УДК 004.67:004.822

О. К. Сулема[1], Д. В. Ланде[2]
[1]Національний технічний університет України
«Київський політехнічний інститут»
Проспект Перемоги, 37, 03056 Київ, Україна
[2]Інститут проблем реєстрації інформації НАН України
вул. М. Шпака, 2, 03113 Київ, Україна

Знаходження оптимальної ієрархії у квазіієрархічному графі за критеріями центральності

Викладено результати дослідження, присвяченого виділенню із квазіієрархічних мереж ієрархічних підмереж, найкращих, з точки зору критеріїв центральності. Розглянуто методи багатокритеріальної оцінки для виявлення вузла, оптимального за критеріями центральності. Проаналізовано алгоритми побудови ієрархії. Проведено експериментальну перевірку отриманих теоретичних результатів на групі квазіієрархічних графів.

Ключові слова: *ієрархія, квазіієрархічні графи, критерії центральності, багатокритеріальна оптимізація, кістякове дерево.*

Вступ

Однією з важливих практичних задач, що вирішуються за допомогою квазіієрархічних графів, є задача дослідження організаційної структури з метою налагодження ефективних інформаційних потоків в організації (установі, компанії, корпорації тощо), зокрема, для забезпечення інформаційної безпеки [1].

Якщо представити структуру деякої крупної організації (компанії) у вигляді графа, то для оптимізації організаційної структури цієї компанії потрібно провести аналіз найбільш ефективних ланок у передаванні інформації з тим, щоб реорганізувати організаційну структуру і покращити керованість бізнес-процесів у ній, а також вирішити іншу задачу — задачу інформаційної безпеки. Для цього потрібно відновити з існуючої квазіієрархічної мережі ієрархію без «зайвих» зв'язків, попередньо виявивши найбільш оптимальний для цього вузол.

Постановка задачі

Квазіієрархічним графом, або квазіієрархією, будемо називати такий зв'язаний граф, в якому, порівняно з ієрархічним графом, є «зайві» зв'язки. Під «зайвими» зв'язками розуміються ребра, які утворюють цикли. Квазіієрархічним графом з N вершинами можна вважати будь-який зв'язний граф, у якого кількість ребер становить $O(N)$.

Метою дослідження, результати якого викладені у даній статті, є виявлення у квазіієрархії «найкращого», з точки зору заданих критеріїв вузла, та побудова відповідної ієрархічної мережі, в якій цей вузол буде виступати як основа — перший рівень.

Для досягнення даної мети необхідно вирішити наступні підзадачі:

— дослідження квазіієрархічного графа за критеріями центральності;

— визначення найбільш оптимального вузла з точки зору задачі, що розглядається;

— аналіз алгоритмів і підходів побудови кістякового дерева;

— отримання ієрархії із квазіієрархічної мережі.

Найбільш оптимальний вузол за критеріями центральності

У теорії графів існує велика кількість характеристик графа мережі. Зокрема, існує ряд характеристик центральності, кожна з яких може розглядатися відповідно до поставленої практичної задачі. Авторами було проаналізовано та виділено групу критеріїв центральності вузлів графа ієрархічної структури [2]:

— критерій мінімального середнього шляху;

— критерій максимального степеню;

— критерій мінімальної кількості рівнів.

За даними критеріями і відбуватиметься обрання кореня дерева — так званої «найкращої» вершини в рамках практичної задачі, що розглядається.

У загальному випадку зазначені критерії центральності дають різні результати для графа, що аналізується. При цьому вони є рівнозначними з точки зору пріоритетності, а тому при отриманні декількох різних оптимальних вершин за цими критеріями не можна просто виділити одну з них, відкинувши інші. З метою отримання єдиного розв'язку поставленої задачі має бути застосована багатокритеріальна оцінка отриманих вершин. Для цього, наприклад, можна обрати підхід Парето [3, 4].

