STUDIA INFORMATICA 2000
Volume 21 Number 3 (41)

Piotr BAJERSKI
Politechnika Śląska, Instytut Informatyki

WYBRANE ZAGADNIENIA OPTYMALIZACJI ZAPYTAŃ PRZESTRZENNYCH WYKORZYSTUJĄCYCH APROKSYMACJE OBIEKTÓW

Streszczenie. W artykule przedstawiono model aproksymacji rozkładu przestrzennego za pomocą linowego drzewa czwórkowego. Scharakteryzowano zadania wyszukiwania dotyczące relacji przestrzennych między różnymi rozkładami przestrzennymi i między rozkładami a innymi obiektami przestrzennymi. Omówiono operacje konieczne do zapisu i wykonania zapytań. Przedstawiono reguły optymalizacji zapytań, koncentrując się na wyznaczaniu części wspólnej obiektów przestrzennych i połączeniu generacji rozkładów z innymi operacjami.

ON OPTIMIZATION OF SPATIAL QUERIES BASED ON OBJECTS APPROXIMATION

Summary. The paper presents a model of spatial distribution approximated by linear quadtree. Queries concerning spatial relations between distributions as well as distributions and other spatial objects are characterised. Operations used for query formulation and execution are discussed. Query optimisation rules are given, with special attention devoted to spatial objects intersection and combining distribution computation with other operations.

1. Wstęp

Artykuł ma na celu charakterystykę zadań wyszukiwania specyficznych dla rozkładów przestrzennych, opisanie operacji koniecznych do ich wykonania oraz przedstawienie wybranych reguł ich optymalizacji. Wykorzystywane struktury danych i algorytmy są dostosowane do specyfiki rozkładów – występowania stosunkowo małej liczby obiektów powierzchniowych o skomplikowanym przebiegu granic i dużej liczbie dziur. Cechą charakterystyczną analizowanej klasy zapytań jest możliwość wystąpienia złożonych obliczeń, w przypadku gdy

zadane rozkłady przestrzenne nie zostały wcześniej wyznaczone i zapamiętane w bazie danych.

Dane opisujące położenie obiektów w przestrzeni oraz ich kształt (geometrię) będą w dalszej części nazywane *danymi przestrzennymi*. Atrybuty obiektu można podzielić na *atrybuty przestrzenne*, których wartościami są dane przestrzenne, i *atrybuty nieprzestrzenne*, nazywane również *atrybutami opisowymi*. Obiekt posiadający atrybuty przestrzenne będzie nazywany *obiektem przestrzennym*. Zbiór obiektów przestrzennych posiadających ten sam zestaw atrybutów definiuje *relację przestrzenną*. Przykładowo obiekt *Gmina* może mieć atrybut przestrzenny *Granica* oraz atrybuty opisowe *Nazwa* i *Powierzchnia*. Zbiór obiektów modelujących podział administracyjny: województwo, powiat, gmina może stanowić relację przestrzenną *ObiektyAministracyjne* (zakładając, że mają takie same atrybuty). Zapytania dotyczące atrybutów przestrzennych nazywane są *zapytaniami przestrzennymi*.

Wyróżnia się następujące reprezentacje danych przestrzennych, [4]:

- wektorową (ang. *vector representation*), w której podstawowym elementem jest punkt, a pozostałe typy obiektów (linie i wielokąty) są opisywane ciągami punktów,
- płatową (ang. *tessellated representation*) – przestrzeń jest dzielona na dyskretne fragmenty (płaty) w sposób regularny lub nieregularny. Obszar zajęty przez obiekty jest opisywany przez zajmowane płaty. Szczególnym przypadkiem jest raster, gdzie wszystkie obiekty są opisane za pomocą jednakowych elementarnych fragmentów (pikseli). Przykładem regularnego rekurencyjnego podziału jest drzewo czwórkowe,
- hybrydową (ang. *hybrid representation*) – stanowi połączenie reprezentacji płatowej i wektorowej,
- analityczną (ang. *analytical representation*) – geometria obiektów jest opisana funkcjami.

W systemach informacji przestrzennej najszersze zastosowanie znalazły reprezentacje wektorowa i płatowa. Zastosowanie reprezentacji analitycznej jest ograniczone z powodu złożoności i braku regularności obiektów przestrzennych.

Termin aproksymacja w odniesieniu do obiektów przestrzennych oznacza przybliżenie ich rzeczywistej geometrii prostszymi figurami geometrycznymi. Zabieg ten umożliwia zmniejszenie ilości danych, które muszą być przechowywane i przetwarzane oraz użycie szybszych algorytmów przy zachowaniu podstawowych charakterystyk obiektów. Dla uproszczenia terminologii przyjęto, że jeżeli nie będzie to prowadziło do nieporozumień, to aproksymacja obiektu będzie w skrócie nazywana obiektem.

Rozważania teoretyczne zostaną zilustrowane badaniem relacji przestrzennych między granicami administracyjnymi a przedziałami wartości dwuwymiarowego rozkładu zanieczyszczenia powietrza. Założono, że dane w reprezentacji płatowej są przechowywane w relacyjnej bazie danych, a dane wektorowe są pobierane z systemu informacji przestrzennej.

Termin *rozkład przestrzenny* (ang. *spatial distribution*) w systemach informacji przestrzennej oznacza rozkład lub zbiór geograficznych obserwacji reprezentujących wartości lub zachowanie danego zjawiska lub charakterystyki w wielu miejscach na powierzchni Ziemi, [7]. Przykładem rozkładu przestrzennego jest średnioroczne stężenie pyłu zawieszonego na terenie województwa śląskiego w roku 1999. Ważne jest rozróżnienie dwóch pojęć: istniejącego rozkładu i jego komputerowej reprezentacji. Pełny opis istniejącego rozkładu (nazywanego również *rozkładem rzeczywistym*) nie jest znany. Dostępne są tylko wartości w wybranych punktach (nazywanych węzłami siatki pomiarowej). W najprostszym przypadku rozkład jest rekonstruowany przy użyciu wzorów interpolacji liniowej, [5, 11, 13]. Podejście to abstrahuje od specyfiki modelowanych procesów fizycznych, wykorzystując jedynie wartości zmierzone w węzłach. Bardziej zaawansowane metody bazują na modelach matematycznych opisujących modelowane procesy fizyczne i wykorzystują inne parametry poza zmierzonymi wartościami rozkładu w węzłach. Dla pomiaru zanieczyszczenia powietrza parametrami takimi może być: prędkość i kierunek wiatru, ciśnienie atmosferyczne czy ukształtowanie terenu. Komputerowa reprezentacja rozkładu w postaci wzorów matematycznych będzie nazywana *modelem rozkładu*. Inne komputerowe reprezentacje rozkładu, w których pamiętane jest jego przybliżenie, będą nazywane *aproksymacjami rozkładu*. Przykładem aproksymacji rozkładu jest tablica wartości obliczonych w punktach dyskretnego obszaru (reprezentacja rastrowa), przebieg izolinii (reprezentacja wektorowa) czy lista określająca przynależności fragmentów obszaru do zadanych przedziałów wartości (reprezentacja płatowa). Jeżeli nie będzie to prowadziło do nieporozumień, aproksymacja rozkładu będzie nazywana w skrócie rozkładem.

W stosunku do analizowanych rozkładów i zapytań poczyniono następujące założenia:

- Założenie *Z1*: rozkłady są dwuwymiarowe.
- Założenie *Z2*: rzeczywisty rozkład można z zadaną dokładnością odtworzyć metodami interpolacji liniowej na podstawie numerycznych wartości w węzłach sieci pomiarowej.
- Założenie *Z3*: z punktu widzenia wykonywanych analiz istotne są przedziały wartości badanej cechy.
- Założenie *Z4*: obliczenia wykonywane są w przestrzeni dyskretnej. Wielkość elementarnych fragmentów tej przestrzeni jest zadawana przez użytkownika w zapytaniu.
- Założenie *Z5*: wartości w węzłach sieci pomiarowej są dokładne – weryfikacja i ewentualna korekta zmierzonych wartości powinna zostać wykonana przy użyciu innych narzędzi przed przetwarzaniem rozkładów.

Założenie *Z2* wymaga, aby rzeczywisty rozkład był ciągłą funkcją o wahaniu ograniczonym między węzłami sieci pomiarowej. Jeżeli założenie to nie jest spełnione, należy obli-

czyć, na podstawie modelu rozkładu, wartości w dodatkowych punktach i traktować te punkty w czasie przetwarzania aproksymacji rozkładu jak węzły sieci pomiarowej.

