STUDIA INFORMATICA 2000
Volume 21 Number 3 (41)

Piotr BAJERSKI
Politechnika Śląska, Instytut Informatyki

PORÓWNANIE EFEKTYWNOŚCI ZŁĄCZENIA OBIEKTÓW POWIERZCHNIOWYCH W REPREZENTACJI WEKTOROWEJ I PŁATOWEJ

Streszczenie. W artykule przedstawiono porównanie czasu wykonania złączenia przestrzennego dostarczającego jako wynik obiekty przestrzenne dla reprezentacji wektorowej i płatowej. Reprezentacja płatowa została zaimplementowana jako drzewo czwórkowe uporządkowane krzywą fraktalną N-Peano. W eksperymentach wykorzystano rozkład zanieczyszczenia powietrza i granice obszarów administracyjnych.

EFFICIENCY COMPARISON OF AREA OBJECTS JOIN IN VECTOR AND TESSELLATED REPRESENTATION

Summary. The paper presents a comparison of efficiency of spatial join with spatial output for vector and tessellated representations. Tessellation was implemented as a quadtree ordered by a Peano N fractal curve. The experiments were carried out using an air pollution distribution and administrative unit borders.

1. Wstęp

Obiekty reprezentujące zjawiska znajdujące się lub wytyczone na, pod lub nad powierzchnią ziemi są nazywane obiektami przestrzennymi. Atrybuty obiektów przestrzennych są dzielone na dwie grupy [9]: atrybuty przestrzenne oraz atrybuty opisowe. Pierwsza grupa reprezentuje położenie obiektów w przestrzeni oraz ich kształt (geometrię), do drugiej należą atrybuty opisujące pozostałe cechy obiektów. W zależności od liczby wymiarów atrybuty przestrzenne są dzielone na: punktowe, liniowe, powierzchniowe i objętościowe. Przy zmniejszaniu skali mapy atrybuty liniowe, powierzchniowe i objętościowe mogą być redu-

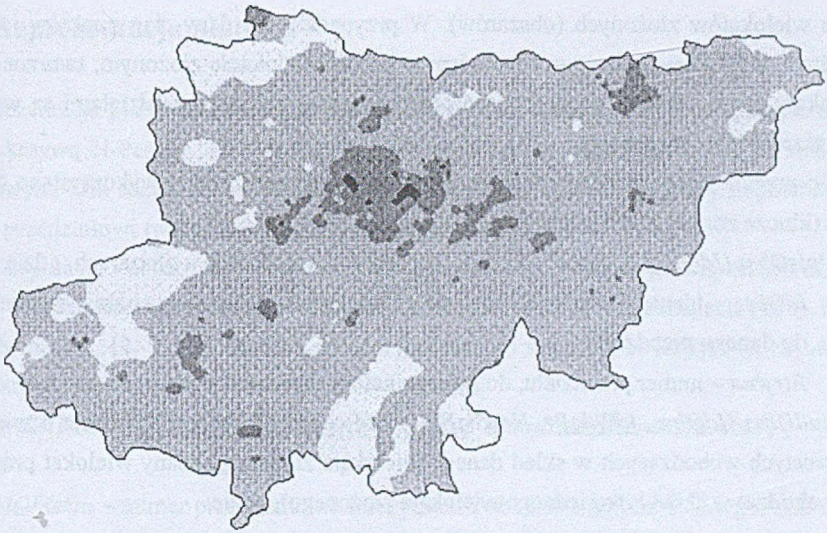
kowane do atrybutów punktowych. W dalszej części artykułu będą rozważane obiekty przestrzenne posiadające jeden atrybut powierzchniowy.

Wyróżnia się następujące reprezentacje danych przestrzennych, [5]:

- wektorową (ang. vector representation) – 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 dany obiekt jest opisywany przez płaty, które pokrywa. Szczególnym przypadkiem jest raster, gdy położenie wszystkich obiektów jest opisane za pomocą jednakowych elementarnych powierzchni (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 równaniami funkcyjnymi. Zastosowanie tej metody w systemach informacji przestrzennej jest ograniczone z powodu złożoności i braku regularności obiektów.

Przedstawione badania miały na celu eksperymentalne porównanie efektywności wykonania złączenia przestrzennego dającego jako wynik obiekty przestrzenne (ang. Spatial Join with Spatial Output), [8], dla dwóch reprezentacji o największym znaczeniu praktycznym: wektorowej i płatowej. Reprezentacja płatowa została zaimplementowana w postaci drzewa czwórkowego. Istotną cechą badanego złączenia jest dostarczanie jako wyniku zbioru obiektów przestrzennych, które mogą służyć do wykonania następnych operacji. Możliwe są również wersje złączenia przestrzennego dające w wyniku pary identyfikatorów obiektów, dla których zachodzi badana relacja przestrzenna (np. sprawdzająca nakładanie się obiektów). W badaniach przyjęto, że jednym z argumentów jest rozkład przestrzenny, a drugim obiekty administracyjne.

Termin *rozkład przestrzenny* (ang. spatial distribution) 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. Rozważania zostaną ograniczone do przypadku, w którym analizowane są przedziały wartości zadanej cechy. Podejście takie prowadzi do dekompozycji analizowanej powierzchni na fragmenty, na obszarze których występują wartości z tego samego przedziału. W badaniach wykorzystano pomiary zanieczyszczenia powietrza wykonane przez Wojewódzką Stację Sanitarno-Epidemiologiczną w Katowicach.



