УДК 519.8

А.М. НАГІРНА[*]

ЛОКАЛІЗАЦІЯ ЗНАЧЕННЯ ЛІНІЙНОЇ ФУНКЦІЇ, ЗАДАНОЇ НА МНОЖИНІ СПОЛУЧЕНЬ

[*]Український державний університет фінансів та міжнародної торгівлі, Київ, Україна

Анотація. *Розглядається алгоритм локалізації лінійної функції, заданої на конфігурації сполучень, з урахуванням її представлення у вигляді неорієнтованого графа. Приведено числовий приклад реалізації алгоритму.*
Ключові слова: *функція, локалізація, конфігурація сполучень, генерування, неорієнтований граф, дерево, піддерево, корінь, гамільтонів шлях.*

Аннотация. *Рассматривается алгоритм локализации линейной функции, заданной на конфигурации сочетаний, с учетом ее представления в виде неориентированного графа. Приведен числовой пример реализации алгоритма.*
Ключевые слова: *функция, локализация, конфигурация сочетаний, генерирование, неориентированный граф, дерево, поддерево, корень, гамильтонов путь.*

Abstract. *This paper considers the localization algorithm of a linear function on the configuration of combinations taking into account its representation in the form of an undirected graph. A numerical example of algorithm realization is given.*
Keywords: *function, localization, configuration of combinations, generation, undirected graph, tree, subtree, root, Hamilton path.*

1. Вступ

Оптимізаційні комбінаторні задачі є одними з найбільш важких з обчислювальної точки зору. В більшості випадків методи їх розв'язання зводяться до повного перебору варіантів, що не є ефективним для задач великої розмірності. Тому при розв'язанні практичних задач часто виникає необхідність розробляти нові та удосконалювати існуючі методи, як точні, так і наближені, які були б застосовні до задач більшої розмірності, ніж метод повного перебору [1, 2].

У процесі розробки і реалізації алгоритму природним чином розкриваються властивості, що відображають комбінаторні характеристики, які використовувалися в модифікації з новими підходами [3–5]. Графи як абстрактні математичні моделі можна використовувати при розв'язанні комбінаторних задач різної розмірності, оскільки вони є відображенням многогранної структури певних комбінаторних конфігурацій. Тому представлення комбінаторних множин у вигляді графів дозволяє отримати нові підходи та методи вирішення [6–8].

При врахуванні взаємозв'язку між комбінаторними конфігураціями і їх графовими моделями вивчаються структурні властивості допустимої області, які забезпечують знаходження розв'язку задачі за лічені кроки.

Дана робота продовжує дослідження робіт [6–10]. В ній описується застосування нового підходу до розв'язання комбінаторних задач локалізації функції на комбінаторній конфігурації сполучень.

2. Алгоритм локалізації значень лінійної функції на конфігурації сполучень

Нехай дано множину $A' = (1, 2, ..., n)$. Сполучення без повторень з n елементів по r – це r-елементна підмножина множини A'. Оскільки порядок запису елементів множини неістотний, то запишемо елементи в кожному сполученні у порядку зростання. Сполучення

$(a_1, a_2, ..., a_r)$ розглядатимемо як рядок чисел $a_1, a_2, ..., a_r$, де $a_1 < a_2 < ... < a_r$. C_n^r – кількість усіх сполучень без повторень з n елементів по r, де n, r – додатні цілі числа, причому $r \le n$. За даним сполученням можна знайти наступне відповідно до лексикографічного порядку.

Алгоритм побудови лексикографічно наступного сполучення [5]:

1. Знайти в рядку $a_1, a_2, ..., a_r$ перший справа елемент a_i такий, що $a_i \ne n - r + i$.
2. Для знайденого елемента виконати присвоювання $a_i = a_i + 1$.
3. Для $j = i+1, i+2, ..., r$ виконати $a_j = a_i + j - i$ або $a_j = a_{j-1} + 1$.

Наступний алгоритм генерує всі k - елементні підмножини так, що кожна наступна підмножина утворюється з попередньої з видаленням одного елемента і додаванням іншого. Розглянемо алгоритм у рекурсивній формі.