Даний підхід полягає у наступному. Складається таблиця, рядки якої — це відповідні критерії, а стовпці — номери вершин графа, що аналізується. Комірки даної таблиці заповнюються відповідними значеннями критеріїв для кожної вершини даного графа. Приклад такої таблиці Парето наведено в табл. 1.

Таблиця 1. Таблиця Парето

Вершина	1	2	3	4	5	6	7	8	9	10	11
Критерій 1	1,09	1,17	2,11	2,68	1,03	1,14	2,21	2,21	1,11	1,87	1,87
Критерій 2	3	2	3	3	2	1	1	1	2	1	1
Критерій 3	0,14	0,14	0,11	0,7	0,46	0,58	0,27	0,64	0,32	0,36	0,27

Після того як була складена таблиця Парето, відбувається поступове порівняння вершин графа (тобто відповідних стовпців таблиці) за принципом «не менше» (або «не більше») за всіма критеріями, а саме: якщо i-та вершина більша (при мінімізації) або менша (при максимізації) за *j*-ту вершину хоча би за одним критерієм при однакових інших, то дана *i*-та вершина більше не береться до уваги. Але у випадку, якщо хоча би за одним критерієм *i*-та вершина менша (при мінімізації) або більша (при максимізації) за *j*-ту при тому, що за одним чи декількома іншими вона більша (при мінімізації) або менша (при максимізації), то в подальшому враховуються обидві вершини [2].

У результаті, множина таких невикреслених вершин і складатиме множину Парето. Як правило, вона складається з двох чи більше вершин. Для остаточного розв'язку задачі вибору кореня ієрархії необхідно визначити найбільш оптимальну для конкретної практичної задачі, що розв'язується, «найкращу» вершину. Для цього можуть бути застосовані так звані «наївні» методи багатокритеріальної оптимізації, одним з яких є об'єднання всіх критеріїв в одну функцію пристосованості $Fitness$ за допомоги лінійного співвідношення

$$Fitness(i) = \alpha \cdot LevelsNumber(i) + \beta \cdot AveragePath(i) + \gamma \cdot VertexDegree(i),$$

де i — вершина із множини Парето; $\alpha,\ \beta,\ \gamma \in [0,1]$ — ваги критеріїв, тобто коефіцієнти оцінки, що визначаються відповідно до практичної задачі, що розв'язується; *LevelsNumber*, *AveragePath*, *VertexDegree* — відповідні критерії, що оцінюються.

Отримана в результаті вершина і буде вважатись, у даному конкретному випадку, найбільш оптимальною для побудови ієрархії.

Побудова ієрархії на основі квазіієрархічної мережі

У рамках задачі побудови ієрархії із квазіієрархічного графа авторами було виділено наступні три алгоритми:

— алгоритм побудови кістякового дерева;

— алгоритм порівневої побудови за всіма суміжними вершинами;

— алгоритм вибіркової порівневої побудови за суміжними вершинами.

Алгоритм побудови кістякового дерева ґрунтується на алгоритмі Прима [5]. Нехай дано неорієнтований зв'язний граф $G = (V, E)$ з n вершинами та m ребрами. Тоді на першому кроці до майбутнього дерева включається деяка довільна вершина графа G.

На другому кроці серед ребер, що з'єднують дану вершину з іншими вершинами, обирається ребро найменшої ваги, яке додається до дерева. У випадку незваженого графа ребро обирається псевдо-випадково.

На третьому кроці серед інцидентних даній вершині ребер аналогічним псевдовипадковим чином обирається наступне ребро та додається до дерева. Відповідно, до дерева додається інцидентна даному ребру вершина. Даний крок повторюється до тих пір, поки дерево не міститиме всі вершини графа (тобто $n - 1$ ребро при n вершинах). Мінімальне кістякове дерево побудовано.

Крім того, авторами запропоновано алгоритм порівневої побудови ієрархії за всіма суміжними вершинами. Алгоритм ґрунтується на тому, що граф може бути описаний матрицею суміжності.