Rys. 1. Przykładowy rozkład zanieczyszczenia powietrza
Fig. 1. An exemplary air pollution distribution

Rozkłady zanieczyszczenia powietrza charakteryzuje mała liczba obiektów powierzchniowych o złożonej budowie (wklęsłe obiekty o skomplikowanym przebiegu granic posiadające dużą liczbę dziur). Charakterystyczne jest duże zróżnicowanie wielkości spójnych obszarów należących do poszczególnych przedziałów wartości. Rysunek 1 przedstawia przykładowy rozkład zanieczyszczenia powietrza.

Zastosowanie zarówno reprezentacji wektorowej, jak i płatowej prowadzi do aproksymacji rozkładu. W reprezentacji wektorowej przebieg granic między przedziałami jest przybliżany linią łamaną. W reprezentacji płatowej, ze względu na skończoną rozdzielczość i metodę generacji rozkładu, fragment przestrzeni może zostać przypisany do danego przedziału, mimo że część tego fragmentu należy do innego przedziału.

2. Reprezentacja wektorowa

W celu opisania rozkładów w reprezentacji wektorowej wprowadzono dwa pojęcia: wielokąt prosty i wielokąt złożony. *Wielokąt prosty* jest wklęsłym lub wypukłym wielokątem bez dziur i węzłów wielokrotnych. *Wielokąt złożony* odpowiada spójnemu obszarowi. Wielokąt złożony składa się z brzegu, który jest wielokątem prostym, i dowolnej liczby dziur, które także są wielokątami prostymi. Wielokąt złożony będzie również nazywany *obszarem*. Dla rozkładu zanieczyszczenia wielokąt złożony reprezentuje spójny obszar pokryty zanieczyszczeniem o wartościach z jednego przedziału. Dany przedział może być reprezentowany przez

wiele wielokątów złożonych (obszarów). W przypadku gdy dany ciąg punktów jest dziurą w jednym wielokącie złożonym i brzegiem w innym wielokącie złożonym, tworzone są dwa wielokąty proste. Punkty wyznaczające przebieg granic między przedziałami są współdzielone przez wielokąty proste.

Do przechowania w bazie rozkładu w reprezentacji wektorowej wykorzystano cztery tablice (klucze zostały podkreślone):

1. *OpisObsz (IdObsz, Atrybut)* – tablica przechowuje informacje o obszarach, gdzie:
   - *IdObsz* – identyfikator wielokąta złożonego aproksymującego spójny obszar należący do danego przedziału,
   - *Atrybut* – numer przedziału, do którego należy dany obszar,
2. *BudObsz (IdObsz, IdWlktPr, NrWlktPr)* – tablica przechowuje informacje o wielokątach prostych wchodzących w skład danego wielokąta złożonego (dany wielokąt prosty może wchodzić w skład tylko jednego wielokąta złożonego), gdzie:
   - *IdObsz* – identyfikator obszaru, którego budowę dany wiersz opisuje,
   - *IdWlktPr* – identyfikator wielokąta prostego,
   - *NrWlktPr* – numer wielokąta prostego w ramach danego obszaru; numer *„1"* oznacza brzeg obszaru, numery większe od *„1"* oznaczają dziury,
3. *BudWlktPr (IdWlktPr, IdPnkt, NrPnkt)* – tablica przechowuje informacje o ciągu punktów tworzących dany wielokąt prosty, gdzie:
   - *IdWlktPr* – identyfikator wielokąta prostego,
   - *IdPnkt* – identyfikator punktu,
   - *NrPnkt* – numer punktu w ramach danego wielokąta prostego,
4. *Punkty (IdPnkt, SzerGeog, DlugGeog)* – tablica przechowuje współrzędne punktów, będących wierzchołkami wielokątów prostych, gdzie:
   - *IdPnkt* – identyfikator punktu,
   - *SzerGeog* – szerokość geograficzna punktu,
   - *DlugGeog* – długość geograficzna punktu.

Obszary administracyjne nie zawierają dziur i można je zapamiętać jako wielokąty proste wykorzystując dwie tablice:

- *GranAdmKont* – tablica przechowuje budowę granic obiektów administracyjnych i ma taką samą strukturę jak tablica *BudWlktPr*,
- *GranAdmPunkty* – tablica przechowuje współrzędne punktów wyznaczających przebieg granic administracyjnych i ma taką samą strukturę jak tablica *Punkty*.

3. Reprezentacja płatowa

Reprezentacja płatowa została zrealizowana w postaci drzewa czwórkowego uporządkowanego krzywą N-Peano, [10, 14, 15]. Drzewo takie będzie nazywane liniowym drzewem czwórkowym. Dla rozkładu przestrzennego przyjęto, że wszystkie kwadraty odpowiadające danemu przedziałowi tworzą jeden obiekt powierzchniowy.

Aproksymację obiektów przestrzennych drzewem czwórkowym uporządkowanym krzywą N-Peano można przechowywać w relacyjnej bazie danych w relacjach Peano o różnych schematach, [2, 9, 10]. W badaniach wykorzystano schemat, w którym są zapamiętywane zakresy kluczy Peano odpowiadające kwadratom:

SchemRelPeano (IdObiektu, KluczPeanoPoczatku, KluczPeanoKonca, Atrybuty),

gdzie:

- *IdObiektu* – numer przedziału lub identyfikator obiektu administracyjnego, do którego aproksymacji dany kwadrat należy,
- *KluczPeanoPoczatku* – klucz Peano lewego dolnego kwadratu elementarnego (minimalna wartość klucza Peano odpowiadająca kwadratowi),
- *KluczPeanoKonca* – klucz Peano prawego górnego kwadratu elementarnego (maksymalna wartość klucza Peano odpowiadająca kwadratowi),
- *Atrybuty* – atrybuty opisowe, związane z danym fragmentem obiektu o identyfikatorze *IdObiektu*.

