усереднювалися для 10720 фреймів.

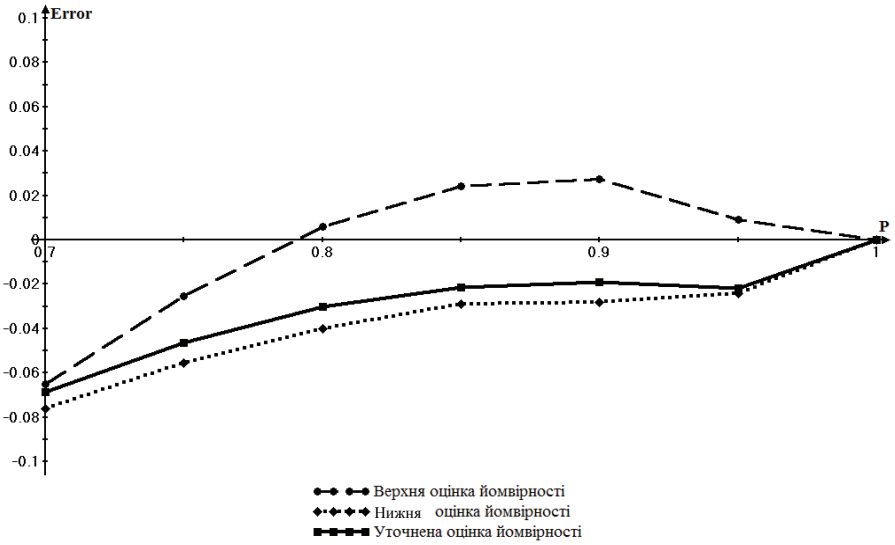



Рис. 6. Похибка оцінювання ймовірності

На рис. 6 похибка визначається за формулою

$$Error = p_{expr} - p_{teor},$$

де $p_{teor}$ — значення ймовірностей, обчислені на основі запропонованих математичних співвідношень (4)—(6); $p_{expr}$ — значення ймовірностей, отримані експериментально.

Як видно з наведеного рис. 6, для великих значень ймовірності, які є найважливішими з огляду на практичне застосування, всі теоретичні оцінки в достатній мірі збігаються з результатами, отриманими експериментально (похибка менша за 0,02).

Оцінювання ефективності двоетапної стратегії пошуку найближчого вектора із заданою ймовірністю здійснювалося за спектральним спотворенням та кількістю операцій, що виконуються в процесі пошуку. Отримані результати наведено в таблиці.

**Продуктивність двоетапного пошуку найближчого вектора за ЗЕМ на основі *kd*-дерева**

| Задане значення ймовірності вибору найближчого вектора в словнику, $p^*$ | | 1 | 0,95 | 0,9 | 0,85 |
|---|---|---|---|---|---|
| Спектральне спотворення, SD (дБ) | | 1,18 | 1,19 | 1,20 | 1,21 |
| Кількість векторів зі словника, до яких необхідно обчислити відстань, $\|C\| = \|C_1\| + \|C_2\|$ | Словник1: $\|C_1\|$ | 11,9 | 2,7 | 1,8 | 1,4 |
| | Словник2: $\|C_2\|$ | 37,9 | 12,9 | 8,2 | 5,8 |
| Загальна кількість операцій під час пошуку ($M = 5 \times 2$, $n = 4096$) | I етап | 6870 | 3270 | 2070 | 1690 |
| | II етап | 1096 | 343 | 220 | 158 |

У таблиці кількість виконаних під час пошуку операцій $N$ оцінювалась таким чином:

— повний пошук за ЗЕМ (обчислення відстані до всіх векторів в словнику згідно з (14)):

$$N = 4 \cdot M \cdot n = 4 \cdot 10 \cdot 4096 = 163840 \text{ (операцій);}$$

— двоетапний пошук на основі *kd*-дерева:

$$N = N_1 + N_2 = \left(N_1' + N_1''\right) + N_2,$$

де $N_1'$ — кількість операцій під час пошуку одного найближчого за ЕМ вектора в словнику на основі *kd*-дерева; $N_1''$ — кількість операцій під час пошуку додатково $|\mathbf{K} - 1|$ векторів в словнику по *kd*-дереву; $N_2 = (4 \cdot M + 2) \cdot |\mathbf{C}|$, $|\mathbf{C}| = |\mathbf{C_1}| + |\mathbf{C_2}|$ — кількість операцій при виборі найближчого за ЗЕМ вектора зі словника серед $|\mathbf{C}|$ кандидатів.

