

група дослідження преподавання. Модель діалога людини-преподавателя контролює діяльність в AutoTutor. 4. А. Коллінз, П. Невіль, К. Білячик. Роль різних засобів спілкування в проектуванні навчальних серед // *International Journal of Artificial Intelligence in Education* (2000), 11. – Р.144–162. 5. Полат Е.С. Дистанційне навчання ([http://scholar.irc.ac.ru/ped\\_journal/numero4/pedag/polat.html](http://scholar.irc.ac.ru/ped_journal/numero4/pedag/polat.html) – 05.09.2003 р.). 6. Федорова Е. Ф. Системне представлення дистанційного навчання // *Збірник наукових праць “Образование и виртуальность Южно-Уральського державного ун-та. – 2002”*. – Харків–Ялта, 2002. 7. Морзе Н.В. Інтерактивні методи в дистанційному навчанні. Академія праці і соціальних відносин, Київ // *Збірник наукових праць “Образование и виртуальность – 2002”*. – Харків–Ялта, 2002.

УДК 004.65

А.Г. Григорович, В.Г. Григорович \*

Дрогобицький педагогічний ліцей,

\*Дрогобицький державний педагогічний університет імені Івана Франка,  
кафедра інформаційних систем і технологій

## ПРОСТОРОВІ БАЗИ ДАНИХ: СУЧАСНИЙ СТАН І ТЕНДЕНЦІЇ РОЗВИТКУ

© Григорович А.Г., Григорович В.Г., 2004

Проаналізовано досягнення та основні напрямки досліджень просторових баз даних. Розглянуто моделі просторових даних: моделі на основі полів, опрацювання растрових зображень, моделі на основі об'єктів, розширення класичних моделей, шарові моделі та розширення об'єктних моделей. Коротко охарактеризовані мови просторових запитів: стандарт OGIS, стандарт SQL3, запити растрової інформації, реалізація опрацювання просторових даних в промислових СУБД. Розглядаються методи зберігання та індексування просторових даних і растрових зображень, а також способи опрацювання та оптимізації запитів. Висвітлюються проблеми просторових мереж, просторового видобування даних та просторових сховищ даних. Аналізуються приклади побудови прикладних систем для опрацювання просторової інформації. Відзначаються перспективні напрямки досліджень в кожній із вказаних областей просторових баз даних.

In the article the authors analyze achievements and the main trends of spatial database research. There are examined spatial data models: field models, bitmap images processing, object models, expansion of classical models, layer models and expansion of object models. There are briefly characterized spatial queries languages: standard OGIS, standard SQL3, bitmap information queries, implementation of spatial data procession in industrial DBMS. There are also examined methods of storage and indexing of spatial data and bitmap images, as well as ways of processing and optimization of queries. The authors focus upon the problems of spatial networks, spatial data mining and spatial data warehouses. They also analyzed samples of creation of application systems for processing of spatial information. There are singled out perspective directions of researches in each of the mentioned fields of spatial databases.

### Постановка проблеми у загальному вигляді

Будь-яка інформація про положення та розміщення в просторі являє собою просторові дані. Вже тривалий час є очевидною необхідність розширення функцій систем управління базами даних (СУБД) для забезпечення підтримки операцій над просторовими даними. Багато існуючих СУБД

(зокрема, Oracle, Informix, DB2) містять інструментальні засоби для опрацювання просторових даних. Отже, можна говорити про виникнення цілої індустрії систем управління просторовими базами даних (СУПБД): як про дослідження моделей та алгоритмів ефективного маніпулювання просторовою інформацією, так і про створення відповідних комерційних комп'ютерних програм.

### Цілі статті

Дослідження в галузі просторових баз даних надзвичайно актуальні. Розроблені моделі даних та алгоритми опрацювання просторової інформації відразу реалізуються і впроваджуються в географічних інформаційних системах (ГІС), системах автоматизованого проектування (САПР), мультимедійних інформаційних системах, сховищах даних, системах спостереження за поверхнею Землі.

Метою статті є огляд досягнень та аналіз основних напрямків досліджень в галузі просторових баз даних.

### Аналіз останніх досліджень

Найбільш систематизований огляд стану і тенденцій розвитку просторових баз даних поданий в книзі SPATIAL DATABASES: A TOUR 1<sup>st</sup>. Edition by SHEKHAN, SHASHI; CAWLA, SANJAY. Prentice Hall, Copyright 2003 by Pearson Education, Inc., перекладений в 2004 р. російською мовою [1].

Дотримуючись загального підходу до систематизації досліджень в галузі просторових баз даних, прийнятого в [1], можна виділити такі напрямки і тематику публікацій:

- моделі просторових даних;
- мови просторових запитів та реалізація опрацювання просторових даних в промислових СУБД;
- зберігання та індексування просторових даних;
- опрацювання та оптимізація запитів;
- просторові мережі;
- просторове видобування даних;
- просторові сховища даних;
- побудова прикладних систем опрацювання просторової інформації.

Розглянемо публікації за кожним із наведених напрямків.

### 1. Моделі просторових даних

Всі моделі просторової інформації можна поділити на дві категорії: *об'єктні* та *польові*. *Польові* моделі найбільш підходять для моделювання явищ, характеристики яких плавно змінюються у просторі, наприклад: кількості опадів, хмарності, кислотності ґрунту, фону радіоактивності тощо. *Об'єктні* моделі найкраще себе проявляють при моделюванні дискретних сутностей, які чітко ідентифікуються та розпізнаються, наприклад: країни на політичній карті, ріки, озера, лісопарки, шосе, залізничні шляхи і т.п. З кожною моделлю пов'язаний свій набір операцій, які характеризують модель та визначаються нею.

#### *Моделі на основі полів*

Згідно з [2], з точки зору *польової* моделі (тобто з *функціональної* точки зору), дані можна змодельовати як функцію, *областю визначення* якої є відповідне географічне місце, а *множиною значень* – множина, яка складається із набору значень вимірюваної характеристики (тобто кількості опадів, висоти над рівнем моря, переважаючої породи дерев тощо). Ця функція (*функція поля*), наприклад,  $f$ , відображає кожну точку простору в один і лише один елемент області значень. Такі функції можна подавати графіками, зрізами за певною координатою, ізолініями.

Опис *польової* моделі вимагає визначення трьох компонентів: *просторової системи координат*, *функції поля* та набору *операцій над полями* [2].

Просторова система координат  $F$  – це скінченна сітка, накладена на базовий простір (наприклад, система географічних координат „широта–довгота”).

Полюва функція  $f$  відображає просторову систему координат  $F$  на область визначення деякого атрибуту  $A$ , тобто  $f: F \rightarrow A$

Зв'язки та взаємодія між полями визначаються *операціями над полями*, які відображають поля на інші поля. Операції над полями поділяються на *локальні*, *фокальні* та *зональні* [2].

Для *локальних* операцій значення результуючого поля в певній точці просторової системи координат залежить лише від значення полів-операндів у цій точці. Наприклад, значення функції „кількість опадів за рік”  $f(x)$  в певній точці  $x$  залежить лише від значень функцій „кількість опадів за зиму”  $f_I(x)$ , „кількість опадів за весну”  $f_{II}(x)$ , „кількість опадів за літо”  $f_{III}(x)$  та „кількість опадів за осінь”  $f_{IV}(x)$ :

$$f(x) = f_I(x) + f_{II}(x) + f_{III}(x) + f_{IV}(x).$$

Це – приклад локальної операції.

При виконанні *фокальних* операцій значення результуючого поля в певній точці залежить від значення вхідного поля в малому околі цієї точки. Нехай вхідним полем є функція висоти над рівнем моря  $h(x,y)$ . Тоді операція, яка розраховує величину нахилу поверхні  $N$ , буде фокальною операцією, бо значення величини нахилу в точці  $(x,y)$ , тобто градієнта висоти, залежить від значення поля висоти в околі точки  $(x,y)$ :

$$N(x,y) = \nabla \cdot h(x,y).$$

*Зональні* операції зв'язані з *агрегуванням*, або *інтегруванням*: наприклад, операція, що розраховує середню річну кількість опадів для кожної області України, – це зональна операція.

*Опрацювання растрових зображень*

Відмінності між векторними та растровими даними детально обговорюються в [3], алгебра карт для картографічного моделювання була введена в [4], алгебра зображень – в [5], геоалгебра – в [6].

Полюві сутності характеризуються тим, що вони – *неперервні*: якщо дві точки із області визначення розміщені близько одна від однієї, то і значення поля в цих точках також мало відрізняються. Важливим наслідком цього є те, що для наведення функції можна використовувати відносно невелику вибірку *середніх* значень поля: функція задається матрицею точок, які її подають. У ГІС така матриця має назву *растра*, а кожна клітинка растра називається *пікселем*.

Бази растрових даних стають надзвичайно важливими: по-перше, саме в такій формі отримується інформація при віддаленому зондуванні; по-друге, обсяг даних, доступних в растровій формі, постійно зростає; по-третє, значна частина растрової інформації не може бути перетворена до векторного подання. Растрові дані можна моделювати як поля.

*Алгебра карт* дає можливість використовувати набір операторів для опрацювання растрових наборів даних.

В алгебрі карт *операндом* є растрова матриця, а операції поділяються на чотири класи: *локальні*, *фокальні*, *зональні* та *глобальні*.

*Локальна* операція відображає один растр на інший так, що значення *клітинки* в *новому* растрі залежить тільки від значення тієї самої клітинки в *початковому* растрі. Прикладом локальної операції є *порогове опрацювання* (рис. 1.1).

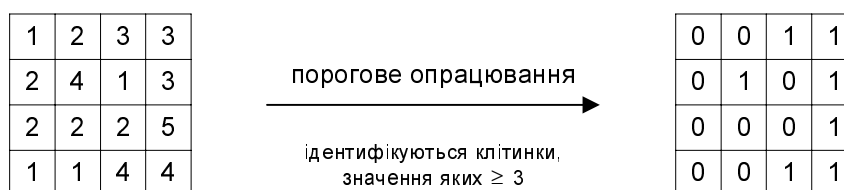


Рис. 1.1. Приклад локальної операції: порогове опрацювання

Для *фокальних* операцій значення клітинки *нового* растра залежить від значення клітинки *початкового* растра, а також сусідніх із нею клітинок. Є три типи *околів* клітинки, за аналогією із шахами їх називають *околами тури, слона та ферзя* (рис. 1.2).

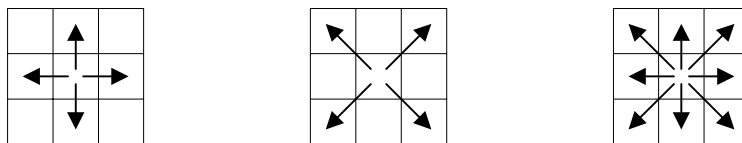


Рис. 1.2. Околи тури, слона та ферзя

Прикладом фокальної операції є розрахунок величини нахилу.

При виконанні *зональних* операцій значення клітинки нового растра є функцією значення тої самої клітинки в *початковому* растрі, а також значення інших клітинок, які знаходяться в *зоні*, заданій ще одним растром (*зональною картою*) – рис. 1.3.

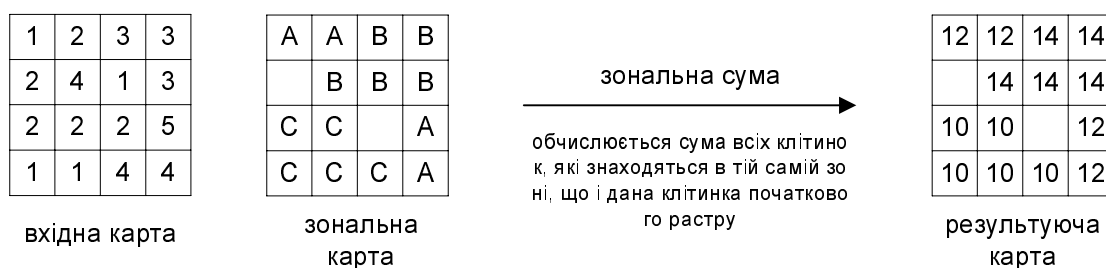


Рис. 1.3. Приклад зональної операції: зональна сума

*Глобальні* операції відрізняються тим, що значення клітинки в *новому* растрі – це функція місця розміщення або значення *всіх* клітинок *початкового* растра.