Нехай дано неорієнтований граф $G=(V,E)$ з n вершинами та m ребрами. Граф G представлено у вигляді матриці суміжності M. Тоді на першому кроці визначається вершина графа, що буде коренем ієрархії. Визначення відбувається на основі критеріїв центральності.

На другому кроці для обраного кореня ієрархії за матрицею суміжності M визначаються всі суміжні з ним вершини. Дані вершини вважаються першим рівнем ієрархії, що будується, та записуються у нову матрицю суміжності K, що формуватиме ієрархічний граф H.

На третьому кроці у матриці суміжності M переходимо до рядка, що відповідає першій із суміжних з коренем вершині. Продовжуємо будувати підієрархію, умовно вважаючи тепер дану вершину її коренем. Для цього повторимо другий крок, продовжуючи заповнювати матрицю суміжності K.

На четвертому кроці у матриці суміжності M переходимо до наступної вершини першого рівня, після чого аналогічно повторюємо третій крок. Після того як усі вершини даного рівня були оброблені, переходимо до вершин наступного рівня та аналогічно виконуємо другий, третій та четвертий кроки.

Таким чином, продовжуємо процес до тих пір, доки всі вершини початкового графа G не будуть оброблені, в результаті чого утвориться зв'язний ієрархічний граф H. Ребра, що не були включені до даної ієрархії, вважаються «зайвими» зв'язками.

Також, авторами розглянуто алгоритм вибіркової порівневої побудови ієрархії за суміжними вершинами. У даному алгоритмі виділення ієрархії із квазіієрархічного графа є вибірковим, оскільки його ключова ідея полягає в тому, що на кожному рівні враховується лише частина суміжних кореню вершин.

Аналогічно попередньому алгоритму, нехай дано неорієнтований граф $G=(V,E)$ з n вершинами та m ребрами. Граф G представлено у вигляді матриці суміжності M. На першому кроці за критеріями центральності визначається коренева вершина графа, що аналізується.

На другому кроці за матрицею суміжності M для кореня ієрархії, що будується, визначаються суміжні вершини, які записуються у нову матрицю суміжності K, що в результаті відповідатиме шуканій ієрархії H. Даний крок повторюється до тих пір, поки в матриці суміжності M у рядку, що відповідає кореню дерева, що будується, не буде знайдений нуль. Після цього всі інші вершини, які також суміжні кореню, тимчасово відкидаються. Перший рівень попередньо побудований.

На третьому кроці, аналогічно попередньому алгоритму, у матриці суміжності M переходимо до рядка, що відповідає першій із суміжних з коренем вершині. Продовжуємо будувати дерево, повторюючи для даної вершини другий крок як для кореня піддерева. В результаті попередньо отримуємо другий рівень.

На четвертому кроці переходимо до наступної з першого рівня вершини та повторюємо третій крок. Аналогічний процес продовжується для всіх вершин першого рівня, після чого алгоритм переходить на другий рівень нової ієрархії H, та кроки алгоритму повторюються тепер для цього рівня.

Таким чином, даний процес продовжується, поки не буде побудований останній рівень. Після цього алгоритм переглядає, чи були вже всі вершини додані до ієрархії H. Якщо після основної роботи алгоритму залишилися незв'язні вершини, то за матрицею суміжності M проводиться ребро, що зв'язує дану вершину з першою ж за даним рядком у матриці M вершиною.

У результаті роботи алгоритму із квазіієрархії G утворюється зв'язний ієрархічний граф H. Ребра, що не були включені до даної ієрархії, також вважаються «зайвими» зв'язками.

Кожен із даних алгоритмів може бути використаний залежно від практичної задачі, що розглядається. Наприклад, алгоритм порівневої побудови за всіма суміжними вершинами даватиме найкращий результат у випадку, коли необхідно отримати найменшу кількість рівнів.

