

Adam PIÓRKOWSKI

Akademia Górniczo-Hutnicza w Krakowie, Katedra Geoinformatyki
i Informatyki Stosowanej

Artur KRAWCZYK

Akademia Górniczo-Hutnicza w Krakowie, Katedra Ochrony Terenów Górniczych, Geoinformatyki i Geodezji Górniczej

WPLYW GENERALIZACJI OBIEKTÓW NA OPTYMALIZACJĘ ZAPYTAŃ W BAZACH DANYCH PRZESTRZENNYCH¹

Streszczenie. Niniejszy artykuł opisuje zagadnienie generalizacji obiektów przestrzennych w bazach danych przestrzennych. Wskazano przykładowe operacje (suma powierzchni obiektów, powierzchnia sumy obiektów), w których redukcja redundantnych węzłów zdecydowanie przyspiesza zapytanie. Naświetlono zagadnienie generalizacji obiektów geometrycznych. Udowodniono NP-trudność operacji redukcji największej liczby węzłów przy zadanej dokładności.

Słowa kluczowe: bazy danych przestrzennych, generalizacja obiektów, zapytania przestrzenne

THE PROBLEM OF OBJECT GENERALIZATION AND QUERY OPTIMIZATION IN SPATIAL DATABASES

Summary. This article describes the issue of generalization of spatial objects in spatial databases. There are presented examples of operations (the sum of surface objects, surface objects total) in which the reduction of redundant nodes strongly accelerates query. Highlighting the issue of generalization of geometric objects. It has been proven NP-difficulty of the operation the largest reduction in the number of nodes at a given accuracy.

Keywords: spatial databases, object generalization, spatial queries

¹ Praca finansowana w ramach badań statutowych Katedry Geoinformatyki i Informatyki Stosowanej

1. Bazy danych przestrzennych

Współczesne systemy informatyczne coraz częściej wykorzystują informacje przestrzenne. Informacje takie są bowiem niezbędne przy wszelkich rozwiązaniach logistycznych i geograficznych, co jest spowodowane m.in. dynamicznym rozwojem sieci logistycznych czy też budową infrastruktur danych przestrzennych. Cyfrowe mapy przestrzenne są także niezastąpione w administracji. Zapotrzebowanie na składowanie i przetwarzanie takich danych zaowocowało rozszerzeniami przestrzennymi, rozwijanymi we wszystkich znaczących i popularnych systemach zarządzania relacyjnymi bazami danych. Jednym z istotnych kroków tej dziedziny baz danych było zdefiniowanie standardów dla danych przestrzennych: OpenGIS OGC [1, 2] i SQL/MM [3]. Pozwoliło to na zunifikowanie formatów przechowywania danych (WKB – Well-Known Binary), definiowania obiektów (WKT – Well-Known Text) i konstrukcji zapytań w przestrzennym rozszerzeniu języka SQL. Interesującym zagadnieniem jest optymalizacja takich zapytań. Operowanie na podzbiorach obiektów (np. obiektach mieszkalnych czy drogach w aglomeracjach) często wymaga wyniku o przybliżonej wartości, ale w miarę szybkim czasie. W takim przypadku warto jest zwrócić uwagę na problem generalizacji map, który redukując punkty redundantne lub mało istotne, przyspieszy wykonywanie zapytań.

2. Generalizacja obiektów przestrzennych

Pojęcie generalizacja wywodzi się z języka łacińskiego od słowa „generalis”, które jest tłumaczone jako uogólnienie i towarzyszy kartografii praktycznie od początku jej istnienia. Ze względu na fakt, że powierzchnia Ziemi zawiera bardzo dużą ilość informacji, ludzie od dawna zmuszeni zostali do wypracowania takich metod reprezentacji danych, w których nie można uniknąć pewnego stopnia ich generalizacji. Kolejnym etapem stosowania generalizacji jest przetwarzanie z map wielkoskalowych do małoskalowych. W tym wypadku zmieniane są reguły reprezentacji obiektów. Na mapie w skali 1:1000 budynek prezentowany może być z bardzo dużą dokładnością. W przypadku skali 1:25000 budynek staje się pojedynczym symbolem na mapie lub, w przypadku bliskiego sąsiedztwa innych budynków, zamieniany jest na jeden spójny obszar zabudowany.