Крім вказаних чотирьох класів операцій, є ще дві операції, призначені для опрацювання зображень. Це *відсікання (trim)* та *розщеплення (slice)*.

Операція *відсікання* утворює підмножину *початкового* растру, границі якої вирівняні за осягами координат (рис. 1.4).

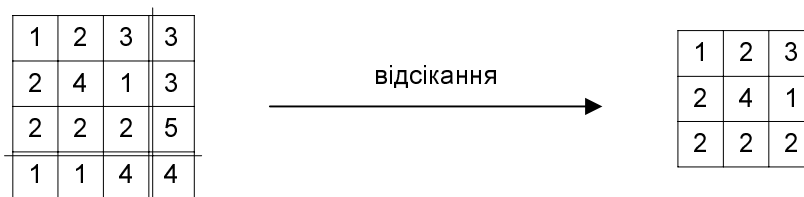


Рис. 1.4. Операція відсікання

Операція *розщеплення* будує проєкцію растрового зображення високої розмірності на одне чи кілька растрових зображень більш низької розмірності (рис. 1.5).

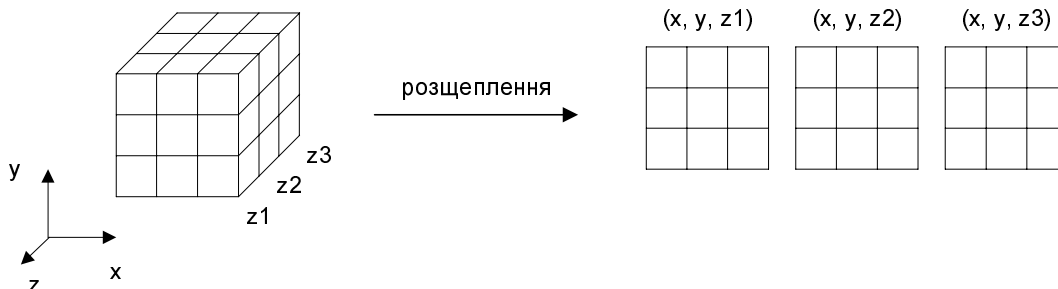


Рис. 1.5. Операція розщеплення: зниження розмірності

### Моделі на основі об'єктів

Топологічні аспекти моделей на основі об'єктів були вперше розроблені в [7], а потім узагальнені в [8]. Обговоренню об'єктної моделі присвячена також робота [9].

Головне питання, яке стосується об'єктних моделей, – це вибір базової множини типів просторових даних. Вирішення проблеми на основі узгодження різних підходів запропонований стандартом OGIS [10], згідно з яким просторові образи передаються *геометрією*, яка описується за допомогою *просторової системи подання* (тобто системи координат).

*Геометрія* ділиться на чотири категорії об'єктів, а саме: *точки* (*point*), *криві* (*curve*), *поверхні* (*surface*) та *геометричні набори* (*geometry collection*). *Точки* описують образи 0-вимірних об'єктів (наприклад, центр міста на карті). *Криві* описують образи 1-вимірних об'єктів (ріки, автодороги на карті). Об'єкти типу *крива* часто апроксимують за допомогою *послідовності ліній* (*linestring*), наведеної за допомогою двох чи більшої кількості точок. *Поверхні* описують образи 2-вимірних об'єктів (країни на карті світу). *Поверхні* часто моделюються *багатокутником* (*polygon*). *Геометричні набори* подають складені об'єкти, такі як група островів тощо. *Геометричні набори* бувають трьох видів: *складена точка* (*multipoint*), *складена крива* (*multicurve*) і *складена поверхня* (*multisurface*).

Відношення, які можуть існувати між об'єктами, визначає базовий простір.

З об'єктною моделлю пов'язані операції, які класифікуються на основі їх математичних властивостей: *множинно-орієнтовані*, *евклідові*, *топологічні* та *метричні*.

Основні операції, допустимі для *множинно-орієнтованого простору*, – це звичайні операції об'єднання, перетину, включення та належності.

Прикладами *топологічних* операцій є *дотикатися*, *знаходиться в межах*, *перекривати*.

Бінарні топологічні відношення між об'єктами  $A$  та  $B$  на площині основані на перетині *границі* ( $\partial A$ ), *внутрішньої* ( $A^\circ$ ) та *зовнішньої* ( $A^-$ ) області об'єкта з *границею* ( $\partial B$ ), *внутрішньою* ( $B^\circ$ ) та *зовнішньою* ( $B^-$ ) областю об'єкта  $B$  [11]. *Матриця дев'яти перетинів* визначає топологічні відношення і може бути наведена так (тут [1] містить помилку):

$$\Gamma_9(A,B) = \begin{vmatrix} A^\circ \cap B^\circ & A^\circ \cap \partial B & A^\circ \cap B^- \\ \partial A \cap B^\circ & \partial A \cap \partial B & \partial A \cap B^- \\ A^- \cap B^\circ & A^- \cap \partial B & A^- \cap B^- \end{vmatrix}$$

Розглядаючи для кожного із дев'яти перетинів вказаної матриці значення „порожня множина” (0) та „непорожня множина” (1), можна виділити  $2^9 = 512$  бінарних відношень. Для двовимірних областей площини можна навести вісім відношень із цих 512, які забезпечують повне взаємовиключне покриття: *не перетинається* (відокремлений), *дотикається*, *перекриває*, *дорівнює*, *містить*, *всередині*, *покриває*, *покритий* (рис. 1.6).

Подібно можна описати топологічні операції над іншими парами типів просторових даних: (*точка, точка*), (*точка, крива*), (*точка, поверхня*), (*крива, крива*), (*крива, поверхня*).

Топологічним зв'язкам та їх вираженню в просторових базах даних присвячена робота [12]. Проблеми топологічних структур в просторових базах даних розглядаються в [13].

Слід зазначити, що не розглядаються топологічні відношення між об'єктами розмірності більше двох.

*Простір напрямків* визначає відношення, зв'язані з напрямками. Їх можна поділити на три типи: *абсолютні* (в глобальній системі координат: „на північ”), *відносно об'єкта* або *відносно спостерігача* („зліва”, „перед”).

*Метричний простір* та *евклідовий простір* визначають відповідно метричні та евклідові операції.

Всі попередні операції були *статичними* – вони не впливали на операнди.

*Динамічні* операції змінюють об'єкти, над якими вони виконуються, три основні динамічні операції – це *створення*, *знищення* та *оновлення*.

Операція *злиття* (*merge*) – це операція рекласифікації карти, яка підтримується багатьма ГІС.

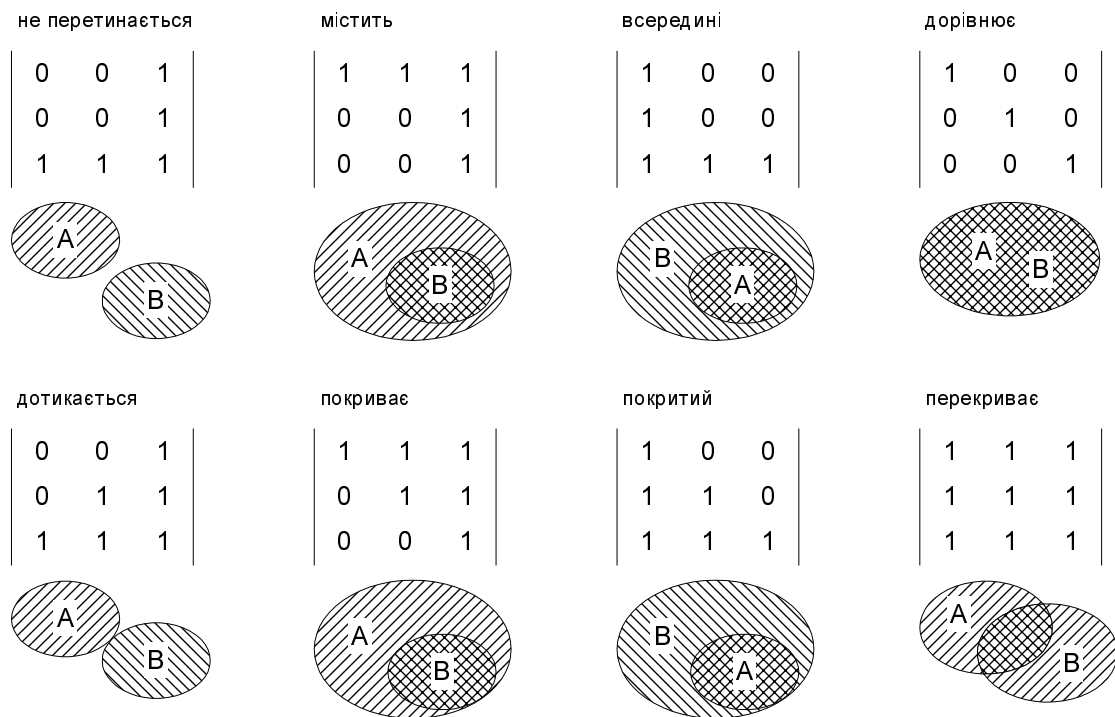


Рис. 1.6. Модель дев'яти перетинів [11]

В основу ще одного класу просторових операцій покладене відношення *видимий*.

Імовірнісні просторові операції розглядаються в роботі [14], запропонована в ній модель описує об'єкти, просторові характеристики яких точно не відомі.

Просторові типи даних розглядаються також в [15], де, крім подання геометрії просторових об'єктів, вводиться поняття *покриття*, – об'єкт, який має властивість, що називається *функцією покриття*, область визначення якої – це деяка просторова область, а множиною значень може бути будь-яка множина. Покриття поділяється на ряд важливих підтипів, зокрема: растровий образ, сітка, покриття дискретними точками, покриття лінійними ланцюгами, тіл-покриття, покриття геометрією, поверхня, покриття багатогранниками, найближчий сусід і втрачена область, покриття сегментованими лініями.

Розглядаються три класи *операцій* над просторовими даними:

- просторові предикати, які виражають відношення геометрії;
- операції, які повертають атомарні просторові дані;
- операції, які повертають числа.

Також описується алгебра ROSE, основана на ґратках (realm) як області визначення типів даних.

*Розширення класичних моделей*

Модель „сутність–зв'язок” та реляційну модель необхідно розширити для того, щоб вони могли відображати особливі характеристики просторових даних [1].

Модель „сутність–зв'язок” можна розширити шляхом введення *пиктограм*, які символізують різні типи просторових даних та зв'язків. Розширення моделі „сутність–зв'язок” для просторових даних досліджені в [16, 17]. Пиктографічне розширення моделі „сутність–зв'язок” (PEER), тобто логічна модель, яка розширює модель „сутність–зв'язок” просторовими пиктограмами, призначена для просторових баз даних, наведено в [18].

UML – одну із мов концептуального моделювання об'єктно-орієнтованих даних – також можна розширити шляхом введення пиктограм для моделювання просторової інформації. Моделювання баз даних для ГІС з використанням сховищ на основі UML обговорюється в [19].

Реляційну модель можна розширити шляхом введення нових *типів даних* та зв'язаних із ними операцій. Для подання просторової інформації використовуються три основних типи даних – точка, лінія та багатокутник.

Активна пропаганда розширення реляційної моделі новими типами даних ведеться в [20]. Слід зазначити, що є принципова відмінність між традиційними та об'єктними моделями, яку К. Дж. Дейт та Х. Дарвен залишають поза увагою. Ця відмінність полягає у різному підході до моделювання та проектування бази даних: якщо традиційний підхід оснований на структурній декомпозиції, то об'єктно-орієнтований базується на синтезі ієрархії. Після побудови моделі "сутність-зв'язок" у межах традиційного підходу ми маємо справу із *статичним* відбитком предметної області. При об'єктно-орієнтованому проектуванні ми завжди враховуємо *поведінку* об'єкта (а не лише його значення в даний момент).

#### *Шарові моделі та розширення об'єктних моделей*

Цікавий спосіб організації даних описаний в роботі [21], в якій автори розглядають два підходи до організації просторових даних: пошаровий та об'єктно-орієнтований.