Експериментальне дослідження

Авторами проведено експериментальне дослідження на прикладі наведених у табл. 2 квазіієрархічних графів, які були згенеровані псевдовипадковим чином, з метою перевірки застосовності критеріїв центральності до обрання «найкращої» вершини та ефективності алгоритмів побудови ієрархії.

Таблиця 2. Квазіієрархічні графи

Назва графа	Кількість вершин	Радіус	Діаметр	Критерії		
				Мінімальний середній шлях	Максимальний степінь вершини	Мінімальна кількість рівнів
$k1$	16	3	6	7	4	12
$k2$	16	3	6	4	10	2, 14
$k3$	15	3	6	5	8	3
$k4$	15	3	6	6	8	4
$k5$	18	3	6	3	11	5
$k6$	21	4	7	5	9	1
$k7$	23	3	6	7	12	4
$k8$	23	3	6	22	6	4
$k9$	25	4	8	18	16	17
$k10$	27	3	6	8	18	5
$k11$	29	4	7	7	19	6, 9
$k12$	30	4	7	4	13	11, 16
$k13$	33	4	7	16	11	2, 16

Після застосування багатокритеріальної оптимізації за методом Парето була отримана множина Парето, до якої застосовувалася функція пристосованості. Для отримання остаточного результату розглядалися три групи ваг критеріїв — для кожного з критеріїв.

Для отримання результату за критерієм мінімального середнього шляху були обрані коефіцієнти $\alpha = 1$, $\beta = 0{,}05$, $\gamma = 0{,}01$. Для критерію максимального степеня вершини були підібрані коефіцієнти $\alpha = 0{,}06$, $\beta = 1$, $\gamma = 0{,}02$. Оптимальний результат за критерієм мінімальної кількості рівнів отримано при коефіцієнтах $\alpha = 0{,}01$, $\beta = 0{,}01$, $\gamma = 1$.

Результати застосування методів багатокритеріальної оптимізації для вищенаведених графів представлено в табл. 3.

Таблиця 3. Оптимізація за підходом Парето та функцією пристосованості

Назва графа	Кількість вершин	Оптимальна вершина за критерієм		
		Мінімальний середній шлях	Максимальний степінь вершини	Мінімальна кількість рівнів
*k*1	16	7	4	7
*k*2	16	4	10	2
*k*3	15	5	8	3
*k*4	15	6	8	4
*k*5	18	3	11	5
*k*6	21	5	9	1
*k*7	23	7	12	4
*k*8	23	22	6	4
*k*9	25	18	16	17
*k*10	27	8	18	5
*k*11	29	7	19	9
*k*12	30	4	4	11
*k*13	33	16	11	16

Після отримання оптимальної вершини (за одним із критеріїв, що був обраний як найбільш пріоритетний за функцією пристосованості) тепер можна «витягнути» із квазіієрархії відповідну ієрархію. На рис. 1 наведений вихідний квазіієрархічний граф *k*11.

До даного графа були застосовані алгоритми перетворення квазіієрархії на ієрархію. Граф був «витягнутий» за вершину, обрану за критерієм мінімальної кількості рівнів.

Результат застосування алгоритму Прима представлено на рис. 2. Результат застосування алгоритму порівневої побудови ієрархії за всіма суміжними вершинами представлений на рис. 3. Результат застосування алгоритму вибіркової порівневої побудови за суміжними вершинами представлений на рис. 4.

Таким чином, бачимо, що для практичної задачі (зокрема, задачі безпеки) отримання ієрархії з мінімальною кількістю рівнів найкращій результат дає алгоритм порівневої побудови за всіма суміжними вершинами (рис. 3).