Każda krotka relacji opisuje jeden z kwadratów, tworzących aproksymację powierzchni zajmowanej przez obiekt.

4. Wyznaczanie części wspólnej

W rozdziale omówiono sposób wyznaczania części wspólnej dwóch wielokątów złożonych w reprezentacji wektorowej i w przedstawionej wersji reprezentacji płatowej.

4.1. Reprezentacja wektorowa

Do wyznaczania części wspólnej wielokątów złożonych zapisanych w postaci wektorowej wykorzystano bibliotekę *PolyBoolean*, [11]. Umożliwia ona wyznaczenie sumy, części wspólnej, różnicy i różnicy symetrycznej dwóch wielokątów złożonych na płaszczyźnie. Ponieważ złożoność algorytmu nie pozwala na jego dokładne przedstawienie i jest on dostępny w Internecie, przedstawiono tylko jego główne kroki:

1. wyznaczenie przecięcia wielokątów prostych,

2. przydzielenie etykiet do krawędzi i wielokątów,
3. utworzenie wielokątów prostych uwzględniających punkty przecięcia; autor algorytmu podkreśla, że zasadniczą ideą tego kroku jest jego wykonanie na podstawie etykiet, a nie współrzędnych,
4. utworzenie wielokątów złożonych, będących wynikiem wykonanej operacji.

Złożoność asymptotyczna algorytmu jest równa:

$$O(n \log^* n + k + z \log n), \quad (1)$$

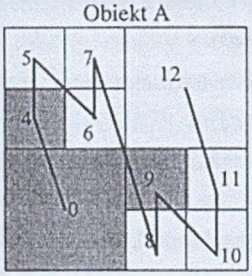
gdzie:

n – liczba krawędzi we wszystkich wielokątach prostych,

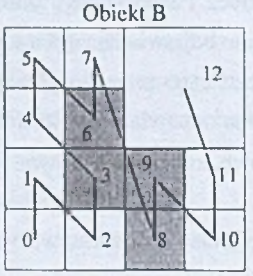
z – liczba wielokątów prostych,

k – liczba znalezionych przecięć krawędzi, w najgorszym przypadku $k = O(n^2)$.

4.2. Reprezentacja płatowa



IdOb	KPmin	KPmax
A	0	3
A	4	4
A	9	9



IdOb	KPmin	KPmax
B	3	3
B	6	6
B	8	8
B	9	9

IdOb	KPmin	KPmax
A and B	3	3
A and B	9	9

Rys. 2. Obiekty *A* i *B* oraz ich część wspólna
Fig. 2. Objects *A* and *B* together with their intersection

W przypadku aproksymacji drzewem czwórkowym uporządkowanym krzywą N-Peano wyznaczenie części wspólnej dwóch wielokątów złożonych sprowadza się do znalezienia nakładających się fragmentów odcinków na krzywej N-Peano (rys. 2). Złożoność asymptotyczna jest równa, [4]:

$$O(n_1 + n_2), \quad (2)$$

gdzie:

n_1 – liczba kwadratów aproksymacji obiektu 1,

n_2 – liczba kwadratów aproksymacji obiektu 2.

5. Badania eksperymentalne

Celem badań było eksperymentalne porównanie czasu wyznaczania części wspólnej obiektów powierzchniowych w reprezentacji wektorowej i płatowej. Dla każdej z reprezentacji badano czas potrzebny do wykonania poszczególnych etapów zapytania, takich jak: odczyt danych, przygotowanie danych do przetwarzania i wyznaczanie części wspólnej.

W badaniach przyjęto, że rozkład przestrzenny został wcześniej utworzony i zapamiętany w bazie. Pomiary pobierania danych z bazy nie uwzględniają czasu potrzebnego na inicjację oprogramowania oraz struktur danych koniecznych do komunikacji z serwerem. Przyjęto, że czynności te są normalnie wykonywane podczas startu programu i mogą mieć wpływ tylko na pierwsze zapytanie. W tabelach przedstawiono wykonanie zapytań po starcie komputera, aby wyeliminować wpływ buforowania danych przez serwer. Zastosowany pomiar czasu miał rozdzielczość 10 ms. Czasy operacji w pamięci zostały wyznaczone poprzez ich wielokrotne wykonanie. Pomiar czasu wykonywania operacji pobrania danych z bazy, trwających mniej niż 0,2 s, dawał duże błędy. Ponieważ nie można było wykonać tych operacji w pętli z uwagi na buforowanie danych przez serwer, czasy zostały oszacowane na podstawie kilku pomiarów i zaokrąglone do 0,1 s.