Zgodnie z założeniem *Z3* użytkownika nie interesują dokładne wartości, lecz ich przyporządkowanie do zadanych przedziałów. Przykładowo, nie jest istotne, że w danym punkcie (ponieważ działamy w przestrzeni dyskretnej, punktowi odpowiada elementarny fragment przestrzeni) średnioroczny opad ołowiu wynosi 232.8 $mg*m^{-2}*r^{-1}$; ważne jest natomiast, że wartość w tym punkcie należy do przedziału nr 3, obejmującego przedział wartości od jednokrotnego do dwukrotnego przekroczenia normy. Uwzględnienie założeń *Z3* i *Z4* prowadzi do rozbicia rozpatrywanego fragmentu przestrzeni na rozłączne części, na obszarze których wartości należą do tego samego przedziału.

Z punktu widzenia zapisu, optymalizacji i wykonania zapytań obiekty przestrzenne można podzielić na dwie grupy:

1. rozkłady przestrzenne,
2. inne obiekty przestrzenne – obiekty istniejące lub sztucznie wytyczone w rozważanej przestrzeni (powierzchniowe, liniowe i punktowe).

W trakcie wykonania zapytania rozkłady przestrzenne są tworzone w reprezentacji płatowej według zadanych przez użytkownika parametrów. Wykorzystywane są przy tym specjalne reguły optymalizacji. Obiekty grupy 2 są pobierane w postaci wektorowej i przekształcane do reprezentacji płatowej w celu wykonania operacji dotyczących rozkładów.

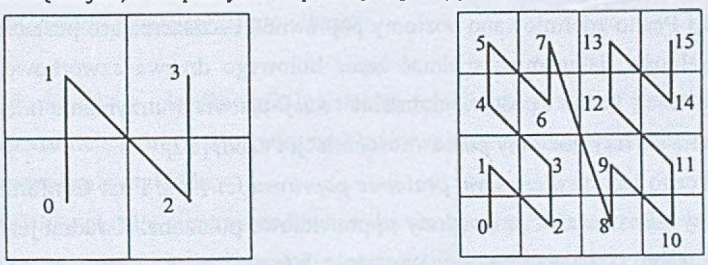
2. Krzywa i relacja N-Peano

W rozdziale zostaną omówione podstawowe pojęcia wykorzystywane przy tworzeniu i zapamiętywaniu rozkładu przestrzennego w relacyjnej bazie danych. Opis został ograniczony do przestrzeni dwuwymiarowej, chociaż przedstawione koncepcje można uogólnić na przestrzeń o większej liczbie wymiarów. Rozpatrywana przestrzeń jest podzielona na elementarne kwadraty o jednostkowej długości boku. Liczba kwadratów elementarnych, wzdłuż każdej z osi współrzędnych, określa *rozdzielczość* aproksymacji rozkładu.

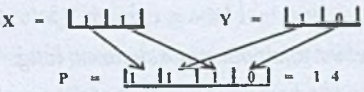
Jedną z metod identyfikacji pojedynczych kwadratów oraz ich grup jest utworzenie ścieżki przechodzącej dokładnie raz przez wszystkie kwadraty elementarne. Kwadrat jest identyfikowany numerem kroku, w którym do niego dochodzimy, nazywanym kluczem. Ścieżka taka odpowiada krzywej wypełniającej rozważany fragment przestrzeni.

Krzywa, która wykorzystuje fraktale do wypełnienia przestrzeni, jest nazywana *krzywą fraktalną*. Krzywą fraktalną zapewniającą proste przejście ze współrzędnych na klucz jest *krzywa N-Peano*, [10, 12], nazywana również sekwencją Mortona, krzywą N i krzywą Z (rys. 1). *Klucz N-Peano*, określający położenie kwadratu na krzywej, tworzony jest poprzez prze-

plot bitów poszczególnych współrzędnych. Przyjęto, że parzyste bity klucza N-Peano odpowiadają współrzędnej X, a nieparzyste współrzędnej Y (rys. 2).



Rys. 1. Krzywa N-Peano – pierwszy (na lewo) i drugi (na prawo) poziom rozwinięcia
Fig. 1. N-Peano curve – first (left) and second (right) level of generation



Rys. 2. Związek miedzy kluczem N-Peano i współrzędnymi X, Y
Fig. 2. The relation between N-Peano key and X, Y co-ordinates

Drzewo czwórkowe (ang. *quadtree*) jest strukturą danych klasy drzewo, [14], powstałą poprzez rekurencyjny podział przestrzeni dwuwymiarowej na cztery części o takich samych wymiarach. Podczas aproksymacji obiektu drzewem czwórkowym proces podziałów kończy się, gdy dany kwadrat w całości leży na obszarze obiektu lub gdy osiągnie minimalną wielkość wynikającą z założonej rozdzielczości. *Liniowym drzewem czwórkowym* będzie nazywane drzewo czwórkowe, którego elementy zostały uporządkowane krzywą fraktalną.

Aproksymację obiektów przestrzennych liniowym drzewem czwórkowym uporządkowanym krzywą N-Peano można przechowywać w relacyjnej bazie danych. Każda krotka relacji opisuje jeden kwadrat tworzący aproksymację geometrii obiektu. Ponieważ relacja taka nie spełnia wymagań dla pierwszej postaci normalnej, [10, 15] – jedna krotka opisująca dany kwadrat może odpowiadać wielu krotkom opisującym podział tego kwadratu – wprowadzono w [9, 10] pojęcie relacji Peano. Można wyróżnić kilka schematów relacji Peano, [2, 10]. W dalszej części będzie używany schemat, w którym są pamiętane zakresy klucza N-Peano odpowiadające kwadratom:

- *SchemRelPeano (Id obiektu, KluczPeanoPoczatku, KluczPeanoKonca, Atrybuty)*, gdzie:
  - *IdObiektu* – identyfikator obiektu,
  - *KluczPeanoPoczatku* – klucz N-Peano lewego dolnego kwadratu elementarnego (minimalna wartość klucza Peano odpowiadająca kwadratowi),
  - *KluczPeanoKonca* – klucz N-Peano prawego górnego kwadratu elementarnego (maksymalna wartość klucza N-Peano odpowiadająca kwadratowi),

- *Atrybuty* - atrybuty opisowe, związane z danym fragmentem obiektu o identyfikatorze *IdObiektu*.

Dla relacji Peano zdefiniowano poziomy poprawności rozszerzające postacie normalne o reguły integralności, jakie musi spełniać zapis liniowego drzewa czwórkowego. Poziomy poprawności relacji Peano i postacie normalne relacji ułatwiają utrzymanie integralności danych. Wyróżnia się trzy poziomy poprawności relacji Peano, [10]:

1. Relacja Peano jest na *pierwszym poziomie poprawności* (ang. First Conformance Level), jeżeli wszystkie tworzące ją kwadraty są prawidłowo położone. Kwadrat jest prawidłowo położony, jeżeli powstał w procesie tworzenia drzewa czwórkowego.
2. Relacja Peano jest na *drugim poziomie poprawności* (ang. Second Conformance Level), jeżeli spełnia warunki pierwszego poziomu poprawności i nie występują w niej nakładające się kwadraty.
3. Relacja Peano jest na *trzecim poziomie poprawności* (ang. Third Conformance Level), jeżeli spełnia warunki drugiego poziomu poprawności i aproksymowany obszar jest opisany minimalną liczbą kwadratów – kwadraty tworzące większy kwadrat zostały połączone.

3. Aproksymacja rozkładu za pomocą liniowego drzewa czwórkowego

Zgodnie z założeniem *Z4* analizowany obszar jest dzielony na elementarne kwadraty, których wielkość jest zadawana przez użytkownika. Elementarne kwadraty są identyfikowane kluczem N-Peano. Kwadraty łączone zgodnie z regułami budowy drzewa czwórkowego tworzą zakresy klucza N-Peano. Przyjęto konwencję, że zadane przez użytkownika rozłączne przedziały wartości badanej cechy (założenie *Z3*) są numerowane kolejnymi liczbami naturalnymi, tak że numer *1* otrzymuje przedział o najmniejszych wartościach granic. Użytkownik zapisując warunki, posługuje się numerami przedziałów.