Позначимо через $G(n,k)$ список, який має всі k - елементні підмножини множини $\{1,...,n\}$, в якій першою підмножиною є $\{1,...,k\}$, останньою – $\{1,2,...,k-1,n\}$, і кожна наступна підмножина утворюється з попередньої з видаленням деякого елемента і додаванням другого. Зазначимо, що якщо $G(n-1,k)$ і $G(n-1,k-1)$ вже побудовані, то $G(n,k)$ можна визначити таким чином: $G(n,k) = G(n-1,k-1), G^*(n-1,k-1) \cup \{n\}$, де $G^*(n-1,k-1) \cup \{n\}$ означає список, утворений з $G(n-1,k-1)$ зміною порядку елементів списку на зворотний та наступним додаванням елемента n до кожної множини. Списку $G(n,k)$, як і у випадку генерування всіх підмножин, можна поставити у відповідність деякий гамільтонів шлях у графі [6].

Розглянуті вище методи генерування комбінаторної конфігурації сполучень не дають можливості побудувати упорядкування за значенням цільової функції. Тому слід розглянути нові методи генерування комбінаторної конфігурації сполучень, які дають можливість, у залежності від складності задачі, представити елементи комбінаторної конфігурації у вигляді графа.

Даний метод генерування полягає у виборі елементів з заданої упорядкованої множини за зростанням, тобто вибираються елементи для прикладу по два – перший, другий; перший, третій; перший, четвертий і т.д. Таким чином будується верхній підграф загального графа послідовності, далі – другий, третій; другий, четвертий т.д.

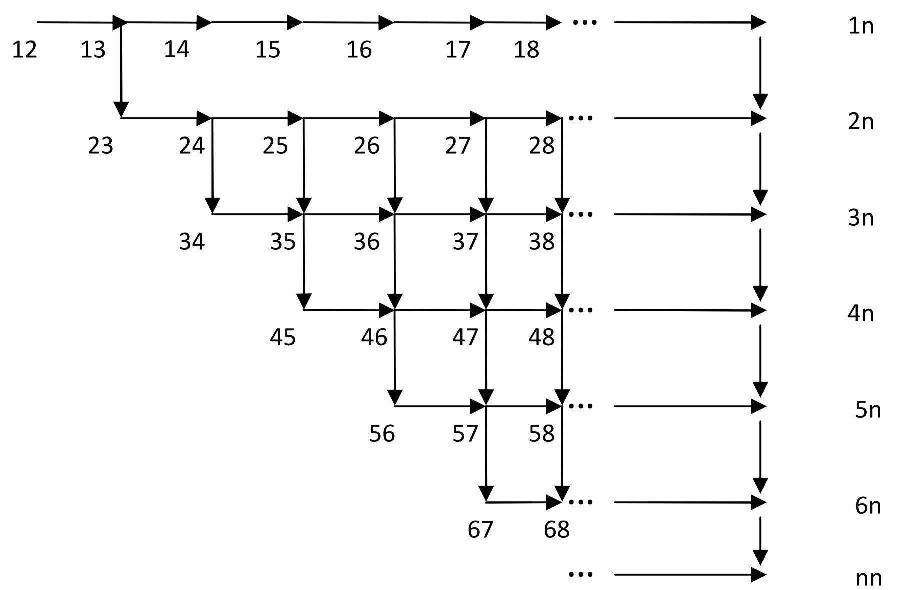

Рис. 1. Граф сполучень з n по 2

Дерево на рис. 1 можна представити у вигляді розкладу на піддерева, тоді кожне піддерево формується з вершин, в яких елемент на останньому місці фіксується і є вибраним з початкової множини $a_1, a_2, ...a_n$ як максимальний, а останні перебираються рекурсивним методом.

Слід зазначити, що дане дерево є орієнтованим, в якому визначений корінь – найнижча вершина останнього піддерева і яка є лексикографічно більша за всіх інших.

Тоді алгоритм побудови наступного сполучення полягає у виконанні таких кроків.

Крок 1. Знайти в рядку $a_1, a_2, ...a_r$ перший справа елемент a_i такий, що $a_i \neq n - r + i$.

Крок 2. Для знайденого елемента виконати присвоювання $a_i := a_i + 1$.

Крок 3. Для $j = i+1, i+2, ..., r$ виконати $a_j := a_i + j - i$ (або, що те ж саме, $\left(a_j := a_{j-1} + 1\right)$.

Даний алгоритм дає можливість будувати наступне по порядку сполучення. Рядок чисел, яким подано лексикографічно наступне сполучення, відрізняється від рядка, що зображає дане сполучення, з позиції i, бо в даному сполученні в позиціях $i+1, i+2, ..., r$ є максимально можливі числа. Отже, $a_i + 1$ – найменше можливе число, яке можна записати в позицію i, якщо хочемо отримати сполучення більше від даного. Тоді $a_i + 2, ..., a_i + r - i + 1$ – найменш можливі числа, які можна записати в позиціях від $i+1$ до r.