Eksperymenty były prowadzone na komputerze *Pentium II* z procesorem 266 MHz i pamięcią operacyjną 64 MB. Aplikacja i serwer bazy danych (*MS Access*) znajdowały się na tym samym komputerze i komunikowały się za pomocą mechanizmu *Data Access Objects* (*DAO*). Aplikacja została napisana w języku *C++* w środowisku *Visual C++ 6.0*. Do utworzenia interfejsu użytkownika i odwołań do mechanizmu *DAO* wykorzystana została biblioteka *Microsoft Foundation Class* (*MFC*). Biblioteka *PolyBoolean* posługuje się współrzędnymi zapisanymi jako liczby całkowite, co spowodowało konieczność konwersji współrzędnych geograficznych, zapisanych w bazie danych jako liczby zmiennoprzecinkowe.

5.1. Charakterystyka wykorzystywanych danych

W badaniach wykorzystano dwa zbiory obiektów przestrzennych: przykładowy dwuwymiarowy rozkład zanieczyszczenia powietrza i obszary administracyjne dawnego województwa katowickiego (przed zmianą podziału terytorialnego w roku 1998). Użyty rozkład został utworzony na podstawie pomiaru średniorocznego stężenia pyłu zawieszonego w powietrzu w roku 1989. Zadano cztery przedziały, których granice były wielokrotnościami dopuszczalnego stężenia tego zanieczyszczenia. Rozkład ten zawierał cechy charakterystyczne dla analizowanych danych: mało obiektów przestrzennych, skomplikowany przebieg granic między przedziałami, duża liczba dziur i jeden dominujący przedział. Reprezentację płatową rozkładu wygenerowano dla rozdzielczości 1024 piksele metodą zapewniającą kompromis

między dokładnością i szybkością, oznaczaną przez *M3*, [3]. Wykorzystuje ona interpolację lokalną, [12, 13]. Decyzja o zakwalifikowaniu kwadratu do przedziału bądź jego dalszym podziale jest podejmowana na podstawie wartości wyznaczonych dla środka kwadratu i środków czterech mniejszych kwadratów. Przedziały 0, 1, 2 i 3 zajmowały odpowiednio 14,7%; 80,1%; 5% i 0,2% obszaru województwa.

Wektorowa aproksymacja rozkładu została utworzona poprzez wektoryzację rozkładu rastrowego. Rozkład rastrowy został wyznaczony poprzez obliczenie wartości dla każdego piksela rozważanego obszaru. Podczas obliczeń użyto tej samej rozdzielczości i tej samej metody interpolacji co przy tworzeniu rozkładu w reprezentacji płatowej.

Jako kryterium oceny dokładności aproksymacji przyjęto liczbę źle zakwalifikowanych pikseli w stosunku do rozkładu rastrowego. Dla obydwu metod reprezentacji liczony w ten sposób błąd wynosił kilka procent.

Aproksymacja obiektów administracyjnych obejmowała 1 województwo, 84 gminy i 13 największych miast. Atrybuty nieprzestrzenne obiektów administracyjnych były przechowywane w tablicy wspólnej dla obydwu reprezentacji.

Tabela 1

Liczba wierszy tablic opisujących rozkład w reprezentacji wektorowej

Tablica	Łączna liczba pozycji	Liczba pozycji dla przedziału			
		0	1	2	3
OpisObsz	80	36	4	35	5
BudObsz	144	37	58	44	5
BudWlktPr	2477	572	1259	618	28

Tabela 1 przedstawia liczbę wierszy tablic opisujących rozkład w reprezentacji wektorowej. Do opisu przebiegu granic przedziałów wykorzystano 1456 punktów. Przebieg granic 98 wyróżnionych obiektów administracyjnych został wyznaczony 2070 punktami, z czego 1834 punktów użyto do opisu przebiegu granic gmin. Liczba wierszy opisujących budowę granic wynosiła 4067, z czego na gminy przypadły 3353 wiersze.

Dla reprezentacji płatowej aproksymacja rozkładu składała się z 6373 kwadratów, z czego na przedziały 0, 1, 2 i 3 przypadało odpowiednio 1149, 3281, 1826 i 117 kwadratów. Aproksymacja obiektów administracyjnych składała się z 26095 kwadratów, z czego na aproksymację województwa przypadało 1535 kwadratów, na aproksymacje gmin 19872 kwadraty i na aproksymacje miast 4688 kwadratów. Aproksymacje obiektów na tym samym poziomie hierarchii administracyjnej były rozłączne.

5.2. Zapytania

W badaniach eksperymentalnych użyto czterech zapytań. Każde z tych zapytań dotyczy innych przedziałów i wykorzystuje inny zakres danych omówionych w rozdziale 5.1.

1. Podaj nazwy gmin, na terenie których występuje zanieczyszczenie z przedziału 3. Podaj również obszary gmin należące do tego przedziału.
   Do przedziału 3 należy 5 plam zanieczyszczenia o małej powierzchni, w sumie ok. 0,2% powierzchni województwa. Plamy te nie zawierają dziur (są wielokątami prostymi). Odpowiedź obejmuje 9 z 84 gmin.
2. Podaj nazwy gmin, na terenie których występuje zanieczyszczenie z przedziałów 2 lub 3. Podaj również obszary gmin należące do tych przedziałów.
   Do przedziałów 2 i 3 należy 40 plam, zajmujących w sumie ok. 5% powierzchni województwa. Plamy te zawierają małą liczbę dziur. Odpowiedź obejmuje 39 z 84 gmin.
3. Podaj nazwy gmin, na terenie których występuje zanieczyszczenie z przedziałów 0 lub 1. Podaj również obszary gmin należące do tych przedziałów.
   Przedziały 0 i 1 pokrywają ok. 95% powierzchni województwa. Obszary te zawierają dużą liczbę dziur. Odpowiedź obejmuje wszystkie gminy.
4. Podaj nazwy gmin, na terenie których występuje zanieczyszczenie z przedziału 1. Podaj również obszary gmin należące do tego przedziału.
   Przedział 1 pokrywa ok. 80% województwa, z czego większość przypada na jedną plamę o skomplikowanych granicach z wieloma dziurami. Odpowiedź obejmuje 83 z 84 gmin.