Zgodnie z przyjętymi założeniami rozkład przestrzenny może być rozpatrywany jako obiekt przestrzenny, z którym są związane dwie grupy danych:

1. grupa pierwsza – dane opisujące obszary należące do poszczególnych przedziałów,
2. grupa druga – dane opisujące proces tworzenia rozkładu (użyty zestaw danych pomiarowych, metodę generacji oraz jej parametry, granice przedziałów, zakres współrzędnych i rozdzielczość).

Z punktu widzenia użytkownika zadającego pytanie rozkład będzie utożsamiany z pierwszą grupą danych. Dane drugiej grupy będą traktowane jako metadane, przechowywane w

specjalnym słowniku rozkładów. W tym sensie rozkład możemy traktować jako relację przestrzenną o schemacie:

<NazwaRozkładu> (NrPrzedziału, Obszar),

gdzie:

- *<NazwaRozkładu>* – określa unikalną nazwę relacji przestrzennej przechowującej rozkład, na podstawie której w słowniku rozkładów można znaleźć wszystkie dane potrzebne do wygenerowania rozkładu,
- *NrPrzedziału* – określa numer przedziału opisywanego przez daną krotkę, każdy przedział jest opisywany przez jeden wiersz, tak więc ten atrybut stanowi klucz relacji,
- *Obszar* – określa wszystkie kwadraty należące do przedziału określonego atrybutem *NrPrzedziału*.

Przedstawiony schemat relacji stanowi model logiczny rozkładu – tak jak użytkownik widzi rozkład w zapytaniach – i nie przesądza o sposobie przechowywania rozkładu w bazie danych.

Przy takich założeniach, rozkład przestrzenny możemy traktować jako funkcję przyporządkowującą każdemu elementarnemu kwadratowi numer przedziału wartości. Oznaczmy przez *KP* zbiór kluczy N-Peano odpowiadających elementarnym kwadratom, na które został zdekomponowany rozpatrywany obszar, a przez *P* zbiór numerów rozłącznych przedziałów zadanych przez użytkownika. Przy takich założeniach rozkład można zapisać jako funkcję *r*, której dziedziną jest zbiór kluczy N-Peano, przyjmującą wartości ze zbioru numerów przedziałów:

$$r: KP \rightarrow P. \tag{1}$$

Wykorzystywane sposoby generacji rozkładu umożliwiają budowę liniowego drzewa czwórkowego metodą zstępującą – w kolejnych przebiegach tworzone są kolejne poziomy drzewa. Do podejmowania decyzji, czy dany kwadrat można przyporządkować do przedziału, czy należy go dalej dzielić, służy kilka metod opisanych w [3]. Podejście takie powoduje konieczność rozszerzenia definicji funkcji *r* (danej wzorem (1)), tak że jej dziedziną stają się dozwolone zakresy klucza N-Peano. Zakres klucza N-Peano jest dozwolony, jeżeli odpowiadający mu kwadrat mógł powstać w procesie budowy drzewa czwórkowego. Niech zbiór wszystkich dozwolonych zakresów klucza N-Peano będzie oznaczany przez *ZKP*. Wartość *-1* oznacza, że danemu kwadratowi na bieżącym poziomie rozwinięcia drzewa nie można przypisać przedziału. W procesie generacji rozkładu odpowiada to sytuacji, w której należy kwadrat podzielić na cztery mniejsze i dla nich badać przynależność do przedziałów. Funkcja rozkładu przybiera postać:

$$r: ZKP \rightarrow \{P \cup \{-1\}\}. \tag{2}$$

Wzór (2) nie wymaga, aby w trakcie wyznaczania rozkładu dokonywać dekompozycji rozpatrywanego obszaru na kwadraty elementarne.

4. Charakterystyka zadań wyszukiwania dotyczących rozkładów przestrzennych

Z założenia każde z rozważanych zapytań dotyczy przynajmniej jednego rozkładu przestrzennego. Poza tym może odwoływać się do:

- danych opisowych (przechowywanych w relacyjnej bazie danych),
- danych wektorowych (przechowywanych w systemie informacji przestrzennej lub w relacyjnej bazie danych).

Można przyjąć, że wiele zapytań będzie dotyczyło typowych obszarów (np. obszar województwa), w typowych skalach, dla przedziałów ustalonych na podstawie norm zanieczyszczenia. Umożliwia to wcześniejsze przygotowanie rozkładów, aproksymacji obiektów administracyjnych i danych zagregowanych. Jeżeli zadany w zapytaniu rozkład nie został uprzednio utworzony, należy go wyznaczyć w trakcie wykonania zapytania. Cecha ta jest charakterystyczna dla analizowanej klasy zapytań i znacząco wpływa na ich optymalizację.

Ze względu na format odpowiedzi zapytania możemy podzielić na:

- dające zwrotnie odpowiedź w formacie tekstowym (tabelarycznym), np. *Podaj nazwy gmin, na terenie których w roku 1998 średnie zanieczyszczenie ołowiem należało do przedziału 2 lub wyższego; dla każdej gminy podaj również wielkość tego obszaru;*
- dające zwrotnie odpowiedź graficzną, np. *Przedstaw rozkład średniego zanieczyszczenia ołowiem w roku 1998; pokaż granice gmin;* w takim przypadku odpowiedzią jest mapa barwna, na której przedziały wartości zaznaczono kolorami; na takim podkładzie wyświetlane są granice administracyjne.

Podział ten ma również wpływ na przetwarzanie zapytań, ponieważ dla zapytań z odpowiedzią tekstową należy dokonać przestrzennego złączenia rozkładów z innymi obiektami przestrzennymi (obiektami grupy drugiej). Dla większości zapytań graficznych wystarczy nałożyć na obraz rozkładu zadane obiekty grupy drugiej.

Warunki dotyczące rozkładów można podzielić na dwie grupy:

1. warunki dotyczące numeru przedziału,
2. warunki określające zachodzenie relacji przestrzennych między rozkładami oraz między rozkładami a innymi obiektami przestrzennymi.

W zapytaniu najpierw wykonywane są warunki grupy 1 – wszystkie wybrane w zapytaniu przedziały danego rozkładu są scalane i traktowane jako jeden obiekt przestrzenny. Wynik operacji przestrzennej może zostać wykorzystany do obliczenia funkcji agregującej, np. wy-

znaczenia powierzchni należących do zadanych przedziałów. Ważona suma zanieczyszczeń (indeks zanieczyszczeń) jest traktowana tak jak pojedyncze zanieczyszczenie – obliczane są wartości indeksu w punktach pomiarowych i wyznaczany jest jego rozkład przestrzenny.

Warunek na część wspólną obiektów w zapytaniach może występować w dwóch znaczeniach:

- predykatu, gdy chodzi o sprawdzenie istnienia części wspólnej, np. *Podaj nazwy gmin, na terenie których dane zanieczyszczenie należy do przedziału 3*,
- operatora wyznaczającego część wspólną, np. *Podaj nazwy gmin, których co najmniej 20% obszaru jest zanieczyszczonych na poziomie przedziału 3*.

Podział ten ma znaczenie z punktu widzenia optymalizatora zapytań, ponieważ sprawdzenie istnienia części wspólnej można wykonać szybciej niż jej wyznaczenie.

Podczas zapisu zapytań użytkownik posługuje się współrzędnymi geograficznymi. W trakcie wykonania tych zapytań, w związku z dyskretyzacją przestrzeni, współrzędne geograficzne są przekształcane do współrzędnych obliczeniowych. *Współrzędne obliczeniowe* określają położenie w jednostkach równych bokowi kwadratu elementarnego i są używane do tworzenia klucza N-Peano.

Przedstawione podejście nie uwzględnia zapytań dotyczących wartości liczonych dla zadanych obszarów (np. podaj średnie zanieczyszczenie dla gminy). Odpowiedzieć na taką klasę zapytań można uzyskać wykonując obliczenia na rastrze i uśredniając wartości w kwadratach elementarnych należących do poszczególnych obszarów.

Oprócz możliwości formułowania zapytań należy użytkownikowi udostępnić polecenia pozwalające na:

- zachowanie rozkładu w bazie,
- usunięcie rozkładu z bazy,
- przedstawienie planu wykonania zapytania.