У *пошаровому* способі всі просторові об'єкти поділяються на тематичні шари (окремо – ріки, окремо – автомобільні дороги і т. п.). Як правило, для зручності в один шар поміщаються лише об'єкти певних геометричних типів.

Привабливою є ідея поєднати пошаровий спосіб та об'єктну модель даних. Це дасть змогу використовувати зручність об'єктної моделі та надасть об'єктам змісту завдяки розподілу за тематичними шарами. Якщо ж в об'єкти включити властивість типу *покриття*, то появиться можливість в об'єктній технології опрацьовувати *польові* дані (наприклад, *растри*).

Модель багатьох наведень просторових даних, яка дозволяє реалізувати різні рівні деталізації, висвітлюється в роботі [22].

Гібридна модель, основана як на ієрархічному методі, так і на методі координат положення об'єктів, подана в роботі [23]. Це розширення збільшує гнучкість та покращує функціонування прикладних систем.

## **2. Мови просторових запитів**

### **Стандарт OGIS**

Стандарт OGIS містить рекомендований набір просторових типів даних та функцій, які вважаються необхідними для опрацювання запитів до просторової інформації.

Специфікація OGIS обмежена *об'єктною* моделлю простору.

Операції, визначені в стандарті OGIS, поділяються на три категорії:

1. *Базові операції*, які можна застосувати до *всіх геометричних типів даних*; наприклад, операція **SpatialReference()** повертає базову систему координат геометрії об'єкта.
2. Операції, які виявляють *топологічні відношення* між просторовими об'єктами; наприклад, операція **Overlap** перевіряє, чи внутрішні області двох об'єктів мають непорожній перетин.
3. Загальні *операції просторового аналізу*; наприклад, операція **Distance** повертає найкоротшу відстань між двома просторовими об'єктами.

Операції OGIS навіть у межах об'єктної моделі обмежені запитом ВИБІРКИ – ПРОЕКЦІЇ – З'ЄДНАННЯ. У стандарті OGIS відсутня підтримка цілого класу метричних операцій, а саме операцій на основі *напрямку*; не підтримуються динамічні операції, основані на понятті видимості.

### **Стандарт SQL3**

SQL3 (SQL-99) – стандартизована платформа об'єктно-реляційного розширення мови SQL. Ця платформа не орієнтована виключно на ГІС чи просторові бази даних, вона охоплює об'єктно-реляційні бази даних загального призначення. Найбільш природний варіант її впровадження – це реалізація стандарту OGIS як складової частини мови SQL3 [1]. Підмножини цього стандарту вже реалізовані в комерційних СУБД (наприклад, Oracle 8 компанії Oracle та DB2 корпорації IBM).

Мова SQL3 (SQL-99) пропонує два основні розширення SQL2 (SQL-92), які можуть виявитися корисними при роботі з просторовими даними:

1. *Абстрактні типи даних (АТД)*. АТД можна визначити за допомогою команди CREATE TYPE. Як і класи в об'єктно-орієнтованому підході, АТД складається із полів (атрибутів) та методів (функцій-членів), призначених для доступу до значень атрибутів:

```
CREATE TYPE Point(
  x NUMBER,
  y NUMBER,
  FUNCTION Distance (:u Point, :v Point) RETURN NUMBER
);
```

двокрапки перед *u* та *v* вказують, що це – локальні змінні.

2. *Тип рядка* (row type) – це тип відношення, який визначає схему відношення:

```
CREATE ROW TYPE Point(
  x NUMBER,
  y NUMBER,
);
```

тепер можна створити таблицю, яка містить екземпляри типу рядка:

```
CREATE TABLE PointTable OF TYPE Point;
```

Для реалізації просторових даних та операцій над ними більш придатні абстрактні типи даних.

Розширення SQL для просторових завдань викладені в [24]. Нормативний документ консорціуму OGIS [25] – спроба узгодження з мовою SQL просторових розширень. Приклади мов запитів для аналізу просторових даних наведені в [26].

#### *Запити растрової інформації*

Операції *алгебри карт* аналізують дані, а не *опрацьовують запити до них*. Ключова проблема полягає в ефективній вибірці із бази даних тих зображень, які задовольняють предикати, задані в запиті. В основу предиката можуть бути покладені метадані чи зміст зображення, тобто запити елементів просторових даних можна виконувати на основі *метаданих* або *за змістом* [1].

*Метадані* – це опис зображення за допомогою більш простих типів даних (числових чи текстових). Запити, сформульовані на основі метаданих (зокрема, за датою запису, типу давача, положенню в просторі тощо) можуть опрацьовуватися за допомогою традиційних методів. Вибірка *за змістом*, або *асоціативна вибірка*, вимагає нових підходів до формулювання та опрацювання запитів.

Запити асоціативної вибірки часто спрямовані на пошук найбільш подібних елементів растрових даних, а не на пошук тих елементів, які задовольняють всі предикати запиту. Методологія асоціативної вибірки містить п'ять етапів і полягає у пошуку таких відношень між просторовими об'єктами:

1. *Топологічна подібність*, основана на топологічних відношеннях, граф топологічних околів яких поданий в роботі [27] (рис. 2.1). Тоді відстань між двома відношеннями  $D_{\text{топ}}(T1, T2)$  дорівнює кількості ребер в найкоротшому шляху по графу топологічних околів між  $T1$  і  $T2$ .

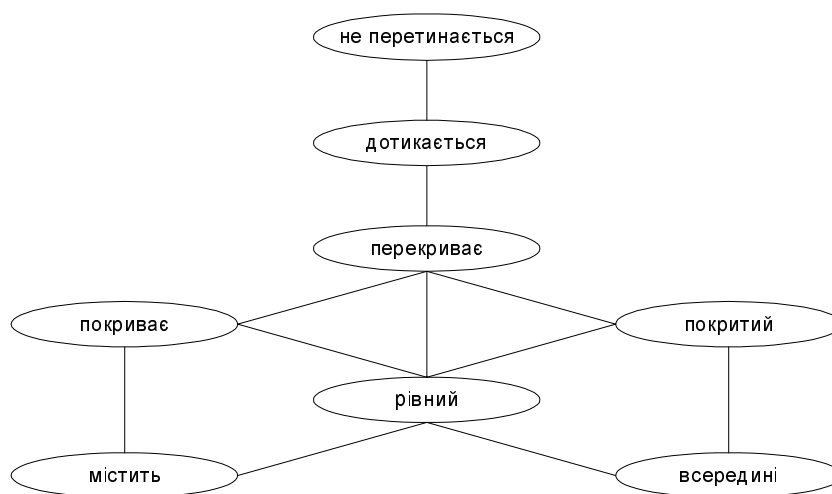


Рис. 2.1. Граф топологічних околів [27].

Відстань між двома відношеннями дорівнює найкоротшому шляху за цим графом



2. *Подібність за напрямом* – це поняття, подібне до топологічної подібності. Вісім відомих напрямів – це *північ, північний захід, захід, південний захід, південь, південний схід, схід, північний схід*. Для неточкових об'єктів (лінія, багатокутник) при асоціативній вибірці кращим є підхід, який втілює нечітке поняття відношення напрямку (де саме закінчується *північ* і починається *північний захід*?). Ідея полягає у використанні графа околів відношень напрямку [27].

3. *Подібність за відстанню* – найменш важлива серед всіх трьох класів просторових відношень: *напрямку, топологічних та метричних відношень*, особливо якщо топологічні відношення та відношення напрямку – незмінні. Для оцінки *подібності за відстанню* вибирається квадрат евклідової відстані між центрами мас об'єктів [27].

4. *Граф відношень атрибутів* (ARG, attribute relational graph) – повністю зв'язний граф, який містить інформацію про об'єкти та зв'язки між ними [28]. У вузлах графа відношення атрибутів проставляються мітки об'єктів, а ребра між двома вузлами помічаються інформацією про зв'язки між вузлами. Кожна мітка несе таку інформацію про зв'язки між об'єктами: 1) тип топологічного відношення (наприклад, *не перетинається*); 2) кут між ними; 3) відстань між об'єктами. Після побудови графу відношень атрибутів для кожного зображення він відображається в точку багатовимірного простору властивостей. Точки цього простору наперед організовані: перший об'єкт, потім всі відношення між ним та іншими об'єктами, потім – другий об'єкт і всі його відношення з наступними об'єктами.

5. Нарешті, *етап вибірки*, який передбачає: по-перше, відображення зображень з бази даних в точки багатовимірного простору властивостей (при цьому для кожного зображення потрібно побудувати граф відношень атрибутів); по-друге, визначення метрики згідно з критерієм *подібності* зображень; по-третє, побудову методу зберігання (наприклад, R-дерева) для кластеризації та індексування багатовимірних точок властивостей; по-четверте, відображення шуканого зображення в точку чи область простору властивостей і відбір точок, розміщених *поблизу* шуканої точки чи *в межах* шуканої області; по-п'яте, вивести як результат зображення, яке відповідає знайденим точкам.

Отже, просторову подібність елементів растрових даних можна визначити за допомогою графу відношень атрибутів. Цей граф є моделлю просторових об'єктів, що знаходяться у відповідних елементах растрових даних, а також їх просторових відношень (топології, напрямку, відстані тощо). Запити асоціативної вибірки можуть опрацьовуватися знаходженням графів відношення атрибутів, найбільш подібних до таких самих графів із складу запиту.

Останнім часом особливо багато уваги приділяють асоціативній вибірці. У [29] опубліковані результати досліджень асоціативної вибірки стосовно мультимедійних баз даних. Просторові аспекти асоціативної вибірки викладені в [30] та [27]. Індексвання графів відношень атрибутів наведено у [28]. Система на основі просторових запитів за загальним описом (spatial-query-by-sketch) для ГІС наведена в [31], вибірка аерофотознімків на основі текстур описана в [32].

#### *Реалізація опрацювання просторових даних у промислових СУБД*

Як зазначалося вище, деякі можливості опрацювання просторових даних вже реалізовані у промислових СУБД – зокрема, Oracle, Informix, DB2.

Особливості СУБД Oracle 8i Spatial при роботі з просторовими даними описуються в [33], тут же згадуються нові типи даних: „точка”, „лінія”, „полігон”, які характерні для більшості існуючих ГІС. Крім того, в СУБД Oracle 8i Spatial передбачені деякі додаткові типи просторових об'єктів, наприклад, „круг”, „дуга”.

Характерним є те, що просторові та атрибутивні дані в СУБД Oracle 8i Spatial зберігаються в одній і тій самій базі (класичні ГІС зберігають дані цих видів окремо – і це інколи приводить до проблем, які не можна вирішити).

Діалект SQL доповнений в Oracle 8i Spatial новими операторами і функціями: наприклад, функція SDO\_GEOM.SDO\_POLY\_INTERSECTION генерує полігон, утворений в результаті перетину двох просторових об'єктів, а функція SDO\_GEOM.AREA обчислює площу двовимірного об'єкта.

Просторові запити в Oracle 8i Spatial опрацьовуються за дворівневою схемою фільтрації-очистки.

### 3. Зберігання та індексування просторових даних

Виникає багато проблем при використанні традиційних методів зберігання та доступу для опрацювання просторової інформації. Проблеми виникають насамперед тому, що ці методи були розроблені для скалярних ключів з нестрогим порядком, який не можна безпосередньо вказати для просторових ключів, визначених на багатовимірному просторі.

Метою *кластеризації* є скорочення часу пошуку  $t_s$  і часу затримки  $t_l$  при формуванні результатів запитів, які часто зустрічаються.

Для просторових об'єктів це означає, що об'єкти, які є сусідніми в просторі і які часто разом шукаються, повинні зберігатися один поряд з другим у дисковій пам'яті. Є три типи кластеризації [34]:

1. *Внутрішня кластеризація* – сторінка диску (мінімальна одиниця передачі даних між диском і оперативною пам'яттю; ціле число, кратне розміру сектора) виділяється для зберігання єдиного об'єкта – якщо розмір об'єкта менший ніж об'єм сторінки, інакше об'єкт зберігається на кількох послідовних сторінках.

2. *Локальна кластеризація* – кілька просторових сусідніх об'єктів розміщується на одній сторінці.