Важливість запропонованого методу генерування комбінаторної конфігурації сполучень у тому, що одержуємо граф (дерево), який можна використовувати для розв'язування комбінаторних задач, оскільки граф складається із усіх елементів сполучення.

Метод, що об'єднує засоби комбінаторного аналізу та теорії графів, передбачає послідовне виконання таких пунктів [9]:

– вибір способу генерування у певній послідовності всіх елементів заданої комбінаторної конфігурації, який найбільше пристосований до заданої функції цілі;

– представлення множини комбінаторної конфігурації у вигляді орієнтованого графа, де дуга відповідає спаданню значень цільової функції;

– побудову поліноміального алгоритму розв'язку задачі на частково упорядкованих вершинах графа.

Цей метод було застосовано для розв'язання задач з лінійною та дробово-лінійною цільовими функціями на перестановках, розбиттях, комбінаціях та розміщеннях. Продовжимо дослідження з метою створення відповідного алгоритму для розв'язання комбінаторної задачі на конфігурації сполучень.

Згідно з методом генерування, елементи множини конфігурації сполучень можна зобразити у вигляді дерева, неорієнтованого графа, що не має циклів, тобто такого, в якому кожна пара вершин з'єднана одним простим шляхом – ланцюгом.

Дерево можна орієнтувати, вибравши для цього довільну вершину, як корінь і ребрам приписати таку орієнтацію, щоб кожна вершина з'єднувалася з коренем тільки одним простим шляхом. Тоді елементи конфігурації сполучень будуть розміщені у вигляді орієнтовного дерева, де піддерево визначається початковим елементом у вершині.

Слід зазначити, що при розв'язуванні екстремальних задач на множині сполучень не має значення порядок розміщення елементів, а тільки їх вибір з вихідної множини, в якій елементи впорядковані за зростанням, тому екстремальне значення функції завжди можна визначити.

Нехай задано числове значення функції $f(x_0)=y_0$. Необхідно знайти точки конфігурації сполучень, в яких досягається дане числове значення.

Алгоритм локалізації значення лінійної функції на конфігурації сполучень полягає у:

1) формуванні множини сполучень: $A_i(a_1,a_2,...,a_k)$, $i=1,...,k$; введенні n, r $(n\ge r)$, визначенні кількості елементів $C_n^r=\dfrac{n!}{k!(n-k)!}$;

2) побудові базового дерева конфігурації сполучень: визначенні кількості вершин базового дерева C_n^r, піддерев (підплощин) $d=n-r+1$;

3) визначенні початкової і кінцевої крайніх точок підграфів та знаходженні $\max f(\overline{x}_j)$, $\min f(\overline{x}_i)$ у цих точках відповідного піддерева;

4) визначенні множини точок піддерева, що задовольняють умову $f(\overline{x}_i)\ge f\left(x^0\right)$, де x^0 – точка, для якої $f\left(x^0\right)\ge y^0$, а $\overline{x}_i, i\in N_k$ – множина точок, значенні цільової функції $f(x)$, у яких більше заданого;

5) визначенні множини вершин піддерева, для яких виконується умова $f(\overline{x}_j)\le f(x^0)$, де x^0 – точка, для якої виконується умова $f(x^0)\le y^0$; а $\overline{x}_j, j\in N_{n-k}$ – множина точок, для яких значення цільової функції $f(x)$ менше заданого;

6) знаходженні множини піддерев , утворених шляхом перетину множин п. 5–6;

7) формуванні множини точок − елементів сполучень, що задовольняють умові $f(x_0)=y_0$. Якщо всі точки знайдені, тобто серед множини піддерев визначені, і задача розв’язана. В іншому випадку – перехід до наступного п. 8;

8) визначенні піддерева, для якого виконується умова $f(\overline{x}_j)\le y_0\le f(\overline{x}_i)$;

9) $n:=n-1$, перехід до п. 1.

Розглянемо числовий приклад застосування алгоритму.

3. Приклад застосування алгоритму

Дано цільову функцію $f(x)=x_1x_2+x_1x_3+x_2x_3$ на конфігурації сполучень C_n^r, де $n=6$, $A_i=(1,2,3,4,5,6)$, $r=3$. Тоді $C_6^3=20$ – кількість точок дерева. Відповідно, кількість піддерев у дереві рівна $d=4$. Визначене значення функції $f(x)=47$.