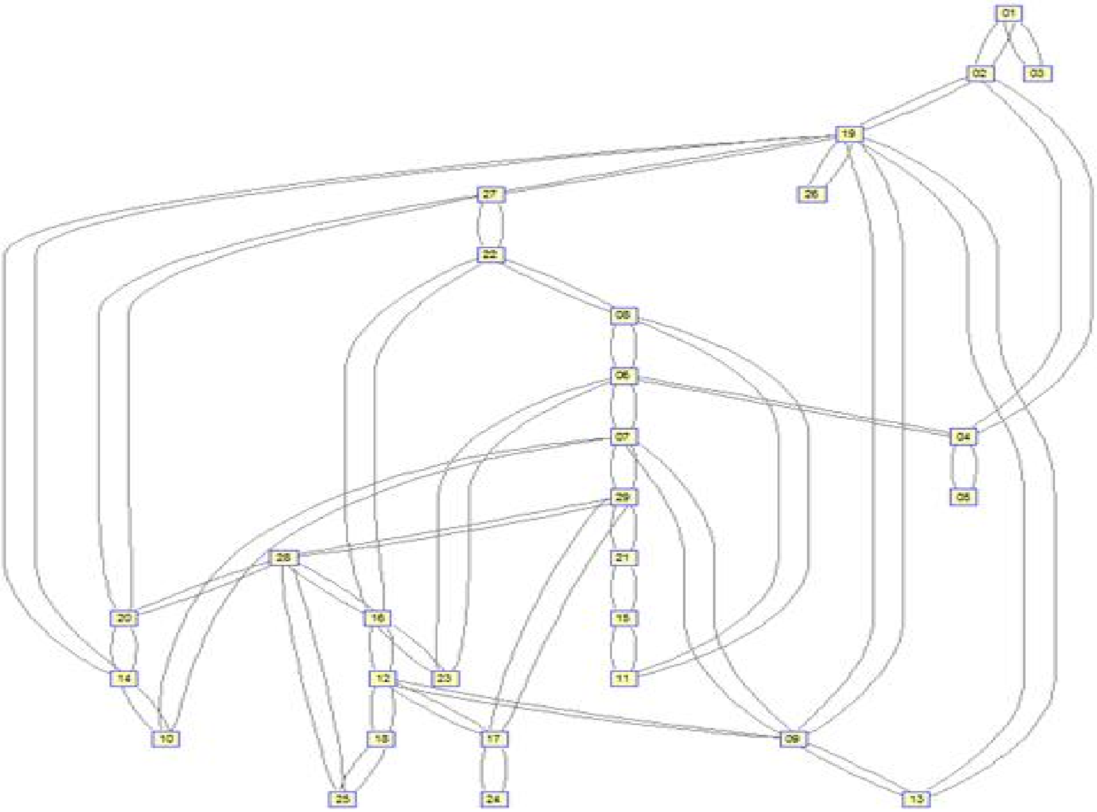

Рис. 1. Квазіієрархічний граф

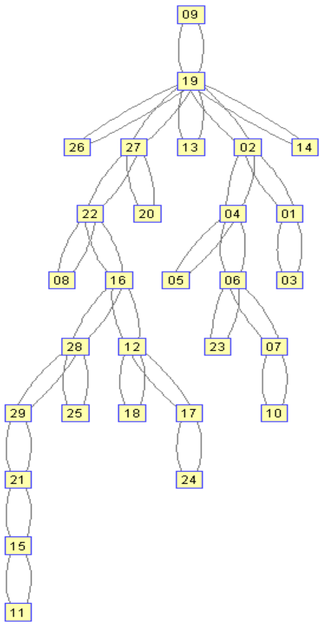

Рис. 2. Ієрархія, отримана
за алгоритмом Прима

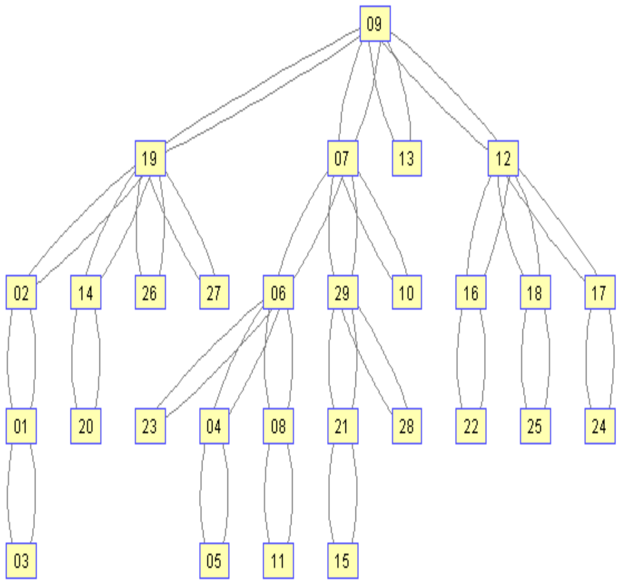

Рис. 3. Ієрархія, отримана за всіма
суміжними вершинами

Висновки

У рамках проведеного авторами дослідження розглянуто різні алгоритми отримання ієрархічного графа із квазіієрархічної мережі. З метою обрання кореня ієрархії до критеріїв центральності застосовано методи багатокритеріальної оптимізації. Результати дослідження експериментально перевірено на групі квазіієрархічних графів, у результаті чого зроблено висновок, що для практичної задачі отримання мінімальної кількості рівнів (зокрема, задачі безпеки) найкраще пристосований алгоритм порівневої побудови ієрархії за всіма суміжними вершинами.

Практичним результатом даного дослідження є математичне та програмне забезпечення, розроблене у середовищі MATLAB для дослідження квазіієрархічних графів, зокрема, визначення «найкращого» вузла та виділення ієрархії із квазіієрархічної мережі з відповідною візуалізацією отриманих результатів.