5. Operacje

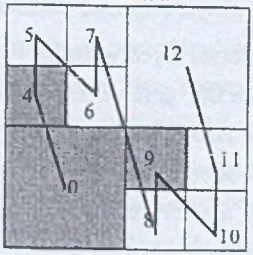
W tym rozdziale omówione zostaną operacje wymagane do zapisu oraz przetwarzania zapytań dotyczących rozkładów. Operacje selekcji, projekcji i złączenia na atrybutach opisowych obiektów przestrzennych mają taką samą semantykę jak w algebrze relacji, [15]. Operacja projekcji dla atrybutów przestrzennych ma takie samo znaczenie jak dla atrybutów opisowych. Dodatkowo, przy współpracy z systemem przetwarzającym dane wektorowe, są potrzebne operacje konwersji postaci wektorowej na płatową (podczas pobierania danych) i płatowej na wektorową (podczas eksportu wyników).

Operacje algebry Peano pozwalają na przetwarzanie obiektów zapisanych w postaci relacji Peano bez wykorzystywania geometrii obliczeniowej. Rozszerzają one algebrę relacji o możliwość przetwarzania danych na podstawie ich położenia w przestrzeni. W [9, 10] wyróżniono trzy grupy operacji:

- boolowskie: część wspólna, suma i różnica; operacje tej grupy na podstawie dwóch podanych obiektów tworzą nowy obiekt, wykorzystując odpowiednią operację logiczną;
- geometryczne: translacja, obrót, skalowanie, symetria, obcięcie do podanego okna, powielenie i redukcja; operacje tej grupy tworzą nowy obiekt stosując odpowiednie przekształcenie geometryczne do podanego obiektu;
- relacyjne: projekcja geometryczna i złączenie Peano; operacje tej grupy działają na relacjach Peano.

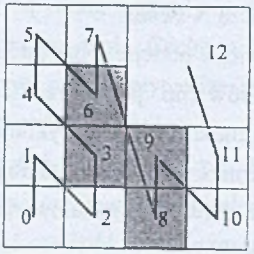
W celu zapisania zapytań użytkownikowi należy udostępnić operacje boolowskie i funkcję obliczającą powierzchnię. Pozostałe operacje, poza powieleniem i projekcją geometryczną, mogą zostać wykorzystane w trakcie wykonania zapytań.

Obiekt a



IdOb	KPmin	KPmax
a	0	3
a	4	4
a	9	9

Obiekt b



IdOb	KPmin	KPmax
b	3	3
b	6	6
b	8	8
b	9	9

Rys. 3. Obiekty *a* i *b* wykorzystywane w przykładach operacji algebry Peano
Fig. 3. Objects *a* and *b* used in examples of Peano algebra operations

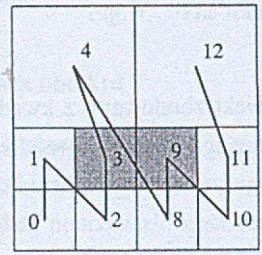
Operacje algebry Peano zostaną omówione na przykładzie obiektów *a* i *b* przedstawionych na rys. 3. Dla uproszczenia, obiekty nie posiadają żadnych atrybutów – zapisywane jest tylko ich położenie. *ZKP* oznacza zbiór dozwolonych zakresów klucza N-Peano (pojęcie szczegółowo omówiono w rozdziale 3), a *zkp* – dozwolony zakres klucza N-Peano. W deklaracjach operacji jest używany typ *TAproks*, oznaczający typ struktury danych przechowującej aproksymację obszaru zajmowanego przez obiekt.

5.1. Część wspólna obiektów

Operacja wyznaczenia części wspólnej (przecięcia) dwóch obiektów tworzy nowy obiekt zbudowany z kwadratów elementarnych, które należą do obydwu obiektów (rys. 4). Formalną definicję w oparciu o klucz N-Peano przedstawia wzór (3).

$$a \cap b = \{zkp : zkp \in ZKP \wedge zkp \in a \wedge zkp \in b\}. \tag{3}$$

Operacja jest wykonywana poprzez przeglądanie kwadratów obydwu obiektów, uporządkowanych według klucza N-Peano. Jeżeli kwadraty nakładają się na siebie, wyznaczana jest ich część wspólna. Sprawdzenie nakładania się kwadratów sprowadza się do kontroli nakładania się na siebie dwóch odcinków na krzywej N-Peano. Wynik jest na trzecim poziomie poprawności.



IdOb	KPmin	KPmax
$a \cap b$	3	3
$a \cap b$	9	9

Rys. 4. Część wspólna obiektów *a* i *b*
Fig. 4. The intersection of the objects *a* and *b*

Deklaracja operacji:

TAproks& Intersect (TAproks& Obiekt1, TAproks& Obiekt2).

Ze względu na większą efektywność sprawdzania części wspólnej, niż jej wyznaczania, wprowadzono predykat:

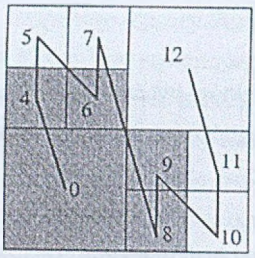
bool IntersectB (TAproks& Obiekt1, TAproks& Obiekt2).

5.2. Suma obiektów

Operacja sumy dwóch obiektów tworzy nowy obiekt zbudowany z kwadratów, które należą do jednego lub drugiego obiektu (rys. 5). Formalną definicję w oparciu o klucz N-Peano przedstawia wzór (4).

$$a \cup b = \{ zkp : zkp \in ZKP \wedge (zkp \in a \vee zkp \in b)\}. \tag{4}$$

Po wykonaniu operacji należy wyeliminować nakładające się kwadraty oraz dokonać ich scalania.



IdOb	KPmin	KPmax
$a \cup b$	0	3
$a \cup b$	3	3
$a \cup b$	4	4
$a \cup b$	6	6
$a \cup b$	8	8
$a \cup b$	9	9
$a \cup b$	9	9

IdOb	KPmin	KPmax
$a \cup b$	0	3
$a \cup b$	4	4
$a \cup b$	6	6
$a \cup b$	8	8
$a \cup b$	9	9

Rys. 5. Suma obiektów *a* i *b*, relacja Peano przed normalizacją i po normalizacji
Fig. 5. The sum of the objects *a* and *b*, Peano relation before and after normalisation

Deklaracja operacji:

TAproks& Sum (TAproks& Obiekt1, TAproks& Obiekt2).

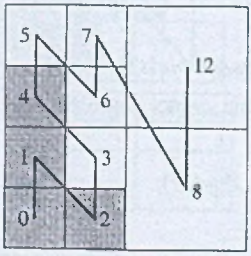
5.3. Różnica obiektów

Operacja różnicy dwóch obiektów tworzy nowy obiekt zbudowany z kwadratów, które należą do obiektu pierwszego i nie należą do obiektu drugiego (rys. 6). Formalną definicję w oparciu o klucz N-Peano przedstawia wzór (5).

$$a - b = \{ zkp : zkp \in ZKP \wedge zkp \in a \wedge zkp \notin b \}. \tag{5}$$

Deklaracja operacji:

TAproks& Diff (TAproks& Obiekt1, TAproks& Obiekt2).



IdOb	KPmin	KPmax
$a - b$	0	0
$a - b$	1	1
$a - b$	2	2
$a - b$	4	4

Rys. 6. Różnica obiektów *a* i *b*
Fig. 6. The difference of objects *a* and *b*

5.4. Translacja obiektu

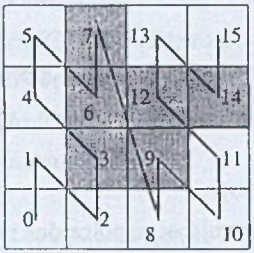
Operacja translacji obiektu tworzy nowy obiekt poprzez translację wszystkich kwadratów przesuwanego obiektu o zadany wektor (rys. 7). Utworzony obiekt wymaga normalizacji.

Deklaracja operacji:

TAproks& Trans (TAproks& Obiekt, float DX, float DY).

Wykonanie translacji o wektor będący całkowitą wielokrotnością boku kwadratu sprowadza się do przekształcenia kluczy N-Peano na współrzędne X i Y, ich zmiany i wyznaczenia nowych kluczy N-Peano. Przesunięcie o wektor innej długości powoduje, że kwadraty cle-

mentarne nie są przekształcane na inne kwadraty elementarne. Prowadzi to do aproksymacji aproksymacji i zmniejszenia dokładności reprezentacji obiektu wyjściowego. Powstająca po translacji relacja wymaga pełnej normalizacji.