3. *Глобальна кластеризація* – просторово сусідні об'єкти зберігаються на кількох послідовних сторінках, які можна прочитати за одне зчитування з диску.

*Крива упаковки (Z-крива, крива Гільберта)* задає нестрогий порядок на множині точок багатовимірного простору. Цей нестрогий порядок використовується для індексування просторових точкових об'єктів за допомогою  $B^+$ -дерева. Крива упаковки використовується для перетворення просторових запитів у запити діапазонів, адресованих  $B^+$ -дереву. Властивості групування кривої упаковки Гільберта обговорюються в [35]. Просторова кластеризація висвітлюється в [34, 36].

Алгоритм ієрархічної кластеризації, оснований на відношенні „сусідній”, наведено в роботі [37].

Для роботи з точками простору використовується *файл решітки* – просторова індексна структура, де кожний просторовий вимір розділяється незалежно від інших і точки розділення зберігаються у вигляді векторів вимірів, які визначають просторову структуру решітки. Каталог решітки містить вказівники на фізичні сторінки вторинної пам'яті. Просторове індексування детально досліджується в [38] та розглядається в [39].

$R$ -дерева – це збалансовані дерева пошуку з прямокутними ключами. Листові вузли  $R$ -дерева містять ідентифікатори та мінімальні обмежуючі прямокутники просторових об'єктів. Проміжні вузли містять адреси та мінімальні обмежуючі прямокутники дочірніх вузлів (рис. 3.1, 3.2).  $R$ -дерева були введені в [40].

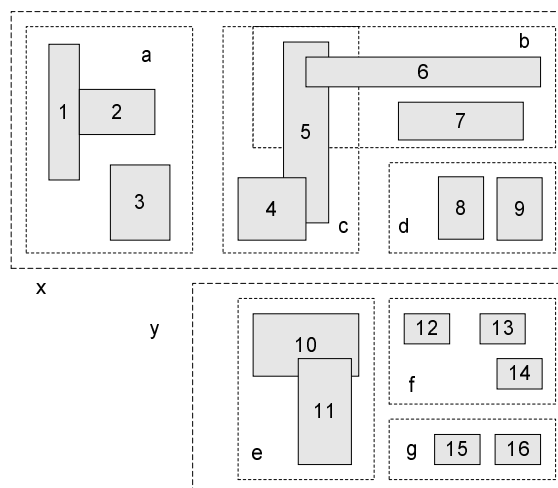


Рис. 3.1. Набір просторових об'єктів

Дерево з  $R$ -зв'язками забезпечує ефективне блокування при реалізації паралельного доступу до індексної структури  $R$ -дерева (рис. 3.3). Для прискорення просторового з'єднання можна використовувати індекс з'єднання.

Індекс з'єднання описує зв'язок між об'єктами двох відношень (наприклад,  $R$  та  $S$ ).

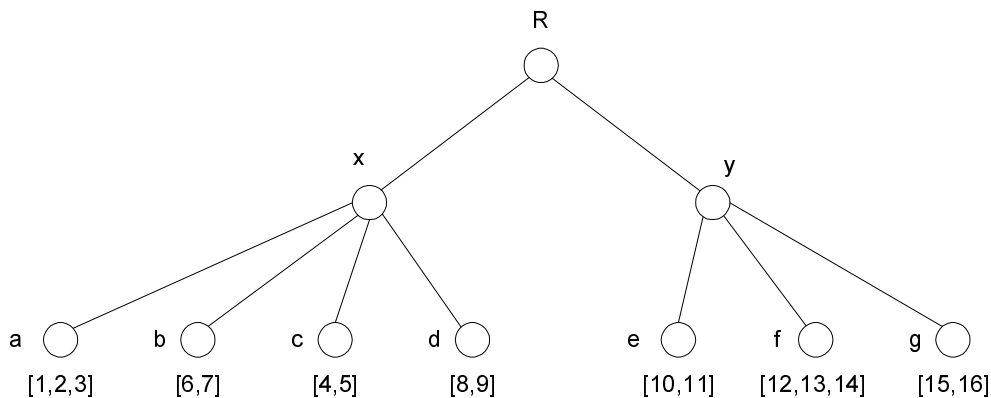


Рис. 3.2. Ієрархія R-дерев

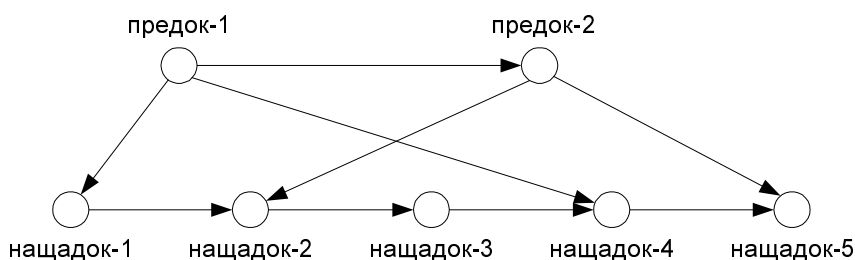


Рис. 3.3. Фрагмент дерева з R-зв'язками [41]

Розглянемо з'єднання  $R$  та  $S$  за атрибутом  $A$  із  $R$  та атрибутом  $B$  із  $S$ . Якщо  $F$  – предикат з'єднання, то індекс з'єднання визначається у вигляді

$$JI = \{(r_i, s_j) \mid F(r_i.A, s_j.B) \text{ – істинне для } r_i \in R \text{ і } s_j \in S\}$$

Індекс з'єднання можна описати дводольним графом  $G=(V_1, V_2, E)$ , де  $V_1$  містить ідентифікатори кортежів відношення  $R$ ,  $V_2$  – ідентифікатори кортежів відношення  $S$ . Множина ребер  $E$  містить ребра  $(v_r, v_s)$  для  $v_r \in R$ ,  $v_s \in S$ , якщо в індексі з'єднання існує кортеж, що відповідає  $(v_r, v_s)$ .

Управління паралельністю щодо просторових баз даних залишається актуальною темою досліджень. Використанню дерев з R-зв'язками для реалізації блокування присвячена робота [42]. В [43] обговорюється проблема управління паралельними процесами в методах багатовимірного доступу. Індекс з'єднання широко використовується в галузі організації сховищ даних, його використанню для просторових даних присвячена робота [44].

Для індексування ребер та вузлів неточкових просторових об'єктів (багатокутників) використовуються як декомпозиція об'єкта на ряд простих складових частин (наприклад, трапецій), так і TR\*-дерев – варіант, призначений для мінімізації операцій в оперативній пам'яті. Основною характеристикою TR\*-дерев є мале максимальне число елементів, яке припадає на один вузол. Подання об'єкта у вигляді TR\*-дерев повністю завантажуються в оперативну пам'ять.

#### Зберігання та індексування растрових зображень

Для обслуговування запитів важливо, щоб растрові зображення зберігалися в базі даних, а не у вигляді окремих файлів файлової системи. Оскільки растрові зображення займають дуже багато дискової пам'яті, то важливо мати особливу стратегію їх зберігання. Таких стратегій є декілька. По-перше, в таблиці разом з іншими атрибутами потрібно зберігати лише важливу інформацію про масив, а саме зображення записувати в окрему таблицю. Щоб реалізувати зв'язок із зображенням, в першу таблицю включають зовнішній ключ – вказівник на таблицю, яка містить зображення.

Поширеним способом зберігання растрових наборів даних є подання за допомогою *мозаїки* – це інша *стратегія*, яка ефективна для великих масивів інформації. Вона полягає у декомпозиції масивів на фрагменти мозаїки і зберіганні цих фрагментів окремо, у вигляді кортежів відношення. Це особливо корисно, коли запит має повернути лише підмножину початкового масиву (як для операції відсікання).

Ще одна *стратегія* оснований на *стискуванні даних*: набір зображень апроксимують меншим набором ортогональних зображень, які можна стиснути, що зменшує об'єм дискового простору, необхідного для зберігання зображень.

Робота [45] присвячена індексації, оснований на сигнатурах, для пошуку в базах даних, які містять двовимірні зображення.

#### 4. Опрацювання та оптимізація запитів

Запити до СУПБД, як і запити до класичних СУБД, виражені в декларативній формі. Сама СУПБД містить компоненти, які відповідають за ефективне виконання запитів.

Просторові операції поділяються на чотири групи [1]. Всі просторові складові складних запитів, як правило, будуються на основі елементів цих груп [39]:

1. *Операції оновлення* – стандартні операції баз даних, такі як зміна, створення і т. д.

2. *Операції вибірки* бувають двох видів:

а) *точкові запити* (*PQ, point queries*):

Дана точка  $p$ , знайти всі просторові об'єкти  $O$ , в яких вона міститься:

$$PQ(p) = \{ O \mid p \in O.G \neq \emptyset \},$$

де  $O.G$  – геометричне місце об'єкта  $O$ .

б) *запити діапазонів, запити областей* (*RQ, range queries*):

Для даного багатокутника  $P$  знайти всі об'єкти  $O$ , які перетинають  $P$ :

$$RQ(P) = \{ O \mid O.G \cap P.G \neq \emptyset \}.$$

3. *Просторове з'єднання* (одна із найважливіших операцій) – з'єднання за просторовим предикатом  $\theta$ :

$$R \bowtie_{\theta} S = \{ (O, O') \mid O \in R, O' \in S, \theta(O.G, O'.G) \}.$$

Приклади предиката  $\theta$ : *перетинається, містить, оточення, відстань, на північний захід, суміжний, дотикається, перекривається*.

4. *Просторові агрегати* використовуються, як правило, в задачі пошуку *найближчих сусідів*: для даного об'єкта  $O'$  знайти всі об'єкти  $O$ , розміщені на мінімальній відстані від  $O'$ :

$$NNQ(O') = \{ (O) \mid \forall O'': \text{dist}(O'.G, O.G) \leq \text{dist}(O'.G, O''.G) \}.$$

Характерною особливістю просторових запитів є те, що вони вимагають великих затрат як ресурсів процесора, так і системи введення-виведення. Для опрацювання просторових запитів використовується технологія *фільтрації-очистки*, призначена для мінімізації затрат процесора та затрат введення-виведення.

На етапі *фільтрації* для подання просторових об'єктів використовуються простіші об'єкти, наприклад, мінімальні обмежуючі прямокутники. Результат перевірки на основі апроксимацій – це множина “кандидатів”, тобто надмножина реальної результуючої множини. На етапі *очистки* вивчаються точні геометричні контури кожного елемента із множини “кандидатів” і розраховуються точні значення геометричних предикатів – це, як правило, потребує значних затрат ресурсів процесора.

Двоетапна методика фільтрації-очистки та алгебра просторових операторів описані в [46, 47]. Алгоритми просторового з'єднання описані в [48, 49, 50, 51, 52, 53]. Просторове з'єднання у

випадку, коли жодне із відношень-операндів не має індексу, описане в [54]. Опрацювання просторового з'єднання за відстанню (spatial distance join) викладене в [55].

Для кожного конкретного запиту існує кілька типових стратегій виконання. Кожна СУБД (СУПБД) має в своєму складі компонент – *оптимізатор запитів*, який генерує кілька планів виконання запиту, а потім вибирає серед них оптимальний (або майже оптимальний). Існує два відомі способи оптимізації запитів: *логічне перетворення* та *оптимізація на основі затрат*.

На етапі *логічного перетворення* дерево запиту, отримане модулем синтаксичного аналізу, перетворюється до еквівалентного запиту. *Динамічне програмування* – це спосіб визначення оптимальної стратегії при наявності набору планів виконання запиту. Оптимальний розв'язок визначається на основі *функцій затрат* – той план, який мінімізує функцію затрат, визнається оптимальним [56].

Важливі досягнення в галузі оптимізації просторових запитів описано в [57, 58]. Аналітичні моделі оцінювання затрат у запитах вибірки і з'єднання при використанні структур на основі R-дерева обговорюються в [59].