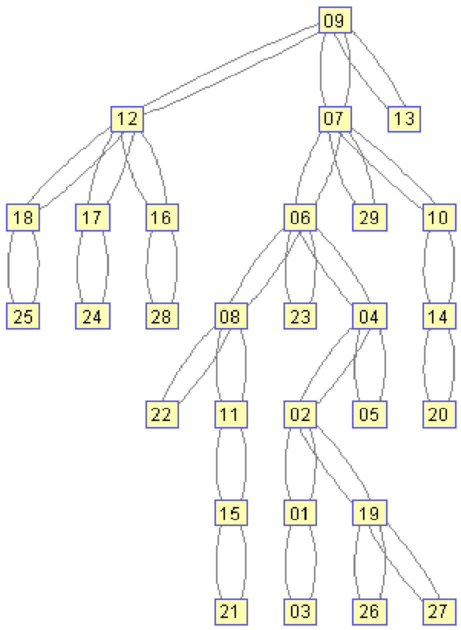

Рис. 4. Ієрархія, що отримана за алгоритмом вибіркової порівневої побудови за суміжними вершинами

1. *Гайворонський М.В.* Безпека інформаційно-комунікаційних систем [Текст] / М.В. Гайворонський, О.М. Новіков. — К.: Видавнича група BHV, 2009. — 608 с.

2. *Сулема О.К.* Дослідження критеріїв центральності в ієрархічних мережах [Текст] / О.К. Сулема, Д.В. Ланде // Материалы XV международной научно-практической конференции «Информационные технологии и безопасность» (Выпуск 15). — К., 2015. — С. 219–223.

3. *Трифонов, А.Г.* Многокритериальная оптимизация [Электронный ресурс] / А.Г. Трифонов. — Режим доступа: http://matlab.exponenta.ru/optimiz/book_1/16.php

4. *Михалевич, В.С.* Методы выпуклой оптимизации [Текст] / В.С. Михалевич, А.М. Гупал, В.М. Норкин. — М.: Наука, 1987. — 326 с.

5. *Рыбаков, Г.* Минимальные остовные деревья [Электронный ресурс] /Г. Рыбаков. — Санкт-Петербургский государственный университет информационных технологий, механики и оптики. — 2005. — Режим доступа : http://rain.ifmo.ru/cat/view.php/theory/graph-spanning-trees/mst-2005

Надійшла до редакції 10.12.2015