

УДК 621.39

Ткаченко О. М., канд. техн. наук
Грійо Тукало О. Ф.

ДОСЛІДЖЕННЯ МЕТОДІВ ШВИДКОГО ПОШУКУ ВЕКТОРІВ НА ОСНОВІ ДІАГРАМИ ВОРОНОГО

Вінницький національний технічний університет

Розглянуто методи швидкого пошуку векторів у кодових книгах. Проаналізовано основні фактори, що впливають на ефективність пошуку. Розроблено декілька модифікацій методу «сусіднього спуску» та проведено оцінювання їх ефективності

Вступ

У сучасних протоколах ущільнення мовленнєвих сигналів для передавання спектральної інформації використовують лінійні спектральні частоти (*LSF, line spectrum frequency*). При цьому широко застосовується векторне квантування *LSF*, яке реалізується за допомогою кодових книг. Проте практичне застосування векторного квантування у реальному масштабі часу обмежено через різке зростання витрат пам'яті та часу на пошук кодового вектора у кодовій книзі. Тому в діючих стандартах застосовують субоптимальне кодування, коли 10-вимірний вектор параметрів розбивають на 2 або більше підвекторів меншої розмірності [1].

З метою зменшення часу пошуку в [2] було запропоновано декілька підходів до впорядкування векторів у кодовій книзі, названих авторами методами швидкого векторного квантування (*fast vector quantization methods*). Було показано, що складність обчислень при застосуванні наведених методів складає порядка 25% від складності обчислень при повному пошуку без суттєвої втрати продуктивності, що оцінювалася по спектральному спотворенню.

В [3] було розроблено метод структуризації кодових книг на основі відношення мажорювання. Це дозволило зменшити складність обчислень до 16-20% від відповідної складності при повному пошуку.

Метод «сусіднього спуску» (*neighbor descent*), що його запропоновано в [4], дозволяє зменшити час пошуку до 2% від часу повного пошуку в описаному

прикладі. Проте цей високий показник було досягнуто для кодових книг конкретної розмірності $M = 5$ та певного розміру $N = 4096$. Крім того, даний результат було показано при істотному збільшенні середнього спектрального спотворення (на 0.22 дБ), що може виявитися непридатним для практичного застосування.

У даній роботі оцінювався вплив розмірності векторів і розміру кодових книг на ефективність пошуку за методом «сусіднього спуску». Запропоновано декілька модифікацій вказаного методу та проведено їх порівняльний аналіз. Показано, що їх застосування дозволяє зменшити обчислювальну складність пошуку найближчого вектора у кодовій книзі до 3% від обчислювальної складності повного пошуку без збільшення спектрального спотворення.

Мета та задачі досліджень

Метою роботи є зменшення складності обчислень при кодуванні мовленнєвих сигналів за рахунок швидкого пошуку найближчого вектора у кодовій книзі.

Для досягнення поставленої мети необхідно розв'язати такі задачі:

- проаналізувати фактори, що впливають на ефективність пошуку за методом «сусіднього спуску»;
- розробити модель і структуру даних для реалізації різних модифікацій метода «сусіднього спуску»;
- оцінити ефективність запропонованих методів.

Основна частина

Нехай кодова книга містить кінцеву множину векторів $Q = \{Y_1, Y_2, \dots, Y_N\}$,

$Y_i = (y_{i1}, y_{i2}, \dots, y_{iM})$. Таким чином, з кожним вектором Y_j у кодовій книзі пов'язаний індекс, або кодове слово j , що може бути записано як N – розрядне ціле число. На вхід квантизатора поступає вектор $X = (x_1, x_2, \dots, x_M)$. В результаті кодування необхідно вибрати таке кодове слово j , що мінімізує спотворення $d(X, Y_j)$ (правило вибору найближчого сусіднього вектора). У даній роботі для вимірювання спотворення використовувалася Евклідова відстань:

$$d^2(X, Y_i) = \sum_{k=1}^M (x_k - y_{ik})^2. \quad (1)$$

Множина вхідних векторів X , що кодуються одним і тим самим індексом j , утворюють регіон Вороного:

$$V_j = \{X : d(X, Y_j) \leq d(X, Y_i); \forall i \in I\}, \quad (2)$$

де $I = \{1, 2, \dots, N\}$ – множина індексів. Отримані таким чином N регіонів утворюють діаграму Вороного (ДВ). Альтернативним варіантом завдання регіону Вороного є

$$V_j = \{X : d(X, Y_j) \leq d(X, Y_i); \forall i \in A_j\}, \quad (3)$$

де $A_j = \{i : V_i \cap V_j \neq \emptyset\}$ – множина кодів слів, у яких регіони Вороного є суміжними (*adjacent*) з даним регіоном. На рис. 1 представлено графічну інтерпретацію регіонів Вороного для двовимірного випадку.