5.3. Wyniki zapytań dla reprezentacji wektorowej

Dla reprezentacji wektorowej w pierwszej kolejności zbadano celowość wykorzystania minimalnych prostokątów ograniczających (ang. Minimal Bounding Rectangle – MBR). Jest to często stosowana w zapytaniach przestrzennych technika polegająca na wyznaczaniu dla każdego obiektu najmniejszego prostokąta obejmującego ten obiekt. Na ogół przyjmuje się, że boki prostokąta są równoległe do osi, co upraszcza sprawdzenie nakładania się MBR. Wykorzystanie MBR pozwala na wykonanie złączenia przestrzennego w dwóch etapach. W pierwszym etapie znajduje się pary obiektów, których MBR się przecinają. W drugim etapie wykonuje się dokładne, bardziej czasochłonne obliczenia, aby określić, które pary rzeczywiście się przecinają, i wyznaczyć część wspólną. Tabele 2 i 3 przedstawiają czas wykonania poszczególnych etapów zapytań w wersjach bez wykorzystania MBR i z wykorzystaniem MBR.

Tabela 2

Czas wykonania zapytań bez wykorzystania MBR

Lp.	Typ pomiaru	Jedn.	Pyt. 1	Pyt. 2	Pyt. 3	Pyt. 4
1	Odczyt danych administracyjnych	s	0,2	0,2	0,2	0,2
2	Odczyt aproksymacji gmin	s	1,8	1,8	1,8	1,8
3	Odczyt aproksymacji rozkładu	s	0,8	1,8	2,5	1,7
4	Łączny czas odczytu danych (1 + 2 + 3)	s	2,8	3,8	4,5	3,7
5	Konwersja danych	ms	43	57	81	78
6	Wyznaczenie części wspólnej	s	3,27	30,70	47,13	17,69
7	Obliczanie pola	ms	4	95	416	176
8	Łączny czas operacji (5 + 6 + 7)	s	3,32	30,85	47,63	17,94
9	Łączny czas (4 + 8)	s	6,1	34,7	52,1	21,6

W kolejnych wykonaniach zapytań czas pobrania danych administracyjnych był krótszy niż 10 ms. Dla zapytań nie wykorzystujących MBR kolejne odczyty aproksymacji gmin trwały 1,8 s, a aproksymacji rozkładów dla pytań nr 1, 2, 3 i 4 odpowiednio 0,5; 1,3; 2,1 i 1,2 s.

Tabela 3

Czas wykonania zapytań z wykorzystaniem MBR

Lp.	Typ pomiaru	Jedn.	Pyt. 1	Pyt. 2	Pyt. 3	Pyt. 4
1	Odczyt danych administracyjnych	s	0,2	0,2	0,2	0,2
2	Odczyt aproksymacji gmin	s	0,6	1,3	1,8	1,8
3	Odczyt aproksymacji rozkładu	s	0,8	1,8	2,5	1,7
4	Łączny czas odczytu danych (1 + 2 + 3)	s	1,6	3,3	4,5	3,7
5	Konwersja danych	ms	22	40	71	67
6	Wyznaczenie części wspólnej	s	0,08	1,53	17,23	15,77
7	Obliczanie pola	ms	4	95	416	176
8	Łączny czas operacji (5 + 6 + 7)	s	0,11	1,67	17,72	16,01
9	Łączny czas (4 + 8)	s	1,7	5	22,2	19,7

W kolejnych wykonaniach zapytań wykorzystujących MBR czas pobierania aproksymacji gmin dla pytań nr 1, 2, 3 i 4 zmniejszył się odpowiednio do 0,6; 1,3; 1,8 i 1,8 s, a aproksymacji rozkładów odpowiednio do 0,4; 1,3; 2,1 i 1,2 s.

Tabela 4

Czas wykonania zapytań z wykorzystaniem MBR i scalaniem obszarów

Lp.	Typ pomiaru	Jedn.	Pyt. 2	Pyt. 3
1	Wyznaczenie części wspólnej	s	1,44	14,56
2	Obliczanie pola	ms	74	278
3	Łączny czas operacji (poz. 5 tab. 5 + 1 + 2)	s	1,55	14,91
4	Łączny czas (poz. 4 tab. 5 + 3)	s	4,9	19,4

Jeżeli dany obszar zawiera dziurę, będącą obszarem, który również należy do przetwarzanego zbioru, to można scalić te obszary. Wyniki przedstawiono w tabeli 4. Ze względu na brak danych topologicznych nie badano scalania przyległych obszarów.

5.4. Wyniki zapytań dla reprezentacji płatowej

Tabela 5 przedstawia czas wykonania zapytań dla reprezentacji płatowej, w rozbiciu na poszczególne etapy. Ponieważ zarówno obszary gmin, jak i przedziałów zanieczyszczenia są wzajemnie rozłączne, dokonano ich scalenia (wiersz 5), otrzymując mniejszą złożoność operacji wyznaczania części wspólnej, [4].