IdOb	KPmin	KPmax
a + (1, 1)	3	3
a + (1, 1)	6	6
a + (1, 1)	7	7
a + (1, 1)	9	9
a + (1, 1)	12	12
a + (1, 1)	14	14

Rys. 7. Translacja obiektu *a* o wektor (1, 1)
Fig. 7. The translation of object *a* by vector (1, 1)

5.5. Obrót obiektu

Operacja obrotu obiektu o zadany kąt wokół początku układu polega na dokonaniu obrotu każdego kwadratu elementarnego. Operację obrotu obiektu względem dowolnego punktu można uzyskać poprzez złożenie translacji z obrotem wokół początku układu. Jeżeli obrót jest dokonywany o 90° lub jego całkowitą wielokrotność, kwadraty elementarne zostają przekształcone na inne kwadraty elementarne i wymagane jest tylko przekształcenie kluczy N-Peano i pełna normalizacja. Przy innych kątach należy każdy obrócony kwadrat aproksymować kwadratami elementarnymi. Prowadzi to do aproksymacji aproksymacji i zmniejszenia dokładności reprezentacji obiektu wyjściowego. Powstająca po obrocie relacja wymaga pełnej normalizacji.

Deklaracja operacji:

TAproks& Rotation (TAproks& Obiekt, float Kąt).

5.6. Skalowanie obiektu

Wykonanie operacji skalowania zależy od współczynnika skali. Najłatwiej jest zrealizować tę operację dla dwukrotnego powiększenia obiektu - klucze N-Peano wszystkich kwadratów są mnożone przez 4, a długości boków przez 2. Dwukrotne zmniejszenie skali odpowiada operacji odwrotnej. Dla elementów o długości boków 1 należy przeprowadzić ich redukcję. Skalowanie o współczynnik będący potęgą liczby 2 sprowadza się do złożenia przedstawionych wariantów operacji skalowania. Skalowanie o inne współczynniki wymaga dekompozycji aproksymacji na kwadraty elementarne i wyznaczenie nowej aproksymacji. Prowadzi to do aproksymacji aproksymacji i zmniejszenia dokładności reprezentacji obiektu wyjściowego. Powstająca po skalowaniu relacja wymaga pełnej normalizacji.

Deklaracja operacji:

TAproks& Scale (TAproks& Obiekt, float Skala).

5.7. Obcięcie obiektu do podanego okna

Obcięcie obiektu polega na pominięciu kwadratów elementarnych, które całkowicie leżą poza podanym oknem.

Deklaracja operacji:

TAproks& Window (TAproks& Obiekt, TRect Okno),

gdzie *TRect* oznacza typ struktury danych przechowującej współrzędne prostokąta.

5.8. Redukcja obiektu

Operacja redukcji obiektu tworzy nowy obiekt poprzez pominięcie szczegółów podanego obiektu. Wszystkie kwadraty o boku mniejszym niż zadana wartość graniczna są zastępowane większymi kwadratami lub pomijane. Podczas redukcji rozkładu problem występuje w przypadku, gdy kwadraty, które mają zostać scalone w jeden kwadrat wyższego poziomu, należą do różnych przedziałów, szczególnie gdy dwa należą do jednego przedziału, a dwa pozostałe do innego.

Deklaracja operacji:

TAproks& Generalization (TAproks& Obiekt, int NowyPoziomRozwinięcia).

5.9. Złączenie Peano

Złączenie Peano oznacza wykonanie złączenia dwóch relacji Peano według klucza N-Peano. Jej wynikiem są identyfikatory obiektów, których zakresy klucza N-Peano nakładają się.

5.10. Normalizacja aproksymacji obiektów

W rozdziale 2 przedstawiono poziomy poprawności relacji Peano. W trakcie wykonania zapytań wykorzystywane są operacje doprowadzające podaną aproksymację do wymaganego poziomu poprawności. Operacje doprowadzające do drugiego i trzeciego poziomu poprawności mogą występować w dwóch wariantach, różniących się czasem wykonania. Pierwszy wariant zakłada, że dane są na poprzednim poziomie poprawności. Drugi sprawdza, czy poprzedni poziom jest zachowany. Jeżeli nie jest, doprowadza dane do tego poziomu.

Poniżej podano deklaracje funkcji doprowadzających aproksymacje do pierwszego, drugiego i trzeciego poziomu poprawności.

- *TAproks& FrstConfLev (TAproks& Obiekt)* – doprowadzenie aproksymacji do pierwszego poziomu poprawności,
- *TAproks& ScndConfLev (TAproks& Obiekt)* – doprowadzenie aproksymacji do drugiego poziomu poprawności,
- *TAproks& ThrdConfLev (TAproks& Obiekt)* – doprowadzenie aproksymacji do trzeciego poziomu poprawności.

5.11. Obliczenie powierzchni

Dla podanej jako argument aproksymacji funkcja daje w wyniku jej powierzchnię, obliczając liczbę tworzących ją kwadratów elementarnych.

Deklaracja funkcji:

int Area (TAproks& Obiekt).

6. Optymalizacja zapytań

6.1. Uwagi ogólne

Optymalizator będzie dostawał zapytanie zapisane jako ciąg operacji algebraicznych. Jego zadaniem będzie znalezienie optymalnego planu wykonania tych operacji.

Optymalizację rozpatruje się na dwóch poziomach, [15]: przekształceń algebraicznych oraz dostępu do danych. Rozdział ten koncentruje się na optymalizacji na poziomie przekształceń algebraicznych, który jest w dużym stopniu niezależny od szczegółów implementacyjnych.

Badanie wszystkich możliwych planów wykonania nawet dla średnio złożonych zapytań jest nieefektywne ze względu na ich dużą liczbę. Dlatego należy przyjąć reguły umożliwiające wyłonienie rozsądnej liczby dobrych planów, spośród których zostanie wybrany najlepszy na podstawie analizy kosztów. Łączny czas generacji planów, ich oceny i wykonania zapytania musi być średnio mniejszy od planu wybieranego losowo i dużo mniejszy od najgorszego planu.

Sensowne wydaje się przyjęcie reguły, że podzapytania są dzielone na grupy odpowiadające źródłom danych. Wykonanie jednych z tych podzapytań może służyć do nałożenia dodatkowych warunków na pozostałe podzapytania i skrócenia czasu ich wykonania. Szeregując podzapytania należy mieć na uwadze, że koszt pojedynczej operacji na obiekcie jest najniższy

w podzapytaniu dotyczącym atrybutów opisowych, operacje przestrzenne na danych w postaci wektorowej są czasochłonne, a dla reprezentacji powierzchniowej czas pobierania danych jest dominujący, [1]. Podzapytania dotyczące atrybutów opisowych powinny być tak formułowane, aby jak najlepiej wykorzystać standardowe reguły optymalizacji serwera.

Dla danych w postaci wektorowej można wyróżnić dwie strategie wykonania podzapytań:

1. wykonanie operacje w postaci wektorowej w systemie informacji przestrzennej (projektowany system z założenia nie będzie wykonywał operacji na danych wektorowych) i przekształcenie wyniku do reprezentacji płatowej,
2. konwersja z postaci wektorowej do płatowej i wykonanie operacji w reprezentacji płatowej.

Podejścia te różnią się dokładnością i czasem obliczeń. Użycie pierwszej strategii powoduje sumowanie się błędów powodowanych przez zaokrąglenia. Szczególnie jest to zauważalne przy wyznaczaniu punktu przecięcia prostych prawie równoległych, [8]. Przy wielu operacjach można otrzymać różne wyniki w zależności od kolejności operacji. Największy wpływ będzie to miało na zapytania o występowanie bądź niewystępowanie na zadanym obszarze zanieczyszczenia z danego przedziału, jeżeli obszar ten leży na granicy plamy zanieczyszczenia. Wykorzystanie drugiej strategii powoduje powstanie błędu w trakcie konwersji z postaci wektorowej na płatową, związanego z określoną rozdzielczością tej drugiej. Dalsze etapy przetwarzania nie wprowadzają błędu, ponieważ po sprowadzeniu wszystkich obiektów do współrzędnych obliczeniowych, wykonywane są tylko operacje boolowskie wykorzystujące operacje porównania i odejmowania liczb całkowitych.