Найпростішим методом пошуку у кодовій книзі вектора, найближчого до X , є повний пошук. При цьому для кодової книги розміром N необхідно N разів обчислити відстань згідно (1), що у багатьох випадках є непридатним для практичного застосування. З іншого боку, при такому підході виконується багато зайвих обчислень, оскільки ніяк не враховуються результати попередніх вимірювань відстаней.

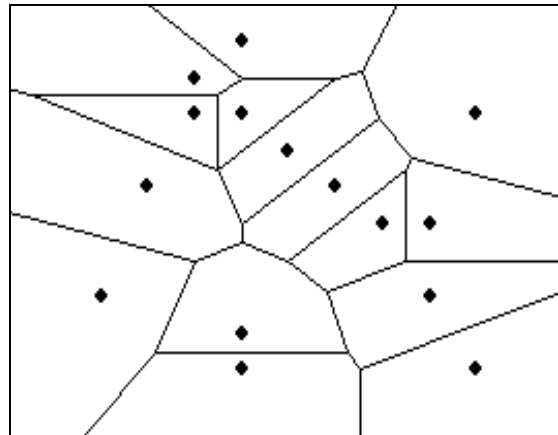


Рис. 1. Приклад діаграми Вороного для двовимірного випадку

У роботі [4] запропоновано метод “сусіднього спуску” (*neighbor descent*), який базується на використанні таблиць суміжності A_j , підготовлених заздалегідь для кожного регіону V_j . Основна ідея даного методу полягає у тому, що результати обчислення відстаней використовуються для переходів до суміжних регіонів у напрямку оптимуму. Перехід від кодового слова j до кодового слова l відбувається за умови

$$d(X, Y_l) \leq d(X, Y_j); l \in A_j. \quad (4)$$

Ілюстрацію даного підходу для двовимірного випадку наведено на рис. 2.

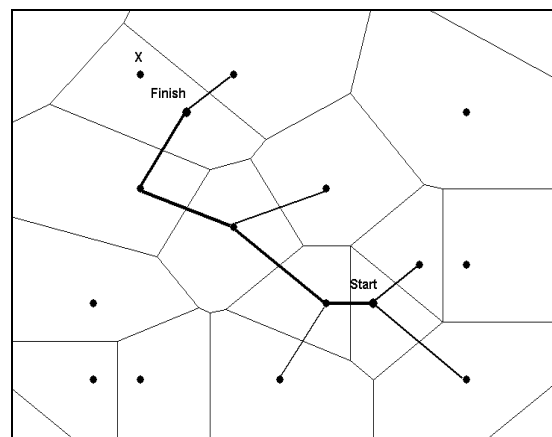


Рис. 2. Пошук найближчого вектора за методом “сусіднього спуску”

У [4] розглядалися дві основні модифікації методу “сусіднього спуску”. У першій з них (*full neighbor descent, FND*) спочатку обчислювалася відстань від вхідного вектора до усіх суміжних кодів

слів для даного регіону Вороного, після чого приймалося рішення про перехід до кодового слова з мінімальною відстанню або про завершення пошуку (повний обхід сусідів):

$$d(X, Y_l) \leq d(X, Y_i); \quad (5)$$

$$\forall i \in A_j, l \in A_j$$

Друга модифікація (*random neighbor descent, RND*) відрізнялася тим, що сусідні кодові слова перевірялися у випадковому порядку і перехід до сусіднього кодового слова відбувався одразу, як тільки виконувалася умова (4) (випадковий обхід сусідів).

Вищу продуктивність було отримано при випадковому обході сусідів. В обох випадках за початкову гіпотезу обиралося вихідне кодове слово, знайдене для попереднього фрейму.

Відомо, що у сучасних протоколах ущільнення застосовуються кодові книги різних розмірів, а також із різною розмірністю підвекторів. З огляду на це, важливо було дослідити, як змінюється ефекти-

вність пошуку із застосуванням ДВ під впливом вказаних факторів.

Побудова векторних кодових книг виконувалася за умов, описаних у [3]. Для тестування було відібрано текст, який містив уривок з роману І. Нечуй-Левицького і складався з 20000 векторів.