Таким чином, зниження обчислювальної складності робить можливою реалізацію запропонованого методу на процесорах з обмеженою продуктивністю.

## Висновки

Запропонована в статті двоетапна стратегія пошуку векторів у словниках великих розмірів дозволяє поєднати переваги методу швидкого пошуку найближчого сусіда на основі *kd*-дерева із застосуванням зваженої евклідової метрики. Для ймовірності вибору найближчого сусіда $p^* = 0{,}95$ кількість операцій зменшується в 45,5 разів порівняно з повним перебором, що дозволяє реалізувати цей метод на процесорах з обмеженою продуктивністю.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Agrell E. Spectral coding by fast vector quantization / E. Agrell // Proc. IEEE Workshop on Speech Coding for Telecommunications. – Sainte-Adèle, Québec, Canada, 1993. — Pp. 61—62.

2. Arya S. Algorithms for fast vector quantization / S. Arya and D. M. Mount // In J. A. Storer and M. Cohn, editors, Proc. of DCC '93: Data Compression Conference, IEEE Press. — 1993. — P. 381—390. — ISBN 0-89871-329-3.

3. Zhou J. Simple Fast Vector Quantization of the Line Spectral Frequencies / Zhou J., Shoham Y., Akansu A. // Image Compression and Encryption Technologies. — 2001. — Vol. 4551. — P. 274—282.

4. Paliwal K. K. Efficient vector quantization of LPC parameters at 24 bits/frame / K. K. Paliwal, B. S. Atal // IEEE Transaction on Speech and Audio Processing. — 1993. — No. 2, vol. 1. — P. 3—14.

5. Ткаченко О. М. Двоетапна стратегія пошуку в векторних кодових книгах для ущільнення мовлення / О. М. Ткаченко, О. Ф. Грійо Тукало // Вісник Вінницького політехнічного інституту. — 2011. — № 3. — С. 194—201. — ISSN 1997-9266.

6. Ткаченко О. М. Пошук векторів у кодових книгах при ущільненні мовлення на основі бінарного дерева / О. М. Ткаченко, О. Ф. Грійо Тукало // Інформаційні технології та комп'ютерна інженерія. — 2011. — № 1. — С. 38—44. — ISSN 1999-9941.

7. Arya S. Algorithms for fast vector quantization / S. Arya and D. M. Mount // In J. A. Storer and M. Cohn, editors, Proc. of DCC '93: Data Compression Conference, IEEE Press. — 1993. — P. 381—390. — ISBN 0-89871-329-3.

8. Chu W. C. Speech Coding Algorithms: Foundation and Evolution of Standardized Coders / Wai C. Chu // NY. : John Wiley & Sons, Inc. — 2003. — 558 p. — ISBN 0-471-37312-5.

9. Hai Le Vu. Efficient Distance Measure for Quantization of LSF and Its Karhunen–Loeve Transformed Parameters / Hai Le Vu and Laszlo Lois // IEEE Transactions on speech and audio processing. — Nov. 2000. — No. 6, vol. 8.

Рекомендована кафедрою обчислювальної техніки

Стаття надійшла до редакції 27.12.12
Рекомендована до друку 10.01.13

***Ткаченко Олександр Миколайович*** — доцент, ***Грійо Тукало Оксана Франсисківна*** — аспірантка.
Кафедра обчислювальної техніки, Вінницький національний технічний університет, Вінниця