Tabela 5

Czas wykonania zapytań dla danych przechowywanych w relacji Peano

Lp.	Typ pomiaru	Jedn.	Pyt. 1	Pyt. 2	Pyt. 3	Pyt. 4
1	Odczyt danych administracyjnych	s	0,2	0,2	0,2	0,2
2	Odczyt aproksymacji gmin	s	3,7	3,7	3,7	3,7
3	Odczyt aproksymacji rozkładu	s	0,1	0,4	0,8	0,6
4	Łączny czas odczytu danych (1 + 2 + 3)	s	4,0	4,3	4,7	4,5
5	Sortowanie i scalanie kwadratów	ms	27,8	28,7	29,8	27,8
6	Wyznaczenie części wspólnej i pola	ms	1,7	5,0	59,7	48,2
7	Łączny czas operacji (5 + 6)	ms	29,5	33,7	89,5	76,0
8	Łączny czas (4 + 7)	s	<4,1	<4,4	<4,8	<4,6

W kolejnych wykonaniach zapytań czas pobrania aproksymacji gmin zmalał do 3,4 s, a aproksymacji rozkładów dla pytań nr 1, 2, 3 i 4 odpowiednio do 0,1; 0,3; 0,7 i 0,4 s.

Z porównania wartości w wierszach 4 i 7 tabeli 5 wynika, że czas wykonania zapytań zależy głównie od czasu przesłania danych z serwera do aplikacji. Czas ten jest proporcjonalny do liczby pobieranych wierszy tablicy i można go skrócić zapamiętując w jednym wierszu ciąg kluczy Peano. Taki sposób przechowywania danych nazwano *aproksymacją z upakowanymi kluczami Peano*. W zastosowanym rozwiązaniu ciąg kluczy był zapamiętywany jako *dane dwójkowe długie* w polu typu *Obiekt OLE*. Razem zapamiętywano kwadraty, między którymi odległość na krzywej Peano nie przekraczała zadanej wartości. Aproksymacje z upakowanymi kluczami przechowywano w tablicach o schemacie:

SchemRelPeanoUpak (IdOb, KPmin, KPmax, LKwdr, CiagKP),

gdzie:

- *IdOb* – identyfikator obiektu administracyjnego lub numer przedziału opisywanego daną grupą kluczy Peano,
- *KPmin* – minimalna wartość klucza Peano w grupie,
- *KPmax* – maksymalna wartość klucza Peano w grupie,
- *LKwdr* – liczba kwadratów w grupie,
- *CiagKP* – ciąg kluczy Peano odpowiadających kwadratom należącym do grupy.

W celu zmniejszenia zapotrzebowania na pamięć zakres klucza Peano dla kwadratu zapamiętywano jako logarytm dwójkowy długości jego boku (długość ta jest całkowitą potęgą liczby dwa), a odległość między kwadratami jako przyrost klucza Peano. Wartości te były

kodowane jako ciąg bajtów. Siedem młodszych bitów służyło do zapamiętania liczby, a najstarszy oznaczał, czy liczba jest kontynuowana w następnym bajcie.

W tabeli 6 przedstawiono czas wykonania zapytań wykorzystujących aproksymacje z upakowanymi kluczami Peano. Jako granicę odległości między kwadratami przyjęto odległość równą 8192 kwadratom elementarnym. Dla danych administracyjnych utworzono 19 grup dla aproksymacji obszaru województwa, 199 grup dla aproksymacji obszarów gmin i 12 grup dla aproksymacji obszarów miast. Dla aproksymacji rozkładu dla przedziałów 0, 1, 2 i 3 utworzono odpowiednio 19, 13, 16 i 3 grupy.

Tabela 6

Czas wykonania zapytań dla danych przechowywanych w spakowanej relacji Peano

Lp.	Typ pomiaru	Jedn.	Pyt. 1	Pyt. 2	Pyt. 3	Pyt. 4
1	Odczyt danych administracyjnych	s	0,2	0,2	0,2	0,2
2	Odczyt aproksymacji gmin	s	0,2	0,2	0,2	0,2
3	Odczyt aproksymacji rozkładu	s	0,1	0,1	0,2	0,2
4	Łączny czas odczytu danych (1 + 2 + 3)	s	0,5	0,5	0,6	0,6
5	Czas rozpakowywania	ms	2,2	19,4	56,0	55,0
6	Sortowanie i scalanie kwadratów	ms	1,8	11,2	28,9	27,5
7	Wyznaczenie części wspólnej i pola	ms	0,7	4,7	59,7	46,7
8	Łączny czas operacji (5 + 6 + 7)	ms	4,7	35,3	144,6	129,2
9	Łączny czas (4 + 8)	s	0,5	<0,6	<0,8	<0,8

W kolejnych wykonaniach zapytań czasy pobierania danych spadły poniżej 0,1 s.

5.5. Omówienie wyników

W tabeli 7 przedstawiono wpływ zastosowania MBR na czas wykonania zapytań w reprezentacji wektorowej. Istotny wpływ MBR widać dla zapytań, w których wybrane przedziały nie zajmują znaczącej części analizowanego obszaru (pytania 1 i 2). Wyraźnie większe przyśpieszenie występuje w obliczeniach niż w pobieraniu danych.