Результати дослідження впливу розмірності кодових книг M на ефективність пошуку за методом “сусіднього спуску” представлено у табл. 1 та проілюстровано на рис. 3. Розмір кодових книг N при цьому був фіксований і складав 1024 вектори. Оцінювання результатів проводилося за кількістю обчислень відстаней (КОВ) у кодовій книзі C , необхідною для пошуку найближчого вектора, вираженою у відсотках відносно розміру кодової книги N :

$$C, \% = \frac{C}{N} \cdot 100\% . \quad (6)$$

Крім того, у таблиці наведено середню кількість сусідів (СКС) для кожного регіону Вороного.

Таблиця 1. Вплив розмірності кодових книг на ефективність пошуку

Розмірність кодової книги M	СКС	КОВ C , %				Середня КОВ для кожного регіону n	
		Середня		Максимальна		FND	RND
		FND	RND	FND	RND		
3	14,38	3,69	2,53	17,97	7,71	10,49	4,46
4	30,36	5,79	4,34	16,60	9,77	23,25	10,02
5	57,48	10,10	7,68	30,76	20,51	43,08	17,40

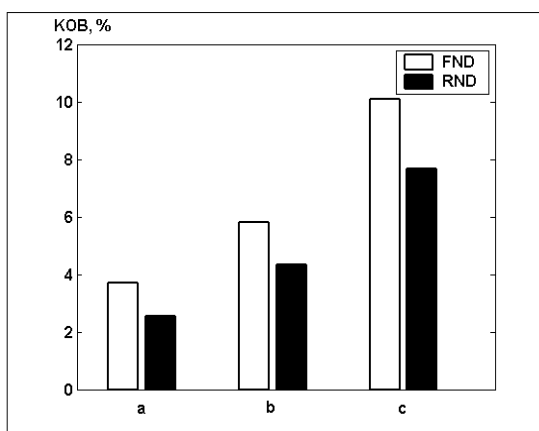


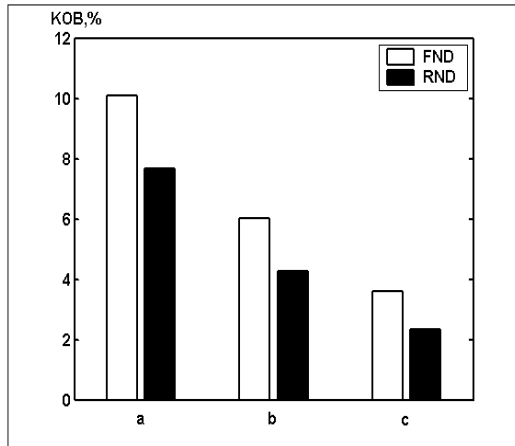
Рис. 3. Залежність між КОВ та розмірністю кодової книги ($a - 3, b - 4, c - 5$)

Як можна бачити з таблиці, із зростанням розмірності кодових книг збільшується СКС та середня КОВ для кожного регіону Вороного та відповідно втрачається ефективність пошуку за методом “сусіднього спуску”. Це пояснюється тим, що для переходу до суміжного регіону згідно (4) або (5) здійснюється витратний лінійний пошук, який власне і призводить до загального зниження ефективності.

У табл. 2 та на рис. 4 представлено результати дослідження впливу розміру кодових книг N на ефективність пошуку найближчого вектора. Результати отримано для кодових книг фіксованої розмірності $M = 5$.

Таблиця 2. Вплив розміру кодових книг на ефективність пошуку

Розмір кодової книги N	СКС	КОВ C , %				Середня КОВ для кожного регіону \bar{n}	
		Середня		Максимальна		FND	RND
		FND	RND	FND	RND		
1024	57,48	10,10	7,68	30,76	20,51	43,08	17,40
2048	62,57	6,03	4,28	19,04	10,01	47,47	17,21
4096	67,08	3,58	2,34	12,33	5,37	51,75	16,68

Рис. 4. Залежність між КОВ та розміром кодової книги (a – 1024, b – 2048, c – 4096)

Дані, наведені в таблиці, свідчать, що зі збільшенням розміру кодової книги N СКС та середня КОВ для кожного регіону (фактично чисельник у (6)) збільшуватимуться повільніше за знаменник. Отже, підвищення ефективності забезпечується тим, що відносна кількість сусідів зменшується (хоча абсолютні значення зростають), за рахунок чого менше часу витрачається на пошук сусіда для переходу згідно умов (4) або (5) для кожного кодового слова j , тобто зменшується час лінійного пошуку.

Порівняння методів FND і RND показує, що метод RND має перевагу за рахунок суттєвого скорочення середньої КОВ для кожного регіону \bar{n} при переході до сусіднього регіону.

Ефективність пошуку суттєво залежить від того, у якому порядку буде здійснюватися обхід сусідніх кодових слів. При цьому логічно зробити припущення, що впорядкувавши обхід векторів для кожного регіону можна ще зменшити час лінійного пошуку і тим самим досягти

подальшого зменшення складності обчислень.