Obecnie znanych i stosowanych jest bardzo dużo sposobów generalizacji (operatorów generalizacji). McMaster i Shea [4] wyróżniają następujące sposoby generalizacji: uproszczenie, wygładzanie, agregacja, łączenie, scalanie, dekompozycja, selekcja, wzmocnienie, przemieszczenie i powiększenie. W przypadku gdy dysponujemy atrybutami obiektów, można jeszcze

wyróżnić klasyfikację i symbolizację. Wielu autorów, takich jak Brassel i Weibel w 1988 roku [5] oraz McMaster i Shea w 1989 roku [6], podejmowało próbę rozwiązania problemu generalizacji informacji geograficznej, tworząc teoretyczne modele generalizacji. W 1989 roku McMaster i Shea [6] zaproponowali model generalizacji oparty na bazie analizy 3 problemów:

- przyczyny generalizacji,
- warunków generalizacji,
- metody generalizacji.

Również w 1989 roku Ratajski [7] zdefiniował nowe pojęcia dotyczące generalizacji – uogólnienie ilościowe i jakościowe obiektów oraz zjawisk przestrzennych. Pozwoliło to na wprowadzenie pojęć generalizacji ilościowej i generalizacji jakościowej. Generalizacja jakościowa realizowana jest przez zmianę ujęcia zjawiska z bezpośredniego na pośrednie, poprzez zastosowanie symbolizacji lub grupowania. Natomiast generalizacja ilościowa polega na celowej redukcji liczby informacji tworzących obraz danych.

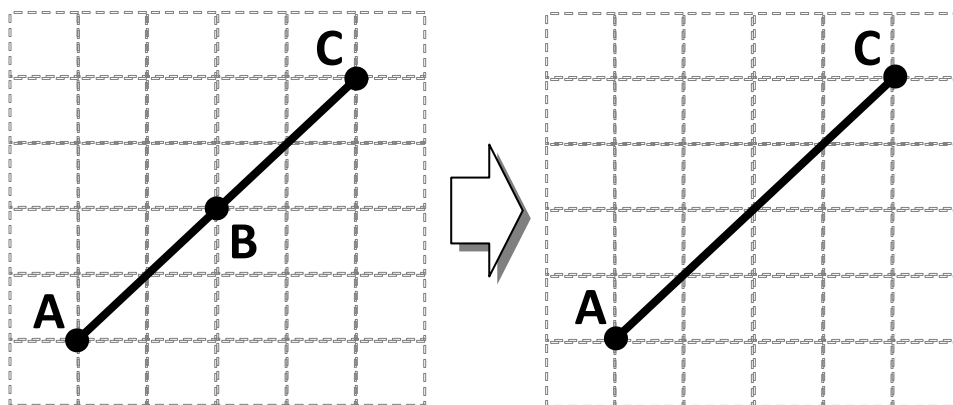
W aspekcie generalizacji uwzględnia się również model danych reprezentujących rzeczywistość geograficzną. Zupełnie inne metody generalizacji stosowane są w modelach wektorowych, a inne w modelach rastrowych.

Najczęściej stosowanym algorytmem generalizacji wektorowych obiektów liniowych jest metoda globalna upraszczania linii Douglasa-Peuckera [8]. Algorytm ten jest oparty na dwóch pojęciach: linii podstawowej i strefie tolerancji. Efekt generalizacji zależy od wielu parametrów i nie zawsze osiągnąć są zadowalające rezultaty.

Celem niniejszej pracy jest przeanalizowanie problemu optymalizacji stosunku czasu realizacji zapytania do bazy danych przestrzennych w relacji do zmian geometrii obiektów będących przedmiotem zapytania. Do realizacji tego zdania wykorzystana zostanie metoda upraszczania geometrii obiektu.

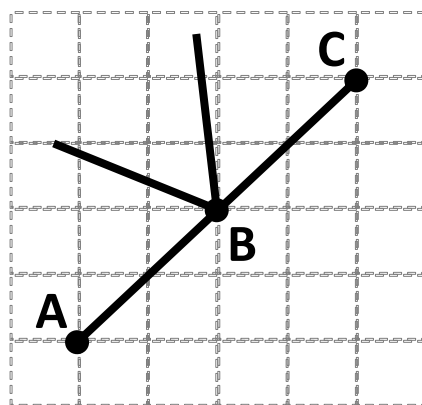
Jedną z metod generalizacji jest upraszczanie. Analizowanym obiektem jest kształt zamknięty wypukły. Ciąg trzech punktów (A, B, C) można uprościć bez konsekwencji dla zmiany kształtu figury przez odrzucenie środkowego (B) i pozostawienie skrajnych (A, C) (rys. 1), jeśli punkty (A, B, C) leżą na jednej prostej, a ów środkowy punkt (B) nie jest węzłem innego ciągu (rys. 2). Inaczej jest w przypadku, gdy punkty nie leżą na jednej prostej. Wówczas usunięcie punktu B powoduje utratę informacji o kształcie figury (rys. 3). W niektórych przypadkach, wskutek pomiarów obarczonych pewnym błędem lub digitalizacji automatycznej, część punktów nie wnosi istotnych danych kształtu. W takim wypadku odrzucenie (redukcja) punktu powinno się odbyć przy spełnieniu pewnego kryterium. Jednym z kryteriów może być ograniczenie redukcji do punktów tworzących z sąsiednimi odpowiedni kąt rozwarty (bliski półpełnemu). Takie kryterium jest niedopuszczalne w przypadku trójkątów o dużych ramionach, gdyż istotnie wpływa na kształt. Skuteczne wydaje się być kryterium, które decyduje o redukcji na podstawie