Необхідно знайти точки – елементи конфігурації сполучень, в яких досягається задане значення цільової функції.

Розв’язання

Для конфігурації сполучень елементи множини зобразимо у вигляді дерева, що складається із трьох піддерев. Елементи конфігурації сполучень розміщуємо у вигляді неорієнтованого графа без циклів. Вибираємо довільну вершину як корінь для орієнтації дерева. Ребрам приписуємо таку орієнтацію, щоб кожна вершина з’єднувалася з коренем тільки одним простим шляхом. Елементи

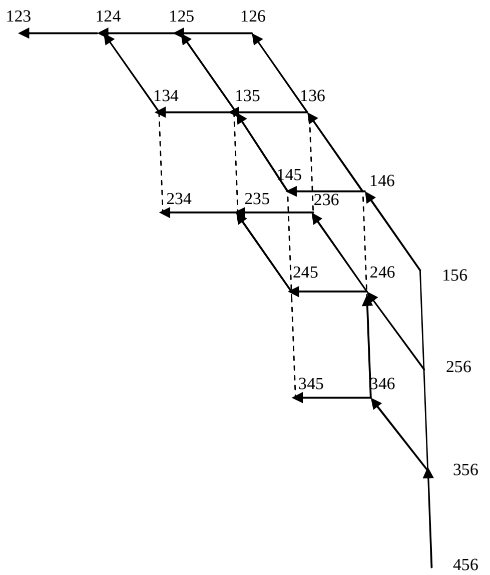

Рис. 2. Стандартний вигляд базового дерева G конфігурації сполучень C_6^3

конфігурації сполучень будуть розміщені у вигляді орієнтовного дерева, де піддерево визначається початковим елементом у вершині.

Відповідно, базове орієнтовне дерево із трьох піддерев (підплощин), утворене множиною сполучень без повторень з шести елементів по три, буде мати такий вигляд.

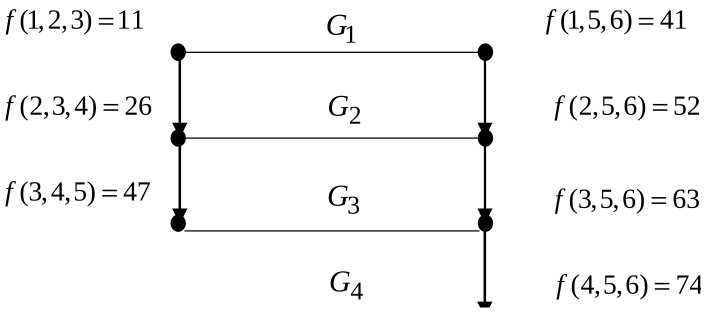

Рис. 3. Структурна схема дерева сполучень для $f(x)$

На рис. 3 структурна схема дерева G конфігурації сполучень розбивається на піддерева G_1, G_2, G_3, G_4, де відмічені крайні точки піддерев (півплощин). Зліва знаходяться вершини, в яких досягаються мінімальні значення функції відповідного піддерева, а справа – максимальні значення функції $f(x)$.

Користуючись базовим деревом конфігурації сполучень, будуємо структурне дерево розв'язків.

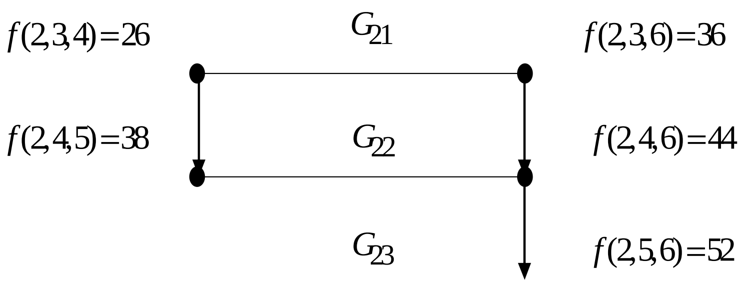

Рис. 4. Структурна схема піддерева сполучень G_2

Задане значення цільової функції $f(x)=47$ досягається в точках (3, 4, 5) відповідно піддерева G_2, де $f(2,3,4)\le f(x)\le f(2,5,6)$ і G_3, $f(3,4,5)=f(x)\le f(3,5,6)$ можуть містити задане значення у відповідних вершинах.