Warunki w zapytaniu są zapisywane przy użyciu współrzędnych geograficznych. W trakcie wykonania zapytania wykorzystywane są współrzędne obliczeniowe. Pociąga to za sobą przekształcenie wszystkich danych wektorowych do postaci płatowej. Jeżeli istnieją w bazie aproksymacje potrzebnych obiektów, należy sprawdzić, czy odpowiadają one rozdzielczości i zakresowi współrzędnych zapytania. Jeżeli nie, należy rozważyć, czy taniej jest przekształcić aproksymacje, czy utworzyć je na nowo w zadanych współrzędnych. Przy przekształcaniu używane są operacje translacji, skalowania, obrotu i redukcji, które mogą znacząco zwiększyć błąd aproksymacji.

Podczas optymalizacji zapytań rozważanej klasy należy brać pod uwagę dwa cele: zmniejszenie czasu wykonania zapytań i zwiększenie dokładności uzyskanych odpowiedzi. Wydaje się, że w wielu zapytaniach poprawienie jednego kryterium będzie prowadzić do pogorszenia drugiego i odwrotnie. Decyzję, czy zmniejszyć dokładność i szybciej wykonać zapytanie, czy dłużej poczekać na dokładniejszy wynik, powinien podejmować użytkownik.

Jeżeli w zapytaniu zadano generację rozkładu, należy sprawdzić, czy taki rozkład jest zapisany w bazie i czy jego pobranie będzie szybsze od jego generacji.

Poniżej podano ogólne reguły optymalizacji zaadaptowane z baz relacyjnych:

- wczesna projekcja na atrybutach przestrzennych (przy odczytywaniu obiektów przestrzennych nie pobierać atrybutów, które nie będą wykorzystywane, np. dla obiektów administracyjnych nie pobierać granic, jeżeli są potrzebne tylko aproksymacje powierzchni),
- opóźnienie odczytu atrybutów przestrzennych (jeżeli istnieje możliwość wyeliminowania części obiektów przestrzennych na podstawie atrybutów opisowych, należy to wykonać w pierwszej kolejności, a dopiero dla tak zredukowanego zbioru obiektów odczytać wymagane atrybuty przestrzenne),
- jeżeli dana operacja akceptuje dane na poziomie poprawności równym lub niższym niż poziom poprawności danych dostarczonych przez poprzednią operację, należy pominąć operację doprowadzającą do wyższego poziomu poprawności,
- jeżeli dana operacja wymaga danych na poziomie poprawności wyższym niż poziom poprawności danych dostarczonych przez poprzednią operację, doprowadzenie do wymaganego poziomu poprawności należy dołączyć do wykonania poprzedniej operacji (prowadzi to do różnych wersji tej samej operacji, zwracających dane na różnych poziomach poprawności),
- wykonanie selekcji na warunkach nałożonych na numer przedziału na etapie pobierania rozkładu z bazy lub w trakcie generacji rozkładu,
- sprawdzenie możliwości wykorzystania w zapytaniu danych zagregowanych, [1], np. wcześniej wyznaczonych powierzchni gmin należących do poszczególnych przedziałów danego zanieczyszczenia.

6.2. Powiązanie selekcji z generacją rozkładu

Jeżeli w trakcie wykonywania zapytania wymagane jest utworzenie rozkładu (nie jest on dostępny w bazie lub szacowany czas jego utworzenia jest mniejszy niż czas odczytu) i zapytanie dotyczy tylko wybranych przedziałów rozkładu, można zredukować czas obliczeń odrzucając w trakcie generacji rozkładu te kwadraty, które nie należą do zadanych w pytaniu przedziałów. Odpowiada to połączeniu operacji generacji rozkładu z selekcją. Algorytm zapisany w pseudojęzyku przedstawiono w rozdziale 8.1.

6.3. Optymalizacja wyznaczania części wspólnej obiektów

Wyznaczanie części wspólnej obiektów zasługuje na szczególne potraktowanie jako operacja często występująca w zapytaniach. Optymalizator ma do wyboru wyznaczenie części wspólnej w pamięci operacyjnej lub na serwerze. W [1] przedstawiono ocenę efektywności wyznaczania części wspólnej na serwerze relacyjnej bazy danych oraz zaproponowano kilka

metod przyśpieszania tej operacji. Podczas optymalizacji należy odróżnić operację wyznaczania części wspólnej od operacji badania, czy część wspólna istnieje, ponieważ drugą operację można wykonać znacznie szybciej. Rozdział ten zajmuje się wyznaczeniem części wspólnej obiektów w pamięci operacyjnej.

Prostym testem, pozwalającym zredukować ilość obliczeń, jest wyznaczenie minimalnej i maksymalnej wartości klucza N-Peano dla wszystkich kwadratów aproksymujących położenie danego obiektu i sprawdzenie nakładania się tych zakresów klucza. Podejście to jest ograniczone własnością krzywej N-Peano, polegającą na tym, że sąsiednie kwadraty mogą się znacząco różnić wartością klucza. Jeżeli zakresy klucza N-Peano nie nakładają się, można stwierdzić, że część wspólna jest zbiorem pustym. W przeciwnym wypadku konieczne jest porównanie zakresów klucza odpowiadających poszczególnym kwadratom.

Oznaczmy przez A i B dwa zbiory obiektów, zawierające odpowiednio k i l elementów, a przez EK zbiór kwadratów elementarnych. Niech a_i i b_i oznaczają i-ty obiekt odpowiednio ze zbioru A i zbioru B. Niech n_{ai} i n_{bi} oznaczają liczbę kwadratów odpowiednio elementów a_i i b_i.

Część wspólna (przecięcie) dwóch obiektów została zdefiniowana w rozdziale *5.1*, wzór (3). Częścią wspólną dwóch zbiorów obiektów jest zbiór obiektów, powstający w wyniku wykonania operacji przecięcia na wszystkich parach obiektów, będących wynikiem iloczynu kartezjańskiego zbiorów wejściowych. Formalnie można to zapisać za pomocą wzoru (6).

$$A \cap B = C = \{c_i : c_i = a_i \cap b_i \wedge a_i \in A \wedge b_i \in B\}. \tag{6}$$

Cechą rozkładu jest jednoznaczne kwalifikowanie elementarnych kwadratów do przedziałów – dany kwadrat może należeć tylko do jednego przedziału i może tam zostać zakwalifikowany tylko raz (drugi poziom poprawności). Podobnie obiekty administracyjne na tym samym poziomie są rozłączne i spełniają drugi poziom poprawności (dany kwadrat elementarny może należeć tylko do jednej gminy; może natomiast należeć jednocześnie do dwóch obiektów na różnych poziomach, np. gminy i województwa). Warunek na rozłączność obiektów można formalnie wyrazić za pomocą wzoru (7).

$$\forall a_i, a_j \in A, i \neq j : a_i \cap a_j = \varnothing. \tag{7}$$

6.3.1. Scalenie obszarów przed złączeniem

Zgodnie z punktem *5.1* wyznaczenie części wspólnej dwóch obiektów sprowadza się do wyznaczenia nakładania się dwóch serii odcinków na krzywej N-Peano. Niech n_1 będzie liczbą kwadratów aproksymacji obiektu *1*, a n_2 liczbą kwadratów aproksymacji obiektu *2*. Odcinki każdej z aproksymacji są uporządkowane według klucza N-Peano i są wzajemnie rozłączne. Algorytm przegląda sekwencyjnie odcinki, eliminując w każdym kroku jeden odcinek. W sumie jest $n_1 + n_2$ odcinków. Do sprawdzenia nakładania się dwóch odcinków wy-

starczają dwa porównania. Maksymalnie jest wykonywanych $2 * (n_1 + n_2)$ porównań. Średni czas działania jest liniowy:

$$O(n_1 + n_2). \tag{8}$$

Wyznaczenie części wspólnej dwóch zbiorów obiektów można zrealizować poprzez porównywanie każdego obiektu z pierwszego zbioru z każdym obiektem z drugiego. Prowadzi to do $k * l$ porównań obiektów, co w najgorszym przypadku daje liczbę porównań kwadratów rzędu:

$$O\left(l * \sum_{i=1}^{k} n_{a_i} + k * \sum_{i=1}^{l} n_{b_i} \right). \tag{9}$$

W takim podejściu nie jest wykorzystywana własność rozłączności obiektów i wiele porównań jest wykonywanych niepotrzebnie.