Розвиток сучасних СУПБД пов'язаний із швидким зростанням обсягів даних та поширенням WWW, тому сучасні СУПБД характеризуються паралелізмом та розподіленою архітектурою. Консорціум OGIS випустив стандарти WMS та GML, сумісність з якими спрощує взаємодію та сприяє розвитку сумісних систем. Потреба візуалізації місцевості в реальному часі – одна із причин, що зумовлює розвиток технологій паралельного опрацювання просторових запитів. Детально паралельні ГІС обговорюються в [60].

## 5. Просторові мережі

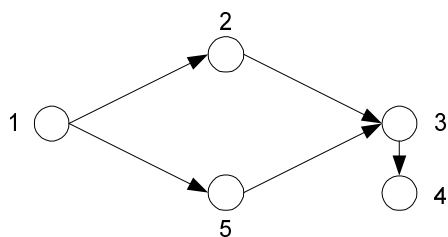
Одним із напрямків використання просторових БД, який найшвидше розвивається, є *просторові мережі*, наприклад, карти доріг. Просторово-мережні дані часто моделюються за допомогою *графів*, вузлами яких є точки простору. Для доступу до просторових мереж з метою розрахунку маршруту та обчислення найкоротшого шляху використовуються операції *get-a-Successor()* та *get-Successors()*. Їх реалізація оснований на зв'язності вузлів, а не на відстані між ними [1]. Моделі даних на основі графів обговорюються в [61].

У реляційній базі даних граф можна подати за допомогою матриці суміжності, матриці інцидентій, списку суміжних вершин, таблиці вузлів (номер вузла, x-координата, y-координата), таблиці ребер (номер вершини-джерела, номер вершини-приймача, відстань), денормалізованої таблиці вузлів (номер вузла, x-координата, y-координата, список наступних вузлів, список попередніх вузлів). Подання за допомогою таблиці вузлів чи таблиці ребер дає можливість виконувати прості операції типу *get-a-Successor()*. Проте мови запитів, основані на реляційній алгебрі (SQL), не можуть виконати багато графових операцій, які потрібні для знаходження *транзитивного замикання* (наприклад, розрахунок найкоротшого шляху). Операція транзитивного замикання відносно недавно була введена в мови запитів до баз даних.

Проблема полягає в тому, що коли відсутня додаткова інформація про структуру графа  $G$ , то, використовуючи операції реляційної алгебри, вивести транзитивне замикання  $G^*$  із  $G$  неможливо. Зокрема, для знаходження транзитивного замикання повинна бути відома довжина найдовшого шляху.

Вивести транзитивне замикання  $G^*$  із  $G$  (рис. 5.1) можна такою послідовністю операцій реляційної алгебри:

1.  $G_1 = G$  та  $G_2 = G$
2.  $T_0 = G_1 \bowtie_{G_1 \text{ .приймач} = G_2 \text{ .джерело}} G_2$
2.  $T_1 = \pi_{G_1 \text{ .джерело}, G_2 \text{ .приймач}}(T_0)$
3.  $T_2 = G \cup T_1$
4.  $T_3 = G \bowtie_{G \text{ .приймач} = T_2 \text{ .джерело}} T_2$
5.  $T_4 = \pi_{G \text{ .джерело}, T_3 \text{ .приймач}}(T_3)$  (тут у [1] помилка)
6.  $G^* = T_2 \cup T_4$

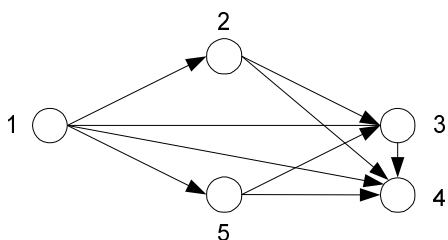


а) граф  $G$

$R$

джерело	приймач
1	2
1	5
2	3
3	4
5	3

б) реляційна форма



в) граф  $G^*$  – транзитивне замикання  $G$

$X$

джерело	приймач
1	2
1	5
2	3
3	4
5	3
1	3
2	4
5	4
1	4

г) транзитивне замикання в реляційній формі

Рис. 5.1. Відношення  $R$  та його транзитивне замикання  $X$

Послідовність кроків, які потрібно зробити для отримання транзитивного замикання, містить дві операції з'єднання. Це пов'язано з тим, що найдовший шлях в графі  $G$  проходить через два проміжні вузли. Якщо інформація про кількість проміжних вузлів відсутня, то жодною операцією реляційної алгебри не можна визначити, чи дійсно отримано транзитивне замикання.

Транзитивним відношенням  $X$  для заданого відношення  $R$  є найменший розв'язок рекурсивного рівняння

$$X = X \cup (R \bowtie X),$$

який містить нерухому точку.

Для роботи з операцією транзитивного замикання в стандарті SQL3 запропонована операція рекурсії RECURSIVE.

Використанню транзитивного замикання присвячена робота [62].

Спеціалізовані програми, які належать до ГІС (ESRI Network Engine [63]) також підтримують просторові мережі і виконують операції над такими даними. Диспут щодо мережних розширень мов запитів продовжується в [64, 65].

Користуючись об'єктно-реляційними можливостями СУБД Oracle, можна сконструювати абстрактні типи даних для реалізації класів *Vertex* (Вершина), *Edge* (Ребро), *Graph* (Граф) [1].

Для розрахунку транзитивного замикання використовується стратегія пошуку вглиб чи вшир. Часткове транзитивне замикання часто обчислюється за алгоритмом Дейкстри. Існують спеціальні стратегії ( $A^*$ , ієрархічний алгоритм), призначені для знаходження найкоротшого шляху.

Ієрархічні алгоритми розбивають великий граф на *граф границь* та *графи-фрагменти*, кожний з яких набагато менший від початкового графа; тим самим зменшуються затрати введення-виведення та оперативної пам'яті. Це особливо корисно для опрацювання запитів до тих графів, які занадто великі для того, щоб їх можна було повністю розмістити в оперативній пам'яті. Основна ідея ієрархічного алгоритму знаходження найкоротшого шляху полягає у декомпозиції початкового графа на набір менших *графів-фрагментів* та зведений *граф границь*. Належна побудова *графа*

*границь* дозволяє розбити запит найкоротшого шляху до *початкового графа* на ряд запитів до *графів-фрагментів*, зберігаючи властивість оптимальності.

Методи зберігання та доступу, призначені максимізувати відносну зв'язність (характеристика, яка не залежить від структури даних) повинні забезпечити найбільшу ефективність введення-виведення при виконанні операцій над графами. Для забезпечення максимального значення відносної зв'язності використовують спеціальні методи кластеризації (ССАМ, Connectivity Cluster Access Method), які групують записи про вузли за допомогою алгоритмів розрізання графів, які зменшують кількість зв'язків між частинами.

Задача про *розрізання графа* полягає в такому розподілі вузлів графа із зваженими ребрами за підмножинами заданого розміру, який мінімізує сумарну вагу всіх ребер, що розрізаються (тобто ребер, що з'єднують два вузли із різних частин графа). Розмір таких підмножин обмежений розмірами сторінки диска. Зараз існує велика кількість прикладних пакетів, які можуть швидко розрізати надзвичайно великі графи (включно до *мільйона* вузлів). Один із поширених способів – використання методу ССАМ, який оснований на мінімізації кількості зв'язків між вузлами.

У роботах [66, 67] обговорюються алгоритми на графах, причому значна увага приділяється затратам введення-виведення. Метод доступу ССАМ, розроблений для просторово-мережних баз даних, описаний в [68].

## 6. Просторове видобування даних

Видобування даних – це галузь, що стосується теорії та практики управління базами даних, статистики та штучного інтелекту [1]. Вона надає дослідникам напівавтоматичні методи виявлення шаблонів у надвеликих обсягах інформації.

Видобування даних – це процес знаходження нетривіальних і корисних інформаційних *шаблонів* у великих базах даних, шукані *шаблони* – це *самородки* цінної інформації, скриті серед великого обсягу даних. Широко відомим став шаблон, який був виявлений в базі даних транзакцій національної компанії роздрібної торгівлі: „люди, які після обіду купують дитячі пелюшки, схильні також купувати пиво”. Цю знахідку з користю використала компанія, отримавши додатковий прибуток від перестановки товарів на полицях магазинів.

Просторове видобування даних – це та галузь видобування даних, яка присвячена розробці способів швидкого аналізу просторових даних. Просторове видобування даних може вплинути на такі наукові напрями, як вивчення глобальної зміни клімату, генетику тощо. Прототип системи просторового видобування даних описаний в [69].

Завдання просторового видобування даних полягає в автоматизації виявлення потенційно корисних шаблонів (залежностей між просторовими даними), які потім можуть бути вивчені фахівцями для подальшої перевірки та підтвердження.

Характерну особливість просторового видобування даних можна коротко висловити законом географії: все зв'язано з усім, проте близькі речі та явища зв'язані між собою сильніше, ніж далекі. Внаслідок цього припущення про незалежність та однаковий розподіл випадкових величин, характерне для класичного видобування даних, не можна застосовувати до видобування просторової інформації. Цю властивість просторових даних називають *просторова автокореляція*.

Просторові дані не тільки не є *незалежними*, вони ще й розподілені *за різними законами* – ця властивість називається *просторовою неоднорідністю*.

Просторово-статистичні методи викладені в [70].

Існують такі основні способи видобування даних [1]: *правила асоціації, кластеризація, класифікація та регресія*.

Мета *класифікації* – оцінка значення одного атрибуту відношення на основі значень інших атрибутів цього відношення. Класифікувати означає знайти функцію

$$f: D \rightarrow L.$$

Тут  $D$  (область визначення  $f$ ) – простір даних атрибутів,  $L$  – множина міток. Мета задачі класифікації – визначити функцію  $f$  для даної скінченної *тренувальної* підмножини

$$Train \subset D \times L.$$

Успішність класифікації пов'язана з точністю функції  $f$  при її застосуванні до *тестового* набору даних, який відрізняється від *тренувальної* множини. Оскільки функція  $f$  використовується для передбачення класів  $L$  у випадку, коли відомі лише дані із  $D$ , то задачу класифікації називають „*побудовою моделі передбачення*”. Є дуже багато методів розв'язання задач класифікації:

- метод максимальної правдоподібності полягає у повному описі розподілу ймовірності сумісних подій  $P(D, L)$  (при цьому використовується теорема Байеса), це – визнаний метод класифікації при віддаленому зондуванні.

- метод дерева розв'язків – один із найпростіших – ділить простір атрибутів на області і присвоює кожній області власну мітку.

- нейронні мережі – це узагальнення дерев розв'язків у випадку нелінійних границь.

- метод регресійного аналізу – створення моделі взаємозв'язку  $D$  та  $L$  за допомогою рівнянь. Лінійний регресійний аналіз використовує лінійні рівняння.

*Правила асоціації* – це виявлення зв'язків між атрибутами відношення. Через високу обчислювальну ефективність правила асоціації часто використовуються при відборі ймовірних підмножин для більш строгого статистичного кореляційного аналізу.

Мовою ймовірностей правило асоціації  $x \rightarrow y$  – це вираження умовної ймовірності  $P(y | x)$ .

Правило асоціації характеризується двома параметрами: *опорою* (*support*) та *довірою* (*confidence*).

Нехай  $I = \{i_1, i_2, \dots, i_k\}$  – множина елементів,  $T = \{t_1, t_2, \dots, t_n\}$  – множина транзакцій, причому кожна транзакція – це підмножина  $I$ . Нехай  $C$  – підмножина  $I$ . Тоді *опора*  $C$  у відношенні  $T$  – це кількість транзакцій, які містять  $C$ :

$$\sigma(C) = \{t \mid t \in T, C \subset t\},$$

( $\sigma(C)$  – означає потужність множини  $C$ , а не операцію реляційної алгебри).

У цьому випадку  $i_1 \rightarrow i_2$  тоді і лише тоді, коли виконуються дві умови:

1) *опора*:  $i_1$  та  $i_2$  відбуваються, принаймні, в  $S\%$  транзакцій:

$$\frac{\sigma(i_1 \wedge i_2)}{|T|}.$$

2) *довіра*: із всіх транзакцій, в яких відбувається  $i_1$ , принаймні  $C\%$  містять  $i_2$ :

$$\frac{\sigma(i_1 \wedge i_2)}{\sigma(i_1)}.$$

Для опису просторових правил асоціації використовують просторові предикати. Просторове правило асоціації – це правило виду

$$P_1 \wedge P_2 \wedge \dots \wedge P_n \rightarrow Q_1 \wedge Q_2 \wedge \dots \wedge Q_m,$$

де хоча б один із предикатів  $P_i$  чи  $Q_j$  є просторовим.