Один з можливих підходів до впорядкування кодових книг базується на використанні теорії мажоризації [5], застосування якого для прискорення пошуку найближчого вектора описано в [3]. Нехай задано вектори $Y = (y_1, y_2, \dots, y_n)$ та $Y' = (y'_1, y'_2, \dots, y'_n)$, $Y, Y' \in Q$, компоненти яких впорядковано за незростанням. Говорять, що Y мажорується Y' або Y' мажорує Y (позначається $Y \prec Y'$), якщо виконується:

$$\sum_{i=1}^k y_i \leq \sum_{i=1}^k y'_i, \quad k = 1, 2, \dots, M-1, \quad (7)$$

$$\sum_{i=1}^M y_i = \sum_{i=1}^M y'_i.$$

Впорядкування, що задаються відношенням мажорування (7), називають мажоризацією.

Якщо виконується нерівність

$$\sum_{i=1}^k y_i \leq \sum_{i=1}^k y'_i, \quad k = 1, 2, \dots, M, \quad (8)$$

говорять, що Y слабо мажорується Y' або Y' слабо мажорує Y (позначається $Y \prec_w Y'$).

У даній роботі ми використаємо (8) для впорядкування векторів кодової книги. При цьому кодову книгу необхідно розбити на окремі класи, що формуються відповідно до рівнів мажоризації. Рівні мажоризації формуються за таким правилом. Будемо вважати, що рівень мажоризації L_i мажорується рівнем мажоризації L_k , якщо для кожного вектора Y , що на-

лежить L_i , на рівні L_k знайдеться вектор Y' , що слабо мажорує Y , або формально

$$\forall Y, Y \in L_i, \exists Y', Y' \in L_k, \quad (9) \\ Y \prec_w Y' \Rightarrow L_i \prec L_k;$$

Розбиття кодової книги згідно із заданим критерієм (9) будемо називати структуризацією. Структуризація має на меті скоротчення обчислювальних витрат на пошук для переходу від кодового слова j до кодового слова l згідно (4).

Пошук буде відбуватися таким чином.

1. Спочатку визначається, на який рівень мажоризації потрапляє вхідний вектор X .

2. Після цього, починаючи з цього рівня L_k , відбувається пошук вектора Y_l для переходу до кодового слова l згідно (4).

3. Якщо такий вектор не знаходиться на рівні L_k , пошук продовжується на сусідніх рівнях $L_{k-1}, L_{k+1}, L_{k-2}, L_{k+2}, \dots$

4. Якщо для кожного вектора з таблиці суміжності відстань не менше, ніж віж поточного, пошук завершується і поточний вектор Y_j вважається найближчим до вхідного вектора X . Інакше відбувається перехід, і пункти 1 – 4 повторюються для кодового слова l .

Зазначимо, що пошук рівня мажоризації для вхідного вектора X не потребує значних витрат часу і зводиться до простої процедури перевірки виконання (7).

В рамках даного підходу було проаналізовано два варіанти пошуку. Різниця між ними полягає у тому, що в першому з них (впорядкування за мажоризацією 1) пошук починався з рівня мажоризації, визначеного для вхідного вектора, а в другому (впорядкування за мажоризацією 2) так само, як в [4], за початкову гіпотезу обиралося вихідне кодове слово, знайдене для попереднього фрейму.

Крім того, було розглянуто впорядкування векторів за середнім [2], яке полягає в тому, що таблиці суміжності кожного вектора кодової книги впорядкову-

ються за зростанням середнього арифметичного (СА) їх координат.

Пошук безпосередньо здійснюватиметься таким чином:

1. Спочатку визначається середнє арифметичне координат вхідного вектора X .

2. Потім, після знаходження першого вектора, СА координат якого більше ніж X , відбувається пошук вектора Y_l для переходу до кодового слова l згідно (4).

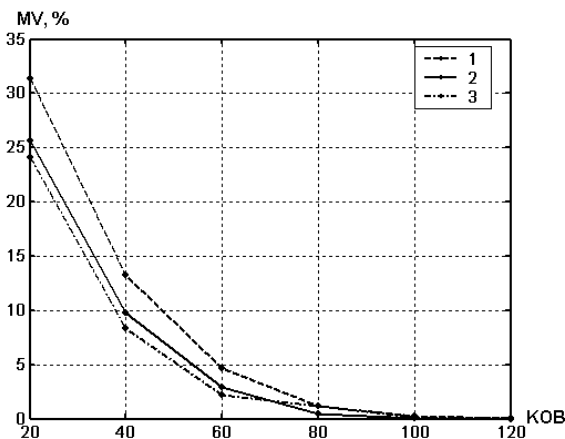
3. Якщо для кожного вектора з таблиці суміжності відстань не менше, ніж від поточного, пошук завершується, і поточний вектор Y_j вважається найближчим до вхідного вектора X . Інакше відбувається перехід, і пункти 1 – 3 повторюються для кодового слова l .