Czas wyznaczenia złączenia przestrzennego dwóch zbiorów obiektów spełniających warunek (7) można zredukować poprzez scalenie aproksymacji obiektów i ich uporządkowanie względem klucza N-Peano. Po scaleniu i uporządkowaniu kwadratów otrzymujemy dwa ciągi zakresów klucza N-Peano. Przez analogię do wzoru (8) czas wyznaczenia części wspólnej zależy liniowo od liczby kwadratów tworzących aproksymacje obydwu zbiorów:

$$O\left(\sum_{i=1}^{k} n_{a_i} + \sum_{i=1}^{l} n_{b_i} \right). \tag{10}$$

Do tego należy jednak doliczyć koszt sortowania podczas scalania obiektów. Pierwszym nasuwającym się rozwiązaniem jest wykorzystanie często używanego algorytmu *QuickSort*. Jego średni czas działania jest rzędu $O(n * lg\ n)$, gdzie symbol lg oznacza logarytm o podstawie 2 ($lg\ n \equiv log_2\ n$). Dla analizowanych danych średni czas działania algorytmu *QuickSort* wyniesie:

$$O\left(\sum_{i=1}^{k} n_{a_i} * \lg\left(\sum_{i=1}^{k} n_{a_i} \right) + \sum_{i=1}^{l} n_{b_i} * \lg\left(\sum_{i=1}^{l} n_{b_i} \right) \right). \tag{11}$$

Algorytm *QuickSort* nie wykorzystuje uporządkowania kwadratów każdego z obiektów. Zastosowanie sortowania przez scalanie daje średni czas działania rzędu:

$$O\left(\lg k * \sum_{i=1}^{k} n_{a_i} + \lg l * \sum_{i=1}^{l} n_{b_i} \right). \tag{12}$$

Możliwe jest również posortowanie w czasie liniowym przy użyciu sortowania przez zliczanie, [6], jednakże wymaga ono utworzenia tablicy o wielkości odpowiadającej zakresowi klucza N-Peano.

Aproksymacje obiektów składają się z wielu kwadratów – w zależności od rozdzielczości liczba ta jest rzędu setek, tysięcy lub większa. Wartość logarytmu z liczby obiektów (wzór (12)) jest mniejsza od wartości logarytmu z liczby kwadratów (wzór (11)). Wynika z tego, że

sortowanie przez scalanie dla rozpatrywanych danych działa szybciej niż *QuickSort*. Sortowanie przez scalanie będzie wolniejsze od sortowania przez zliczanie o logarytm z liczby obiektów.

Średni czas wyznaczenia części wspólnej razem ze scaleniem (złożenie wzorów (10) i (12)) jest rzędu:

$$O\left((\lg k+1)*\sum_{i=1}^{k} n_{a_i}+(\lg l+1)*\sum_{i=1}^{l} n_{b_i}\right). \tag{13}$$

Czas wyznaczenia części wspólnej bez scalania zależy od liczby obiektów (wzór (9)), użycie scalania powoduje, że czas ten zależy od logarytmu z liczby obiektów (wzór (13)). Tak więc scalenie kwadratów tworzących aproksymacje obiektów pozwala przyśpieszyć wyznaczenie części wspólnej dwóch zbiorów rozłącznych obiektów.

6.3.2. Wyznaczanie części wspólnej obiektów punktowych i powierzchniowych

Szczególnym przypadkiem wyznaczania części wspólnej jest badanie, czy kwadrat elementarny, aproksymujący obiekt punktowy, należy do obiektu powierzchniowego, aproksymowanego dużą liczbą kwadratów, oznaczoną przez n. Ponieważ kwadraty aproksymujące obiekt są uporządkowane według klucza N-Peano i rozłączne, można zastosować wyszukiwanie binarne, co pozwala sprawdzić istnienie części wspólnej przy wykorzystaniu $O(\lg n)$ porównań. W przypadku szukania części wspólnej ze zbiorem obiektów powierzchniowych, ze względu na koszt operacji podczas scalania, należy niezależnie badać istnienie części wspólnej z każdym z tych obiektów.

Wyznaczenie części wspólnej zbioru obiektów punktowych i złożonego obiektu powierzchniowego można przyśpieszyć zawężając zakres kwadratów branych pod uwagę przy kolejnych wyszukiwaniach binarnych. W tym celu należy na początku posortować obiekty punktowe według wartości klucza N-Peano (np. rosnąco). Następnie dla obydwu zbiorów należy wyznaczyć minimalną i maksymalną wartość klucza N-Peano. Wartości tych można użyć do odrzucenia kwadratów, dla których zakresy klucza N-Peano na pewno nie będą się pokrywać. Dla kolejnych obiektów punktowych należy użyć wyszukiwania binarnego. W kolejnych krokach pomijane są kwadraty obiektu powierzchniowego o kluczach N-Peano mniejszych od klucza aktualnie przetwarzanego obiektu punktowego. Pozwala to zmniejszyć liczbę kwadratów, które należy brać pod uwagę dla kolejnych obiektów punktowych. Podejście to można również wykorzystać do wyznaczenia przecięcia obiektów, których aproksymacje znacząco różnią się liczbą kwadratów.

6.3.3. Powiązanie złączenia z generacją rozkładu

Zapytania mogą wymagać utworzenia wielu rozkładów i wyznaczenia ich części wspólnej. Przykładem jest zapytanie o przekroczenie norm kilku substancji w tym samym miejscu

lub o przekroczenie normy dla kilku okresów czasu w tym samym miejscu tak, że dla każdego okresu jest tworzony niezależny rozkład. W takim przypadku, jeżeli dany kwadrat ma się znaleźć w odpowiedzi, musi należeć do wszystkich rozkładów. Obserwacja ta pozwala zoptymalizować generację poprzez wprowadzenie kroku sprawdzającego, czy badany kwadrat należy do już wygenerowanych rozkładów. Jeżeli został odrzucony w jednym z tych rozkładów, można pominąć obliczenia służące do jego klasyfikacji i odrzucić go zaraz po jego utworzeniu. Jeżeli część rozkładów istnieje w bazie, należy wykonać ich złączenie, a wyniku użyć do odrzucania kwadratów w trakcie generacji pozostałych rozkładów. Część wspólna będzie miała na ogół mniej kwadratów (zajmowała mniejszy obszar) niż dowolny z łączonych rozkładów. W najgorszym przypadku, gdy zadane przedziały dla wszystkich rozkładów zajmują ten sam obszar, część wspólna będzie im równa. Algorytm zapisany w pseudojęzyku przedstawiono w rozdziale 8.2.

7. Podsumowanie

W artykule przedstawiono model rozkładu przestrzennego, czyli opisu rozmieszczenia zadanej cechy (zmiennej, wielkości) w przestrzeni. Przy założeniu działania w przestrzeni dyskretnej i przetwarzania rozłącznych przedziałów wartości scharakteryzowano zadania wyszukiwania dotyczące rozkładów przestrzennych. Rozkłady i inne przetwarzane obiekty przestrzenne są aproksymowane liniowym drzewem czwórkowym uporządkowanym krzywą N-Peano. Omówiono operacje działające na takiej reprezentacji i oceniono je pod kątem użycia na etapie optymalizacji i wykonania zapytania. Przedstawiono reguły optymalizacji zapytań, dotyczących rozkładów, koncentrując się na operacji wyznaczenia części wspólnej obiektów i optymalizacji generacji rozkładów poprzez jej powiązanie z operacją selekcji. Szczególnie obiecujące wydaje się połączenie wyznaczania części wspólnej z generacją, w trakcie której następuje selekcja.

Dalsze prace będą obejmowały optymalizację dostępu do danych oraz implementację optymalizatora i jego badania eksperymentalne. Wydaje się, że największy wpływ na przyśpieszenie zapytań ma połączenie wyznaczania części wspólnej rozkładów z ich generacją. Przyśpieszenie to będzie zależeć od przetwarzanych danych i warunków w zapytaniu. Jeżeli rozkłady będą przyjmowały w tych samych rejonach wartości z tych samych przedziałów, przyśpieszenie będzie nieznaczne. Jeżeli natomiast zadane w pytaniu przedziały będą się znajdować w innych miejscach, zysk będzie znaczny ze względu na wyeliminowanie czasochłonnych obliczeń wykonywanych w trakcie generacji. Kolejnym problemem jest opracowanie reguły pozwalającej na określenie, w jakiej kolejności należy generować rozkłady, aby jak najszybciej następowała redukcja części wspólnej i przez to ograniczenie obliczeń.

Aproksymacje rozkładów mogą być zapamiętane w relacyjnej bazie danych, obiektowej bazie danych lub systemie plików. Zalety i wady tych podejść, z punktu widzenia optymalizacji dostępu do danych, wymagają dalszych badań.