*Кластеризація* – це процес виявлення „груп” чи кластерів у великих базах даних. На відміну від класифікації, кластеризація не використовує жодної апріорної інформації про кількість груп чи характер міток; тобто, в кластеризації немає поняття тренувальних чи навчальних даних, тому кластеризацію часто називають „навчання без вчителя”.

Кластери формуються за критерієм „подібності”, який використовують для визначення зв'язків між кожною парою кортежів в базі даних. Кортежі, які виявляються подібними, об'єднуються в групу, і ця група позначається своєю міткою.

Задачу кластеризації зручно розглядати у межах багатовимірного простору атрибутів. За наявності  $n$  об'єктів даних, описаних  $m$  змінними, кожний об'єкт подається точкою в  $m$ -вимірному



просторі. Тоді кластеризація зводиться до *визначення груп високої густини на множині нерівномірно розподілених точок*.

Для просторових атрибутів задача кластеризації може трактуватися двоюко – залежно від того, що вважати кластером: область, яка домінує над оточенням, чи область однорідних значень.

Кожний із цих способів видобування даних потрібно модифікувати для використання при видобуванні просторових даних. Є дві стратегії такої модифікації: по-перше, можна відкоригувати базову статистичну модель, основу на припущенні про незалежності та ідентичності розподілу (метод просторової регресії); по-друге, можна внести зміни до цільової функції, яка керує процесом пошуку, ввівши в неї просторовий член (алгоритм максимізації очікування). Просторова регресія обговорюється в [71, 72]. Просторові шаблони сумісного розміщення описані в [73]. Кластеризації присвячені роботи [74, 75, 76].

У роботі [77] розглянуто питання видобування відеоінформації із великих наборів просторових даних. Робота [78] присвячена проблемам штучного інтелекту – навчанню на прикладах для збільшення ефективності видобування просторових даних.

### 7. Просторові сховища даних

Просторове сховище даних – це особливий вид бази даних, розроблений для ефективного опрацювання підсумкових запитів статистичної інформації [1]. Операції агрегування можна поділити на три групи: *розподілювальні, алгебраїчні та цілісні*.

Функція агрегування  $F$  називається *розподілювальною*, якщо існує інша функція  $G$ , така, що значення  $F$  в  $N$ -вимірному прямокутному паралелепіпеді може бути розраховане шляхом застосування функції  $G$  до значення  $F$  в  $(N+1)$ -вимірному паралелепіпеді (наприклад, функція  $Min$  – мінімальний елемент).

Функція агрегування  $F$  є *алгебраїчною*, якщо значення  $F$  для  $N$ -вимірному прямокутного паралелепіпеді можна розрахувати на основі фіксованої кількості значень функцій агрегування для  $(N+1)$ -вимірному прямокутного паралелепіпеді, (наприклад, функція  $Average$  – середнє значення).

Функція агрегування  $F$  називається *цілісною*, якщо значення  $F$  для  $N$ -вимірному прямокутного паралелепіпеді не можна розрахувати на основі відомої кількості значень функцій агрегування для  $(N+1)$ -вимірному прямокутного паралелепіпеді (приклад:  $Range$  - ранг).

Розподілювальні та алгебраїчні функції агрегування обчислюються за допомогою невеликої кількості буферів пам'яті при однократному перегляді наборів даних навіть тоді, коли набори даних занадто великі для того, щоб розмістити їх в оперативній пам'яті. Сховища даних часто працюють з однією таблицею фактів та набором таблиць вимірів, причому таблиця фактів зв'язана з усіма таблицями вимірів (так реалізоване подання *гіперкуба* даних). Підсумкові результати можна описати як підмножини вимірів. За наявності  $N$  вимірів існує  $2^N$  підсумкових таблиць, кожна з яких можна отримати за допомогою відповідного запиту на групування (запит із секцією GROUP BY). Оператор CUBE – це узагальнення SQL-секції GROUP BY для процесу отримання всіх  $2^N$  підсумкових таблиць за допомогою одного оператора. Для аналізу результатів оператора CUBE можуть використовуватися такі операції, як *зведені таблиці (pivot)*, *згортка (roll-up)*, *розгортка (drill-down)*, *розщеплення та розрізання (slice and dice)*.

*Згортка* – це підвищення рівня абстракції. Ця операція узагальнює один чи кілька вимірів і агрегує відповідні показники.

*Розгортка* – зниження рівня абстракції або підвищення деталізації. Дозволяє отримати детальну інформацію про один чи кілька вимірів і наводить результати агрегування більш низького рівня.

*Розщеплення та розрізання* – вибірка і проєкція. Розщеплення вздовж одного виміру нагадує розгортку за цим виміром на один рівень вниз, але кількість елементів, які відображаються, обмежене тією кількістю, яка задана в команді розщеплення. Розрізання – це розщеплення більш ніж на один вимір.

*Зведена таблиця* – це переорієнтування багатовимірного представлення даних. Представляє показники в різних варіантах розміщення зведеної інформації.

Просторові сховища даних описані в [79, 80].

## 8. Побудова прикладних систем для опрацювання просторової інформації

Публікації про реалізацію конкретних геоінформаційних систем можна поділити на кілька груп.

*По-перше*, – ті, на основі яких можна зрозуміти принципи реалізації ГІС, вони містять більш чи менш детальний опис моделі даних, на основі якої побудована ГІС.

*Другу* групу утворюють публікації, в яких зовсім не висвітлюється модель даних ГІС, зате описуються принципи її функціонування.

Нарешті, *третья* група – це опис моделі даних предметної області, який може вважатися передумовою для майбутньої розробки ГІС.

*Четверта* група – це публікації на тему використання геоінформаційних технологій для вирішення конкретних прикладних задач. Оскільки такі роботи не стосуються проблем побудови інформаційних моделей і розробки систем просторових баз даних, то розглядати їх не будемо.

До *першої* групи належить робота [81], де розглянуто два загальні підходи до побудови електронних карт. Перший оснований на пошаровому методі, другий – об'єктно-орієнтований. Багатошарова організація електронної карти за наявності гнучкого механізму управління шарами дозволяє об'єднати і відобразити значно більше інформації, ніж на звичайній карті.

Об'єктно-орієнтований підхід дозволяє створювати більш відкриті системи, які легше піддаються модифікації, оскільки конструкція таких систем базується на стійких об'єктах програми (не плутати з просторовими об'єктами). Для модифікації програми буває достатньо заміни об'єкта програми із ієрархічного ланцюга успадкування. Ті об'єкти програми, які знаходяться на вершині ієрархічного ланцюга, реалізують найпростіші системи; переміщуючись вниз, отримуємо більш складні об'єкти.

Далі розглядається застосування об'єктно-орієнтованого підходу при створенні „ГІС-95” (розробка НТЦ „Інтелтек”).

Просторові об'єкти групуються в такі класи: *CPoint* (точка), *CLine* (лінія), *CPolygon* (полігон), *CText* (текст), *CBitmap* (растрове зображення). Для реалізації спорідненості цих класів між собою їх предком робиться клас *CPrimitive* (примітив).

Крім відношень успадкування, існують інші типи відношень – наприклад, відношення використання і володіння. Лінія має точки початку та кінця, – *CLine* містить два об'єкта типу *CPoint* і ми кажемо, що *CLine* використовує *CPoint*. Лінія (в загальному випадку – ламана) характеризується і набором проміжних точок, кількість яких наперед невідома і може змінюватися (наприклад, при спрямленні *CLine*), – це примушує використовувати об'єкт „список”, вміщуючи у нього необхідні *CPoint*. Список володіє об'єктами *CPoint*, а *CLine* використовує список.

До кожного класу додається властивість – належати до якогось конкретного шару. Залежно від значення цього параметра визначають інформаційну суть об'єкта. *CLine* може зображувати і залізничну колію, і ріку – в одному випадку вона потрапить в один шар, в іншому – в другий.

Отже, семантичні властивості абстрагуються від об'єктів і переносяться в об'єкт *CLayer* (шар). Для того, щоб можна було при описі семантично різних характеристик використовувати один і той самий клас *CLayer*, до нього вводиться таблиця *CTable*, яка описує інформаційну суть об'єктів даного шару. Кількість стовпчиків визначається кількістю параметрів в об'єктах, які розміщуються в даному шарі, кожний стовпчик має свою назву і відповідний тип даних. Кількість рядків таблиці залежить від кількості об'єктів в шарі.

Отже, структура ГІС виглядає так: в базі даних зберігаються об'єкти *CMap*, кожний з яких реалізує карту чи її фрагмент. Об'єкт *CMap* – це впорядкований список *CLayer*'ів. Кожний *CLayer* містить список об'єктів, які відносяться до ієрархії класів, похідних від *CPrimitive*. Семантичні властивості цих об'єктів визначаються параметрами таблиці – об'єкта *CTable*, який використовується *CLayer*'ом.

Вибір об'єктного підходу обґрунтовується природністю подання моделі реального світу, керованістю процесом розробки проекту, простотою модернізації та необхідністю зберігати і опрацьовувати неоднорідну інформацію.

До *другої* групи належать роботи [82, 83 та 84]. Розглянемо роботу [82], де описано концепцію формування єдиного інформаційного середовища наукового центру “Мінерал”, суттєву частину якого становлять географічні інформаційні системи (ГІС), створені на АРМ наукових працівників за основними напрямками досліджень ІГЕМ РАН. Ці ГІС створюються на єдиних програмних продуктах, цифрових картографічних основах і за єдиними технологічними принципами, що дозволяє формувати єдине інформаційне середовище ІС.

Проектування функціонування ГІС здійснюється в три етапи. На першому етапі вхідна інформація у вигляді векторних і растрових цифрових карт перетворюється у просторові бази даних, які містять картографічну та атрибутивну частини. На другому етапі інформація із просторових баз даних опрацьовується для отримання необхідних даних для розв’язання прикладних задач – вибірки даних, управління шарами карт. Для управління процесами другого етапу та організації взаємодії з робочими місцями інших дослідників, отримання потрібних цифрових карт і виходу в INTERNET до АРМ під’єднаний картографічний сервер та сервер INTERNET.

На третьому етапі формуються інтегральні прикладні моделі, інформаційне наповнення яких здійснюється опрацьованими на другому етапі даними.

Робота з моделлю, внесення коректив здійснюється оператором. На цьому ж етапі результати моделювання подаються користувачу у вигляді цифрових карт, фактографічних даних, графіків, таблиць і т.п.

До *третьої* групи можна зарахувати роботи [85, 86]. Зокрема, в роботі [85] зазначається важливість використання концептуального моделювання та процедури нормалізації для проектування систем баз даних і описується модель “сутність – зв’язок” для побудови автоматизованих систем державного кадастрового обліку земельних ділянок.

Є кілька зауважень щодо змісту вказаної публікації. Мало того, що значна частина тексту відведена для реклами давно відомого підходу до розробки інформаційних систем та проектування баз даних; розроблена автором модель предметної області охоплює лише атрибутивну частину, картографічні дані залишаються зовсім поза увагою.

Отже, найбільшу цінність для дослідника з метою отримання інформації про існуючі розробки становлять публікації *першої* категорії.

## Висновки

Аналізуючи сучасні публікації, можна зробити висновки про такі перспективні напрямки досліджень у кожній із вищезазначених областей просторових баз даних.

### 1. Моделі даних

Перспективними є дослідження в напрямку як розширення класичних моделей (наприклад, „сутність–зв’язок”), так і застосування об’єктно-орієнтованого моделювання даних. Особливо багатообіцяючою є спроба поєднати зручність об’єктного підходу з автоматичним отриманням растрових даних (об’єкти з властивостями типу „покриття”).

Полеміка щодо переваг та недоліків кожного з двох основних підходів (польового та об’єктного) не вщухає, достатньо пригадати Третій маніфест [20] і полеміку з його приводу [87, 88]. Навряд чи якась одна сторона зуміє переконати іншу в тому, що саме її підхід єдино правильний. Тому праві практики, які, поки тривають суперечки теоретиків, вдосконалюють та розширюють існуючі моделі даних.