Перевага даного методу полягає в тому, що він не потребує додаткової пам'яті.

Як можна побачити з табл. 1 і 2, максимальна та середня КОВ суттєво відрізняються. Якщо для оцінювання складності обчислень орієнтуватися на значення середньої, а не максимальної КОВ, можна зменшити час, що виділяється на пошук, у 2,5 – 3 рази. Проте середня КОВ не є інформативною характеристикою для ущільнення мовленнєвих сигналів у реальному часі, оскільки час, заощаджений на обчислення в одному фреймі, неможливо використати в іншому. Тому складність обчислень має бути константою для різних фреймів, що визначається часовим інтервалом (вікном), відведеним на пошук найближчого вектора у кодовій книзі. Логічно припустити, що цей інтервал є величиною, пропорційною КОВ, і оцінити для різних вікон кількість пропущених векторів MV і спектральне спотворення SD . Результати дослідження, що їх було отримано для кодової книги розмірністю $M = 5$ та розміром $N = 4096$, наведено у табл. 3 та представлено на рис. 5 (1 – впорядкування з мажоризацією 1, 2 – впорядкування з мажоризацією 2, 3 – впорядкування за середнім).

Таблиця 3. Кількість пропущених векторів і спектральне спотворення для різних вікон

КОВ	Відсоток пропущених векторів MV , %				Спектральне спотворення SD , дБ			
	RND	Впорядкування за мажоризацією 1	Впорядкування за мажоризацією 2	Впорядкування за середнім	RND	Впорядкування за мажоризацією 1	Впорядкування за мажоризацією 2	Впорядкування за середнім
20	96,75	31,34	25,61	24,11	1,080	1,110	0,990	0,973
40	64,50	13,27	9,81	8,31	1,028	0,917	0,888	0,880
60	42,60	4,66	2,87	2,15	0,940	0,872	0,862	0,859
80	23,08	1,18	0,43	1,16	0,893	0,858	0,855	0,857
100	10,88	0,22	0,08	0,15	0,871	0,854	0,854	0,855
120	4,73	0,02	0,00	0,02	0,862	0,854	0,854	0,854

Рис. 5. Залежність MV від КОВ

Як можна бачити з табл. 3, усі методи пошуку із впорядкуванням векторів мають ефективність, вищу за метод *RND*. Таким чином, впорядкування кодової книги дозволяє суттєво зменшити складність обчислень за рахунок скорочення числа векторів, що аналізуються під час лінійного пошуку. Найкращі результати при використанні вікон розміром 100 – 120 векторів досягнуто при впорядкуванні векторів за відношенням мажорювання 2.

Висновки

Складність обчислень при застосуванні впорядкування векторів у кодовій книзі складає 3 – 3,5% від складності обчислень при повному пошуку. При цьому витрати пам'яті зросли в 12 – 15 разів відповідно при впорядкуванні за середнім та за мажоризацією.

Таким чином, підвищення ефективності пошуку в процесі передавання мовленнєвих сигналів досягається за рахунок значних обчислювальних витрат на підго-

товчому етапі, необхідних для створення таблиць суміжності A_j , а також збільшення обсягів пам'яті.

Список літератури

1. Paliwal K. K., Atal B. S. Efficient vector quantization of LPC parameters at 24 bits/frame — IEEE Transaction on Speech and Audio Processing, 1993. — No. 2, vol. 1. — P. 3–14.
2. Zhou J., Shoham Y., Akansu A. Simple Fast Vector Quantization of the Line Spectral Frequencies // Image Compression and Encryption Technologies. — 2001. — Vol. 4551. — P. 274–282.
3. Біліченко Н.О., Ткаченко О. М., Феферман О. Д., Хрущак С. В. Швидкий пошук при векторному квантуванні лінійних спектральних частот // Реєстрація, зберігання і обробка даних. — 2008. — №2, т. 10. — С. 37–47.
4. Agrell E. Spectral coding by fast vector quantization // Proc. IEEE Workshop on Speech Coding for Telecommunications, Sainte-Adèle, Québec, Canada. — 1993. — P. 61–62.
5. Маршалл А., Олкин И. Неравенства: теория мажоризации и ее приложения: пер. с англ. — М.: Мир, 1983. — 576 с.