Розглянемо піддерево G_2.

Піддерева G_{21}, G_{22} не містять необхідних точок, а єдина точка $(2,5,6)$ піддерева G_{23} набуває значення $f(2,5,6)=52$, що значно більше $f(x)=47$.

Згідно з рис. 2, піддерево G_3 складається із трьох вершин: $(3,4,5)$, $(3,4,6)$, $(3,5,6)$. Оскільки $f(3,4,5)=47$, тоді значення функції $f(x)$ в наступних вершинах рівне $f(3,4,6)=54$, $f(3,5,6)=63$, що не задовольняє умову.

Отже, функція $f(x)$ досягає значення 47 у вершині $(3,4,5)$.

4. Висновки

У роботі описано та досліджено алгоритм локалізації лінійної функції на конфігурації сполучень, елементи якої, з урахуванням методу генерування, представлені у вигляді неорієнтованого графа, що не має циклів. За рахунок структурних схем піддерев сполучень загального базового дерева знаходяться або вершини, або відповідні піддерева, що містять необхідні значення функції. Якщо локальне значення функції не потрапляє в числові екстремальні інтервали відповідних піддерев, то розглядаються наступні піддерева і т.п. Якщо ж не існує конкретної вершини базового дерева, що забезпечує досягнення даного числового значення функції, то тоді розв'язком буде ребро відповідного піддерева, в екстремальний інтервал якого потрапляє значення функції.

Алгоритм об'єднує засоби комбінаторного аналізу та теорії графів і надає можливість не робити повний перебір елементів конфігурації сполучень, на якій розглядається задача.

У результаті наукового дослідження було розглянуто приклад задачі на конфігурації сполучень, представленої у вигляді неорієнтованого дерева.

Подальші дослідження будуть спрямовані на адаптацію нових підходів та алгоритмів розв'язання комбінаторних задач на інших конфігураціях з нелінійними функціями з можливістю зростання потужності множини із застосуванням графових моделей та програмної реалізації методів.

СПИСОК ЛІТЕРАТУРИ

1. Баранов В.И. Экстремальные комбинаторные задачи и их приложения / В.И. Баранов, Б.С. Стечкин. – М.: Наука, 1989. – 160 с.
2. Сергиенко И.В. Модели и методы решения на ЭВМ комбинаторных задач оптимизации / И.В. Сергиенко, М.Ф. Каспшицкая. – К.: Наукова думка, 1981. – 288 с.
3. Семенова Н.В. Поліедральний підхід до розв'язання одного класу векторних задач комбінаторної оптимізації / Н.В. Семенова, Л.М. Колєчкіна, А.М. Нагірна // Доповіді НАН України. – 2009. – № 6. – С. 46 – 53.
4. Колечкина Л.Н. Многокритериальные комбинаторные задачи оптимизации на множестве полиразмещений / Л.Н. Колечкина, Е.А. Родионова // Кибернетика и системный анализ. – 2008. – № 2. – С. 152 – 160.
5. Колєчкіна Л.М. Властивості задач багатокритеріальної оптимізації на комбінаторних множинах та методи їх розв'язання / М.Л. Колєчкіна. – Полтава: РВВ ПУСКУ, 2008. – 162 с.
6. Донец Г.А. Построение гамильтонова пути в графах перестановочных многогранников / Г.А. Донец, Л.Н. Колечкина // Кибернетика и системный анализ. – 2010. – № 1. – С. 10 – 16.
7. Донец Г.А. Экстремальные покрытия графов / Г.А. Донец, А.Я. Петренюк. – Кировоград: ОАО «Кіровоградське видавництво», 2009. – 170 с.
8. Донец Г.А. Об одном подходе к решению комбинаторной задачи оптимизации на графах / Г.А. Донец, Л.Н. Колечкина // Управляющие системы и машины. – 2009. − № 4. – С. 36 − 42.
9. Донец Г.А. Локализация значения линейной функции, заданной на перестановках / Г.А. Донец, Л.Н. Колечкина // Радиоэлектроника и информатика. – 2009. − № 1. – С. 76 – 81.
10. Донец Г.А. Метод упорядочения значений линейной функции на множестве перестановок / Г.А. Донец, Л.Н. Колечкина // Кибернетика и системный анализ. – 2009. − № 2. – С. 50 − 61.

Стаття надійшла до редакції 13.06.2014