Слід також звернути увагу на потребу у 3-вимірних моделях просторових даних, а не лише у 2-вимірних географічних моделях. По-перше, незабаром людство почне в промислових масштабах використовувати навколоземний космічний простір, як зараз використовується поверхня Землі. По-друге, і на Землі достатньо багато задач, які потребують автоматизації опрацювання 3-вимірних просторових даних: проектування будинків, механізмів, корпоративних та будинкових мереж тощо. Цей перелік можна ще довго продовжувати.

### 2. Мови просторових запитів

Введення операцій над просторовими даними до існуючих СУБД – ще одна важлива галузь, яка має велике практичне значення.

Стандарт SQL3 за рахунок об'єктно-реляційного розширення вже дає можливість опрацювати просторові дані, тим самим стандарт SQL ще більше відходить від класичної реляційної моделі. Очевидно, що таку „гібридність” – поєднання реляційних основ та нових об'єктних можливостей – можна буде подолати лише за допомогою якісно іншої моделі даних.

### *3. Зберігання та індексування просторових даних*

Завдання структури зберігання просторових даних полягає в спрощенні запитів просторової вибірки та з'єднання: метод просторового доступу повинен переглядати лише потрібну підмножину об'єктів із простору пошуку. Тому перспективними є дослідження для виявлення нових ефективних алгоритмів декомпозиції просторових об'єктів, управління паралельними обчисленнями та ефективною побудовою просторових індексів.

### *4. Опрацювання та оптимізація запитів*

Традиційно опрацювання і оптимізація запитів поділяється на два етапи: 1) проектування і тонке налаштування алгоритмів виконання кожного базового реляційного оператора; 2) перетворення запитів високого рівня в набір зазначених базових реляційних операторів та оптимізація з використанням інформації, отриманої на першому етапі.

Сфера застосування просторових баз даних різноманітна, і ще не досягнуто консенсусу стосовно множини просторових операцій, які покривали би всі можливі випадки. Узгоджено лише класи просторових операцій. Просторові операції накладають високі вимоги як до ресурсів ЦП, так і до ресурсів системи введення-виведення.

Все більший вплив отримують Web-технології. Прогнозується можливість для кожного комп'ютера бути підключеним до Internet і активно використовувати всі світові інформаційні ресурси – це виражено словами „користувач не повинен здогадуватися, що в нього є вінчестер” [89]. Тому перспективними є дослідження ефективності просторових баз даних на базі Web та реалізація паралельних систем просторових баз даних.

### *5. Просторові мережі*

Просторово-мережні бази даних (spatial network databases) – важлива складова частина просторових БД. Вони утворюють ядро багатьох прикладних систем, таких як планування перевезень, управління транспортними потоками міста, водопостачальні, електричні і газові комунальні служби, телефонні мережі, проектування міського будівництва, річковий та залізничний транспорт; управління зрошувальними системами тощо.

Просторова близькість – традиційне базове відношення для просторових об'єктів, яке у випадку просторових мереж переходить у відношення на основі зв'язності. Зміщення акценту від близькості до зв'язності приводить до особливої інтерпретації просторово-мережних баз даних. Тому в цій галузі перспективні напрями досліджень методів доступу до просторових мереж, реалізація операцій над графами в середовищі просторових баз даних.

### *6. Просторове видобування даних*

Видобування даних (data mining) – систематична процедура пошуку потенційно корисної інформації, схованої в цифрових даних, – все ще „гаряча ” тема наукових досліджень. Багато організацій (урядових, комерційних та промислових) усвідомили, що великі об'єми даних, накопичені за тривалий час, можуть стати об'єктом систематичних досліджень з метою пошуку нових перспектив власної діяльності. Тому дані, які раніше вважалися зайвим тягарем з точки зору зберігання та управління, раптом перетворилися на джерело збагачення.

Значна частина досліджень у галузі видобування даних орієнтована на створення і аналіз алгоритмів, отже, перспективи полягають у побудові ефективних алгоритмів виявлення як просторових шаблонів, так і визначення просторових викидів (якісних відмінностей від середньостатистичних просторових показників).

### *7. Просторові сховища даних*

Сховище даних (data warehouse) – це сукупність технологій підтримки прийняття рішень, які дозволяють фахівцям, що займаються аналізом та опрацюванням інформації (адміністраторам, керівникам, аналітикам), швидше приймати більш ефективні рішення. Сховища даних містять великі об'єми інформації, зібраної з багатьох незалежних джерел, і часто підтримуються незалежно від робочої бази даних.

Робочі бази даних оптимізуються для оперативного опрацювання транзакцій (OLTP), коли вирішальне значення мають цілісність і відновлюваність інформації. Транзакції, як правило, невеликі і звертаються до малої кількості записів. Тобто робочі бази даних містять інформацію про поточний стан справ. На відміну від робочих баз даних, сховища даних містять історичну інформацію і спроектовані для аналітичного опрацювання (OLAP), під час якого запити агрегують великі об'єми даних для виявлення тенденцій та аномалій розвитку.

Просторові сховища містять дані географічного характеру (наприклад, супутникові зображення). Основна відмінність просторових сховищ даних від звичайних полягає у візуалізації результатів – результатом роботи просторового сховища часто повинна бути не лише таблична чи графічна інформація, а і просторові карти.

Проблема полягає в тому, що не існує стандартів для операцій агрегування просторових типів даних. Тому в цій галузі перспективними є дослідження як ефективного опрацювання надвеликої кількості багатовимірних даних, отриманих з просторової інформації, так і побудова моделей операцій просторового агрегування.

Зважаючи на те, що протягом найближчого майбутнього прогнозується [89] черговий прорив в апаратних та програмних технологіях (з'являться нові обчислювальні комплекси, які будуть реалізовані на принципово інших технологіях зберігання та опрацювання інформації, та нові операційні системи для них), очевидним буде перегляд багатьох „догм” в організації баз даних. Зокрема, якщо весь масив даних можна буде розмістити в оперативній пам'яті (обсяг якої досягне порядку терабайт), то про жодне використання B<sup>+</sup>-дерев для організації доступу до даних вже не може бути і мови, – адже цей спосіб ефективний лише при розміщенні даних у вторинній пам'яті.

Різке збільшення апаратних можливостей призведе до кардинального перегляду всіх теоретичних основ баз даних. Адже переважна більшість теоретичних розробок була викликана саме необхідністю суворої економії оперативної пам'яті та (можливо, і не так суворо) – дискової пам'яті, тому треба чекати теоретичного обґрунтування якісно нового уявлення про організацію баз даних.

1. Шаши Шехар, Санжей Чаула. *Основы пространственных баз данных / Пер. с англ.* – М.: КУДИЦ-ОБРАЗ, 2004. – 336 с. 2. Worboys, M. *GIS: A Computing perspective.* Taylor and Francis. 1995. 3. Couclelis, H. *People manipulate objects (but cultivate fields): beyond on the raster-vector debate in gis // Theories and Methods of Spatio-temporal reasoning in Geographic Space.* 1992. – P. 65–77. 4. Tomlin, C. *Geographic information systems and cartographic modeling.* Prentice-Hall. 1990. 5. Ritter, G., Wilson, J., Davidson, J. *Image algebra: An overview // Computer vision, Graphic and Image Processing.* 1990. – P. 297–331. 6. Takeyama, M., Couclelis H. *Map dynamics: integrating cellular automata and gis through geo-algebra // International Journal of GIS,* 1997. 11(1). – P. 73–91. 7. Egenhofer, M. *Extending SQL for geographical display // Cartography and Geographic Information systems.* 1991, 18(4). – P. 230–245. 8. Clementini, E., Felice, P.D., van Oosterom, P. *A small set of formal topological relationship suitable for end-user interaction // Lecture Notes in Computer Science.* Springer-Verlag. 1993. Vol. 692. – P. 277–295. 9. Shekhar, S., Chawla, S., Ravada, S., Fetter, A., Liu, X., Lu, C. *Spatial databases – accomplishments and research needs // IEEE TKDE.* 1999. 11(1). – P. 45–55. 10. *Open GIS Consortium: Open GIS simple features specification for SQL (Revision 1.1), 1999 // <http://www.opengis.org/technology/specs/00-028.pdf>*. 11. Egenhofer, M, Frank, A. U., and Jackson, J. P. *A topological data model for spatial databases // SSD '89, Design and Implementation of Large Spatial Databases, First Symposium – P. 47–66.* Springer-Verlag. 1989. 12. Geerts, F., Kuipers, B. *Expressing Topological Connectivity of Spatial Databases // Lecture Notes in Computer Science,* 2000. Volume 1949, p. 224. 13. Erik Hoel, Sudhakar Menon and Scott Morehouse. *Building a Robust Relational Implementation of Topology // Lecture Notes in Computer Science. Volume 2750 / 2003, pp. 508 – 509.* 14. Jinfeng Ni, Chinya V. Ravishankar and Bir Bhanu. *Probabilistic Spatial Database Operations // Lecture Notes in Computer Science. Volume 2750 / 2003. – P. 140–158.* 15. Бобов П. *Пространственные типы данных // Электронный бюллетень «ГИС-релиз».* <http://cgis.kem.uiggm.nsc.ru/bulletin> – февраль, 2000 г. 16. Hadzilacos, T. and Tryfona, N.

*An extended entity-relationship model for geographic application // SIGMOD Record, 1997, 26(3).*

17. Shekhar, S., Coyle, M., Liu, D.-R., Goyal, B., Sarkar, S. *Data models in geographic information systems // Communication of the ACM, 1997, 40(4).*

18. Shekhar, S., Vatsavai, R. R., Chawla, S., Burk, T. E. *Spatial pictogram enhanced conceptual data models and their translation to logical data models // Integrated Spatial Databases: Digital Images and GIS. Lecture Notes in Computer Science. 1999: 1737. – P. 77–104.*

19. Brodeur, J., Bedard, Y., Proulx, M.-J. *Modelling geospatial application databases using uml-based repositories aligned with international standards in geomatics // ACM-GIS 2000. Proceedings of the Eighth ACM Symposium on Advances in Geographic Information Systems, November 10–11, 2000. ACM, p. 39-46.*

20. Дейт К.Дж., Дарвен Х. *Основы будущих систем баз данных. Третий манифест. М. „Янус-К”, 2004. – 656 с.*

21. Королев Ю.К., Тихонова Н.М. *Различные модели пространственных данных. / Материалы второго семинара «Проблемы ввода и обновления пространственной информации». (Москва, 24–27 февраля 1997 г.) <http://www.gisa.ru/2046.html>*

22. Sheng Zhou and Christopher B. Jones. *A Multi-representation Spatial Data Model // Lecture Notes in Computer Science. Volume 2750 / 2003. – P. 394–411.*

23. Changhao Jiang and Peter Steenkiste. *A Hybrid Location Model with a Computable Location Identifier for Ubiquitous Computing // Lecture Notes in Computer Science. Volume 2498 / 2002, p. 246.*

24. Egenhofer, M. *Spatial SQL: A Query and Presentation Language // IEEE TDKE, 1994. 6(1): p. 86-95.*

25. *Open GIS simple features specification for SQL. Open GIS Consortium, Inc., 1998 <http://www.opengis.org>*

26. Lin, H., Huang, B. *Sql/sda: A query language for supporting spatial data analysis and its web-based implementation // IEEE Transactions on Knowledge and Engineering, 2001. 13(4). – P. 671–682.*

27. Ang, C., Ling, T., Zhou, X. *Qualitative spatial relationships representation and its retrieval // Database and Expert systems Applications (DEXA). Springer-Verlag, 1998. V. 1460. – P. 65–77.*

28. Petrakis, E., Faloutsos, C. *Similarity searching in medical image databases // IEEE TDKE, 1997. 9(3). – P. 433–447.*

29. Yoshitaka, A., Ichikawa, T. *A Survey on Content-Based Retrieval for Multimedia Databases // IEEE TKDE, 1999. 11(1). – P. 81–93.*

30. Papadias, D., Karacapilidis, N., Arkoumanis, D. *Processing fuzzy spatial queries: A configuration similarity // International Journal of GIS, 1999. 13(2). – P. 93–128.*