Tabela 7

Wpływ zastosowania MBR na czas wykonania zapytań w reprezentacji wektorowej

Stosunek czasów (bez MBR / MBR)	Pyt. 1	Pyt. 2	Pyt. 3	Pyt. 4
wykonania zapytań (poz. 9 tab. 2 / poz. 9 tab. 3)	3,6	6,9	2,3	1,1
wykonania odczytu danych (poz. 4 tab. 2 / poz. 4 tab. 3)	1,8	1,2	1,0	1,0
operacji (poz. 8 tab. 2 / poz. 8 tab. 3)	33,2	18,1	2,7	1,1
wyznaczenia części wspólnej (poz. 7 tab. 2 / poz. 7 tab. 3)	40,9	20,1	2,7	1,1

Przy odczycie danych MBR miał wpływ tylko na odczyt granic gmin, ponieważ rozkład był utworzony dla województwa i wobec braku warunków na gminy dla obiektu reprezentującego plamę zanieczyszczenia zawsze istniała gmina przecinająca jego MBR. Fakt, że większa część rozpatrywanego rozkładu przestrzennego może należeć do jednego obiektu, ogranicza możliwość użycia MBR i metod indeksacji przestrzennej, takich jak R-drzewa, [6]. Sca-

lanie obszarów w reprezentacji wektorowej przyśpieszyło wyznaczenie części wspólnej dla zapytania 2 o 1,1; a dla pytania 3 o 1,2.

Z danych przedstawionych w tabeli 5 wynika, że przy przetwarzaniu zapytań wykorzystujących reprezentację płatową dominującą operacją jest odczyt aproksymacji z bazy. Mała różnica między pierwszym pobraniem a następnymi, wykorzystującymi dane znajdujące się w buforze (na korzystanie z bufora wskazuje brak dostępu do dysku), świadczy o tym, że na etapie pobierania danych większość czasu jest poświęcana na ich przesłanie między serwerem i aplikacją. Podczas odczytu danych większość czasu jest poświęcana na pobranie aproksymacji obiektów administracyjnych. Wynika to z dwóch czynników. Po pierwsze, aproksymacja obszaru gmin zawiera przeszło 3-krotnie więcej kwadratów niż aproksymacja rozkładu. Po drugie, w każdym zapytaniu odczytywane są aproksymacje wszystkich gmin oraz fragmenty rozkładu odpowiadające wybranym przedziałom. Około 8-krotną redukcję czasu pobierania danych uzyskano wykorzystując aproksymacją z upakowanymi kluczami Peano (tabela 6). Podejście takie uniemożliwia jednak formułowanie zapytań bezpośrednio w języku SQL, [1, 2].

Tabela 8

Stosunek czasów wykonania zapytań dla reprezentacji wektorowej i płatowej

Stosunek czasów (reprezentacja wektorowa / reprezentacja płatowa)	bez grupowania KP				z grupowaniem KP			
	P1	P2	P3	P4	P1	P2	P3	P4
wykonania zapytań (poz. 9 tab. 3 lub poz. 4 tab. 4 / poz. 8 tab. 5 i poz. 9 tab. 6)	0,4	1,1	4,0	4,3	3,4	8,2	24,3	24,6
odczytu danych (poz. 4 tab. 3 / poz. 4 tab. 5 i poz. 4 tab. 6)	0,4	0,8	1,0	0,8	3,2	6,6	7,5	6,2
operacji (poz. 8 tab. 3 lub poz. 3 tab. 4 / poz. 7 tab. 5 i poz. 8 tab. 6)	3,7	46,0	167	211	23,4	43,9	103	124
wyzn. części wspólnej (poz. 6 tab. 3 lub poz. 1 tab. 4 / poz. 6 tab. 5 i poz. 7 tab. 6)	47,1	288	244	327	114	306	244	338

Tabela 8 przedstawia stosunek czasu wykonania poszczególnych operacji dla reprezentacji wektorowej i płatowej. Dla reprezentacji wektorowej wykorzystano najszybsze wykonanie (MBR i scalanie obszarów) i zestawiono je z wykonaniem dla reprezentacji płatowej bez grupowania i z grupowaniem kluczy Peano. Uwagę zwraca znacznie szybsze wyznaczenie części wspólnej w reprezentacji płatowej. Związane jest to z zastąpieniem czasochłonnej geometrii obliczeniowej prostym algorytmem o złożoności liniowej wykorzystującym tylko operacje porównania i odejmowania liczb całkowitych. W reprezentacji płatowej narzut czasowy na przygotowanie danych (sortowanie i rozpakowanie) jest porównywalny z wyznaczeniem części wspólnej. W reprezentacji wektorowej dla zapytań dotyczących małej liczby stosunkowo prostych obiektów dominujący jest czas odczytu danych (pytania 1 i 2). Jeżeli zapytanie dotyczy dominującego przedziału (pytania 3 i 4), czas wyznaczania części wspólnej znacznie przekracza czas pobierania danych.

Odczyt aproksymacji obiektów administracyjnych w reprezentacji płatowej trwa ok. 2-krotnie dłużej niż w reprezentacji wektorowej. Z kolei odczyt rozkładu w reprezentacji płatowej jest wykonywany kilkakrotnie szybciej niż w reprezentacji wektorowej. Można wskazać dwie przyczyny takich wyników. Po pierwsze, w rozkładzie przestrzennym występują duże obszary, które można efektywnie kodować drzewem czwórkowym. Aproksymacja obszarów administracyjne wymaga większego rozbicia drzewa – w rozważanym przypadku aproksymacja obszarów gmin wymagała przeszło 3-krotnie więcej kwadratów niż aproksymacja rozkładu na tym samym obszarze. Po drugie, w postaci wektorowej łatwiej jest reprezentować granice administracyjne niż "poszarpany" przebieg granic między przedziałami zanieczyszczenia.