8. Algorytmy

8.1. Algorytm wyznaczania rozkładu w powiązaniu z selekcją

1. ***Funkcja** WyznRozklSel*
2. *Wyznacz kwadrat reprezentujący cały obszar*
3. ***Jeżeli** kwadrat można przypisać do przedziału*
4. *Przypisz go do przedziału*
5. ***W przeciwnym przypadku***
6. *vbJestRozwinięcie* = PRAWDA
7. *Powtarzaj dopóki* vbJestRozwinięcie *I nie osiągnięto maksymalnego rozwinięcia*
8. vbJestRozwinięcie = RozwińDrzewoSel
9. ***Zwróć** wyznaczony rozkład*

10. ***Funkcja** RozwińDrzewoSel*
11. *vbRozwiniecie = FAŁSZ*
12. ***Dla wszystkich** kwadratów w drzewie*
13. *Jeżeli kwadrat nie jest przypisany do przedziału*
14. *Zastąp kwadrat czteroma mniejszymi*
15. *Dla każdego z nowo utworzonych kwadratów*
16. *Zaznacz kwadrat jako nie przypisany do przedziału*
17. ***Jeżeli** kwadrat nie należy do rozpatrywanego obszaru lub nie może należeć do przedziałów zadanych w warunku zapytania*
18. *Odrzuć go*
19. ***W przeciwnym przypadku jeżeli** kwadrat można przypisać do przedziału lub kwadrat jest elementarny*
20. *Przypisz kwadrat do przedziału*
21. *Zaznacz kwadrat jako rozwinięty*
22. *vbRozwiniecie* = PRAWDA
23. *Zwiększ poziom rozwinięcia*
24. ***Zwróć** vbRozwiniecie*

8.2. Algorytm wyznaczania części wspólnej w powiązaniu z generacją rozkładów

1. ***Funkcja** WyznCzWspGen*
2. ***Jeżeli** któreś z zadanych w zapytaniu rozkładów znajdują się w bazie danych*
3. *Odczytaj je*
4. *Wyznacz ich część wspólną i zapamiętaj w strukturze danych vCzWsp*

5. *W przeciwnym przypadku*
6. *Wybierz rozkład*
7. *Wyznacz wybrany rozkład (funkcja WyznRozklSel ())*
8. *vCzWsp := rozkład wyznaczony w kroku 7*
9. *Dla każdego z pozostałych rozkładów*
10. *Wyznacz rozkład (funkcja WyznRozklZlacz ())*
11. *vCzWsp := Intersection (vCzWsp, rozkład wyznaczony w kroku 10) – wyznacz nową część wspólną (część wspólna z poprzedniego kroku może zawierać kwadraty nie należące do nowego rozkładu)*
12. *Zwróć wyznaczoną część wspólną rozkładów*

13. *Funkcja WyznRozklZlacz*
14. *Wyznacz kwadrat reprezentujący cały obszar*
15. *Jeżeli kwadrat można przypisać do przedziału*
16. *Przypisz go do przedziału*
17. *W przeciwnym przypadku*
18. *vbJestRozwinięcie = PRAWDA*
19. *Powtarzaj dopóki* vbJestRozwinięcie *I nie osiągnięto maksymalnego rozwinięcia*
20. vbJestRozwinięcie = *RozwińDrzewoZlacz*
21. *Zwróć wyznaczony rozkład*

22. *Funkcja RozwińDrzewoZlacz*
23. *vbRozwiniecie = FAŁSZ*
24. *Dla wszystkich kwadratów w drzewie*
25. *Jeżeli kwadrat nie jest przypisany do przedziału*
26. *Zastąp kwadrat czteroma mniejszymi*
27. *Dla każdego z nowo utworzonych kwadratów*
28. *Zaznacz kwadrat jako nie przypisany do przedziału*
29. *Jeżeli kwadrat nie należy do części wspólnej lub nie może należeć do przedziałów zadanych w warunku zapytania*
30. *Odrzuć go*
31. *W przeciwnym przypadku jeżeli kwadrat można przypisać do przedziału lub kwadrat jest elementarny*
32. *Przypisz kwadrat do przedziału*
33. *Zaznacz kwadrat jako rozwinięty*
34. *vbRozwiniecie* = PRAWDA
35. *Zwiększ poziom rozwinięcia*
36. *Zwróć vbRozwiniecie*

LITERATURA

1. Bajerski P.: Efektywność przetwarzania w języku SQL zapytań przestrzennych wykorzystujących aproksymacje obiektów. Zeszyty Naukowe Politechniki Śląskiej, seria Informatyka zeszyt 37, Gliwice 1999.
2. Bajerski P.: Możliwości zapisu w języku SQL zapytań przestrzennych wykorzystujących aproksymacje obiektów. Zeszyty Naukowe Politechniki Śląskiej, seria Informatyka zeszyt 37, Gliwice 1999.
3. Bajerski P.: Sprawozdanie z realizacji projektu badawczego Komitetu Badań Naukowych nr 8T11C 028 12. Wykorzystanie drzew czwórkowych i algebry Peano do prezentacji i przetwarzania rozkładów zanieczyszczenia powietrza. Politechnika Śląska, Gliwice 1998.
4. Breuning M.: Integration of Spatial Information for Geo-Information Systems. Springer, Berlin 1996.
5. Clark I.: Practical Geostatistics, Elsevier Applied Science Publishers Ltd. 1987 (reprinted)
6. Cormen T., Leiserson C., Rivest R.: Wprowadzenie do algorytmów. WNT, Warszawa 1998.
7. Goodall B.: Dictionary of Human Geography. Penguin 1987.
8. Jankowski M.: Elementy grafiki komputerowej. WNT, Warszawa 1990.
9. Laurini R., Francoise M.: Spatial database queries: relational algebra versus computational geometry. Proceedings of the Fourth International Conference on Statistical and Scientific Database Management, Rome, Italy 1988, M. Rafamelli et al. Springer Verlag, Berlin.
10. Laurini R., Thompson D.: Understanding GIS, Academic Press Limited, third printing 1994.
11. Nielson M.: Scattered Data Modelling, "IEEE Computer Graphics & Applications", Vol. 1, 1993.
12. Oosterom P., Vijlbrief T.: The Spatial Location Code in Advances in GIS Research II, edited by Kraak M.J. and Molenaar M., Taylor&Francis Ltd. 1997.
13. Sabin M.: Contouring – the State of the Art, "Fundamental Algorithms for Computer Graphics", Springer-Verlang, Berlin 1985.
14. Samet H.: The design and Analysis of Spatial Data Structures, Addison-Wesley, 1990.
15. Ullman J.: Systemy baz danych. WNT, Warszawa 1988.

Recenzent: Dr hab. inż. Wojciech Mokrzycki

Wpłynęło do Redakcji 14 stycznia 2000 r.

Abstract

The goal of the paper is to characterize queries concerning spatial distributions, describe operations needed to their execution, and present chosen rules of their optimization. The term spatial distribution denotes a set of geographic observations representing the values or behavior of a particular phenomenon or characteristic across many locations on the surface of the earth [7]. Some assumptions about distributions processed were made:

1. only two-dimensional (2D) distributions are taken into account,
2. an approximation of an unknown real distribution is computed using linear local interpolation methods [5, 11, 13] and the numerical values obtained at measuring points,
3. queries concern disjoint intervals of distribution values,
4. computations are carried out in a discrete space with the resolution given by a user,
5. values measured are accurate – data verification and correction should be performed by means of other specialized tools.

Distributions and other spatial objects are approximated by region quadtrees ordered by Peano N curve and stored in Peano relations in relational database.

Peano-tuple algebra [10], which offers some tools to manipulate Peano relations, is described. A feasibility study showed that boolean operations (union, intersection and difference) must be available in query language and geometric operations (translation, rotation, scaling, symmetry, window extraction and simplification) as well as Peano join are used only during query optimization and execution.

Query optimization employs strategies emerging from the nature of spatial distributions and rules adopted from relational databases and from GIS. The optimizer must take into account two goals: execution time and exactness. Some query optimization rules are given, with special attention devoted to spatial objects intersection and combining distribution computation with other operations. The most important rules are:

- merging quadrants belonging to different intervals or disjoint objects before intersection,
- combining spatial distribution creation with selection,
- combining spatial distribution creation with intersection.