31. Egenhofer, M. *Query Processing in Spatial-Query-by-Sketch // Journal of Visual languages and Computing, 1997. 8(4). – P. 403–424.*

32. Ma, W., Manjunath, B. *A texture thesaurus for browsing large aerial photographs // Journal of the American Society for Information Science, 1998. 49(7). – P. 633–648.*

33. Блинкова О. *Oracle 8i Spatial: СУБД для работы с пространственными данными // [http://www.pcweek.ru/year\\_1999/n35/cp1251/News/chat9.htm](http://www.pcweek.ru/year_1999/n35/cp1251/News/chat9.htm)*

34. Brinkhoff, T., Kriegel, H., Schneider, R., Seeger, B. *Multi-step processing of spatial joins // Proceeding of ACM SIGMOD Int. Conf. on Management of Data. ACM Press, 1994. – P. 197–208.*

35. Leutenegger, S. T., Lopez, M. A. *The effect of buffering on the performance of r-trees // IEEE Transactions on Knowledge and data engineering, 2000. 12(1). – P. 33–44.*

36. Faloutsos, C., Roserman, S. *Fractals for secondary key retrieval // Proceeding of the ACM conference on Principles of Database Systems, 1989. – P. 247–252.*

37. Yi-hong Dong. *Hierarchical Clustering Algorithm Based on Neighborhood-Linked in Large Spatial Databases // Lecture Notes in Computer Science. Volume 2639 / 2003. – P. 619–622.*

38. Samet, H. *The design and analysis of spatial data structures. Addison-Wesley, 1990.*

39. Gaede, V., Gunther, O. *Multidimensional access methods // ACM Computing Surveys, 1998. 30(2). – P. 170–231.*

40. Guttman, A. *R-tree: A dynamic index structure for spatial searching // SIGMOD'84, Proceeding of the ACM SIGMOD Conference. ACM Press, 1984.*

41. Kornacker, M. and Banks, D. *High-concurrency locking in R-trees // VLDB '95, Proceedings of 21th International Conference on Very Large Data Bases. Morgan Kaufmann. 1995.*

42. Kornacker, M., Banks, D. *High-concurrency locking in R-trees // VLDB'95. Proceedings of 21th International Conference on Very large Data Bases. Morgan Kaufmann, 1995.*

43. Chakrabarti, K., Mehrotra, S. *Efficient concurrency control in multidimensional access methods // SIGMOD 1999, Proceedings ACM SIGMOD International Conference on Management of Data. ACM Press, 1999. – P. 25–36.*

44. Shekhar S.Lu., C. Chawla, S. Ravada S. *Efficient join index based join processing; a clustering approach // TR99-030, 1999.*

45. Essam A. El-Kwae. *Signature-Based Indexing for Retrieval by Spatial Content in Large 2D-String Image Databases // Lecture Notes in Computer Science. Volume 1932 / 2000,*

p. 97. 46. Brinkhoff, T. and Kriegel, H.-P. The impact global clustering on spatial database systems // *Proceedings of the 20th International Conference on Very Large Data Bases, (VLDB '94)*. Morgan Kaufmann. 1994. 47. Gutting, R. An Introduction to Spatial Database Systems // *The VLDB Journal*. 1994, (4). – P. 357–399. 48. Brinkhoff, T., Kriegel, H., and Seeger, B. Efficient processing of spatial joins using r-trees // *Proceedings of ACM SIGMOD Int. Conf. on Management of Data*. 1993. – P. 237–246. ACM Press. 49. Patel, J. M. and DeWitt, D. J. Partition Based Spatial-Merge Join // *Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data*. ACM Press. 1996. 50. Lo, M. and Ravishankar, C. Spatial hash joins // *ACM SIGMOD International Conference on Management of Data*. 1996. – P. 247–258. ACM Press. 51. Mamoulis, N. and Papadias, D. Integration of spatial join algorithms for processing multiple inputs // *SIGMOD 1999, Proceedings ACM SIGMOD International Conference on Management of Data, June 1-3, 1999, Philadelphia, Pennsylvania, USA, pages 1-12*. ACM Press. 52. Patel J.M. and DeWitt D. J. Clone join and shadow join: two parallel spatial join algorithms // *ACM-GIS 2000, Proceedings of the Eighth ACM Symposium on Advances in Geographic Information Systems, November 10–11, 2000, Washington, D.C., USA*. – P. 54–61. ACM. 53. Song J.-W., Whang, K.-Y. Lee, Y.-K. Lee, M.-J. and Kim S.-W. Transformation-based spatial join // *Proceedings of the 1999 ACM CIKM International Conference on Information and Knowledge Management, Kansas City, Missouri, USA, November 2–6, 1999*. – P. 15–26. ACM. 54. Arge, L., Procopiuc, O., Ramaswamy, S., Suel, T., and Vitter, J. S. Scalable sweeping-based spatial join // *VLDB '98, Proceedings of 24rd International Conference on Very Large Data Bases, August 24–27, 1998, New York City, New York, USA*. – P. 570–581. Morgan Kaufmann. 55. Shin H., Moon B., and Lee S. Adaptive multi-stage distance join processing // *SIGMOD Record*. 2000, 29(2). – P. 343–354. 56. Adam N. and Gangopadhyay A. Database issues in Geographical Information Systems. Kluwer Academics. 1997. 57. Hellerstein J. and Stonebraker M. Predicate migration: Optimizing queries with expensive predicates // *SIGMOD '93, Proceedings of the ACM SIGMOD International Conference on Management of Data*. 1993. – P. 267–276. ACM Press. 58. Chaudhuri S. and Shim K. Optimization of queries with user-defined predicates // *VLDB '96, Proceedings of 22nd International Conference on Very Large Data Bases*. 1996. – P. 87–98. Morgan Kaufmann. 59. Theodoridis Y., Stefanakis E., and Sellis T. K. Efficient cost models for spatial queries using r-trees // *IEEE Transactions on Knowledge and Data Engineering*. 2000, 12(1). – P. 19–32. 60. Shekhar S., Coyle M., Liu D.-R., Goyal B. and Sarkar S. Data models in geographic information systems // *Communication of the ACM*. 1997, 40(4). 61. Gutting R. Graph DB: Modeling and querying graphs in databases // *VLDB '94, Proceedings of 20th International Conference on Very Large Data Bases*. Morgan Kaufmann. 1994. 62. Geerts F. Linear Approximation of Semi-algebraic Spatial Databases Using Transitive Closure Logic, in Arbitrary Dimension // *Lecture Notes in Computer Science*, 2002. Volume 2397. – P. 182–197. 63. ESRI. Network analysis: Modeling network systems. Environmental Systems Research Institute, Inc. 1991. 64. Mannino M. and Shapiro L. Extensions to query languages for graph traversal problem // *IEEE TKDE*. 1990, 2(3). – P. 353–363. 65. Biskup J., Rasch U., and Stiefeling H. (). An extension of sql for querying graph relations // *Computer Languages*. 1990, 15(2). – P. 65–82. 66. Jing N., Huang Y. and Rundensteiner E. Hierarchical encoded path views for path query processing: An optimal model and its performance evaluation // *IEEE Transactions on Knowledge and Data Engineering*. 1998, 10(3). – P. 409–432. 67. Jiang B. Traversing graphs in a paging environment, bfs or dfs? // *Information Processing Letters*. 1991, 37(3). – P. 143–147. 68. Shekhar S. and Liu D.-R. A connectivity-clustered access method for networks and network computation // *IEEE TKDE*. 1997, 9(1). – P. 102–117. 69. Han J., Koperski K. and Stefanovic N. Geominer: A system prototype for spatial data mining // *SIGMOD 1997, Proceedings ACM SIGMOD International Conference on Management of Data, May 13–15, 1997, Tucson, Arizona, USA*. – P. 553–556. ACM Press. 1997. 70. Cressie N. A. Statistics for spatial data. Wiley Series in Probability and Statistics. 1993. 71. Anselin L. Spatial econometrics: Methods and models. Kluwer. 1988. 72. LeSage J. Regression analysis of spatial data // *Journal of Regional Analysis and Policy (Publisher: Mid-Continent Regional Science Association and UNL College of Business Administration)*. 1997, 27(2). – P. 83–94. 73. Shekhar S. and Huang Y. Discovering spatial co-location patterns: A summary of results // *Proceedings of*

the 7th International Symposium on Spatial and Temporal Databases, SSTD 2001, Redondo Beach, CA, USA, July 12–15, 2001, volume 2121 of Lecture Notes in Computer Science. – P. 236–256. Springer. 74. Ordonez C. and Cereghini P. Sqlem: Fast clustering in sql using the em algorithm // SIGMOD Record. 2000, 29(2). – P. 559–570. 75. Sheikholeslami, G., Chatterjee, S., and Zhang, A. Wavecluster: A multi-resolution clustering approach for very large spatial databases // VLDB '98, Proceedings of 24rd International Conference on Very Large Data Bases, August 24-27, 1998, New York City, New York, USA. – P. 428–439. Morgan Kaufmann. 1998. 76. Wang W., Yang J. and Muntz R. R. Sting: A statistical information grid approach to spatial data mining // VLDB '97, Proceedings of 23rd International Conference on Very Large Data Bases, August 25–29, 1997, Athens, Greece. – P. 186–195. Morgan Kaufmann. 1997. 77. Daniel A. Keim, Christian Panse and Mike Sips. Visual Data Mining of Large Spatial Data Sets // Lecture Notes in Computer Science. Volume 2822 / 2003 – P. 201–215. 78. Lubos Popelinský. Efficient Relational Learning from Sparse Data // Lecture Notes in Computer Science. Volume 2443 / 2002. – p. 11. 79. Barclay T., Slutz D. R., and Gray J. Terraserver: A spatial data warehouse // SIGMOD Record. 2000, 29(2). P. 307–318. 80. Shekhar S., Schrater P. R., Vatsavai R. R., Wu, W., and Chawla, S. Spatial contextual classification and prediction models for mining geospatial data // IEEE Transactions on Multimedia. 2002, 4(2). – P. 1–15. 81. Андреев А.М., Березкин Д.В., Куликов Ю.В., Смагин А.Ю., Смелов А.В. Геоинформационные системы, объектно-ориентированный подход к проектированию // Объектные ГИС, «Геодезия и картография», 1995 г, №9. <http://www.intelect.ru/publish/articles/objtech/geodesy.shtml> 82. Веселовский А.В. Географические информационные системы научного центра «Минерал»// ГИС – технологии и проблемы геоинформатики. Вестник ОГГТГН, N1(7) ' 99. 1999. 83. Койнов Е., Патренина М. Вопросы применения технологий ГИС/Интернет на примере справочно-информационной системы «Весь Академгородок» // [http://www.giscenter.ru/bulletin/content/all\\_ac.htm](http://www.giscenter.ru/bulletin/content/all_ac.htm) Электронный бюллетень «ГИС-релиз» - июль, 2000. 84. Коритнюк Т., Лагоднюк О. Проект автоматизованої системи реєстрації землекористувачів, землевласників, земельних ділянок і прав на них – „Реєстр 1.0” // <http://www.geomatica.kiev.ua/published/ua/fmaterials/lviv1/article28.shtml> 85. Константинов А.Ю. Моделирование данных в земельных кадастрах. // Геодезия и картография. №12, 2003. 86. Кузнецов А. Пространственно-логические отношения между элементами кадастрового учета на примере внедрения денежной оценки земель города Днепрпетровска. // <http://www.geomatica.kiev.ua/published/ua/fmaterials/lviv1/article35.shtml> 87. Кузнецов С.Д. Третий манифест К.Дейта и Х.Дарвена: предпосылки и обзор // „Открытые системы”, #04, 2000 г. <http://www.osp.ru/os/2000/04/061.htm> 88. Кузнецов С.Д. Третий манифест К.Дейта и Х.Дарвена: немного формализма // „Открытые системы”, #07-08, 2000 год. <http://www.osp.ru/os/2000/07-08/058.htm> 89. Асиломарский отчет о направлениях исследований в области баз данных // „Открытые системы”, #01, 1999 год. <http://www.osp.ru/os/1999/01/61.htm>.