Pamięć zajmowana przez dane w reprezentacji wektorowej była przeznaczona głównie na tablicę punktów (jedna pozycja składała się z identyfikatora punktu i dwóch współrzędnych zmiennoprzecinkowych, co wymagało 4 B + 2 * 8 B = 20 B) i wskaźniki na punkty tworzące wielokąty proste (4B). Dla przedstawionych danych wymagało ok. 100 kB pamięci, z czego ok. 40 kB przypadło na rozkład. W przypadku reprezentacji płatowej utworzono tablice struktur opisujących kwadraty aproksymacji. Struktura ta odpowiadała schematowi relacji *SchemRelPeano*. Identyfikator obiektu i klucze Peano zostały zapamiętane jako liczby czterobajtowe. Dla przedstawionych danych wymagało to ok. 400 kB pamięci, z czego ok. 80 kB przypadło na rozkład. Reprezentacja wektorowa danych administracyjnych wymagała przeszło 5-krotnie mniej pamięci, a reprezentacja rozkładu ok. 2-krotnie mniej pamięci.

6. Podsumowanie

W artykule przedstawiono porównanie efektywności złączenia obiektów powierzchniowych w reprezentacji wektorowej i płatowej. W badaniach użyto przykładowego rozkładu zanieczyszczenia powietrza utworzonego w rozdzielczości 1024 piksele, wybranej tak, aby całą mapę można było zobaczyć na przeciętnym monitorze. Dalszych badań wymaga związek między rozdzielczością, w której został utworzony rozkład, a czasem przetwarzania zapytań.

Wydaje się, że przedstawiona reprezentacja płatowa jest naturalnie predestynowana do reprezentacji i przetwarzania rozkładów przestrzennych ze względu na efektywne kodowanie obiektów powierzchniowych i prostotę obliczeń. Podstawowym problemem tej reprezentacji jest czas dostępu do wierszy tablicy przechowującej aproksymacje obiektów. Sytuacja uległaby zmianie, gdyby operacja złączenia została wbudowana w serwer bazy danych. Ponieważ krotki mają prostą budowę i zajmują mało miejsca, optymalnym rozwiązaniem byłaby tablica zbudowana w oparciu o B-drzewo. W takim przypadku można by od razu odczytać posortowane dane i wykonać złączenie bezpośrednio na stronach danych.

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. Bajerski P.: Wybrane zagadnienia optymalizacji zapytań przestrzennych wykorzystujących aproksymacje obiektów. Zeszyty Naukowe Politechniki Śląskiej, seria Informatyka (złożone w redakcji).
5. Breuning M.: Integration of Spatial Information for Geo-Information Systems. Springer 1996.
6. Brinkhoff T., Kriegel H., Seeger B.: Efficient Processing of Spatial Joins Using R-trees. SIGMOD RECORD vol.22, No 2, June 1993.
7. Goodall B.: Dictionary of Human Geography. Penguin 1987.
8. Hoel E., Samet H.: Benchmarking Spatial Join Operations with Spatial Output. Proceedings of the 21st VLDB Conference Zurich, Switzerland 1995.
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. Leonov M., Nikitin A.: A Closed Set of Algorithms for Performing Set .Operations on Polygonal Regions in the Plane, http://members.xoom.com/msleonov/pbpaper.html (tłumaczenie M. Leonova z języka rosyjskiego publikacji zawartej w *A. P. Ershov Institute of Informatics Systems*, Preprint 46, 1997).
12. Nielson M.: Scattered Data Modelling, "IEEE Computer Graphics & Applications", Vol. 1, 1993.
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. Samet H.: The Quad-tree and related hierarchical data structures, ACM Computing Surveys, 1984, 16(2).

Recenzent: Dr hab. inż. Wojciech Mokrzycki

Wpłynęło do Redakcji 7 stycznia 2000 r.

Abstract

The paper presents a comparison of efficiency of spatial join with spatial output for vector and tessellated representations. Experiments were carried out using an exemplary two-dimensional distribution of ambient air pollution and administrative unit borders. The distribution chosen was a typical spatial distribution approximated by a relatively small set of areal objects with complex borders, many holes and one dominating interval (fig. 1).

In the tessellated representation the distribution and administrative unit borders were approximated by region quadtrees ordered by Peano N curve and stored in Peano relations in relational database. In such a representation the intersection of two areal objects can be computed by finding overlapping line segments on Peano N curve (fig. 2). Operations on vector representation were performed using *PolyBoolean* library, [11].

The most time consuming operation in the tessellated representation based on Peano relation is data retrieval (table 5). Grouping quadrants according to Peano keys and storing such groups in one tuple speeded up data retrieval about eight times. Such a representation was called compact Peano relation.

The table 8 shows ratio of query execution time in vector representation to query execution time in tessellated representation. The most significant improvement was for computing intersections of objects (more than 100-times faster). It is attributed to the replacement of computational geometry by testing relations between line segments on the Peano curve. Data retrieval in the vector representation was faster than in the tessellated representation and slower than retrieval in compact Peano relation.

According to the research results the presented tessellated representation is intrinsically dedicated to spatial distribution processing due to simple computation and efficient representation of area objects.