utruty powierzchni przy takiej operacji. Jeśli zmiany są niewielkie (np. zadany procent powierzchni obiektu), redukcja jest dopuszczalna, a utrata precyzji danych o kształcie (takich jak powierzchnia oraz prawdopodobieństwo wystąpienia nieprawidłowej oceny zależności między obiektami) jest pod kontrolą.



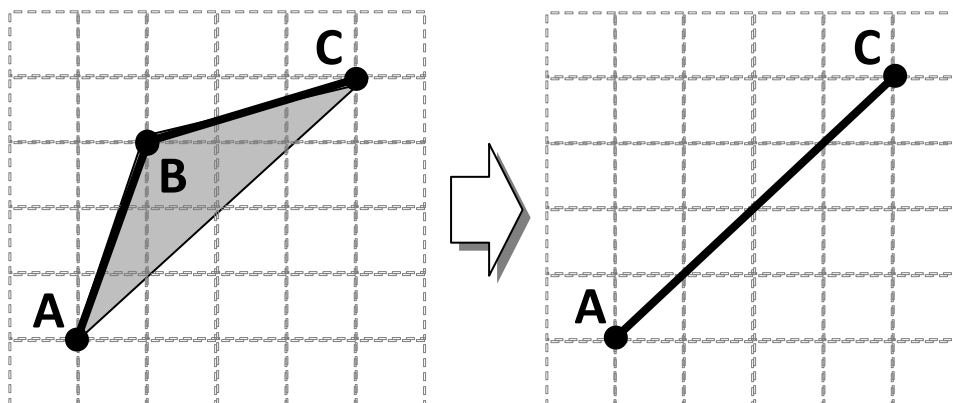
Rys. 1. Redukcja punktu na prostej

Fig. 1. Reduction of point on a line



Rys. 2. Węzeł

Fig. 2. A node



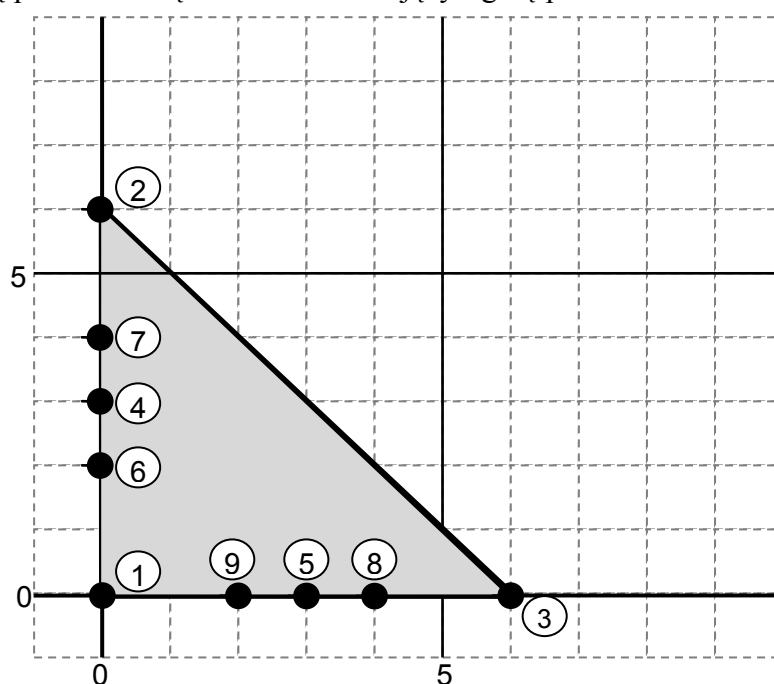
Rys. 3. Redukcja stratna

Fig. 3. Lossy reduction

3. Wpływ generalizacji obiektów na czas zapytań przestrzennych

Wszelkie zapytania przestrzenne, działające na obiektach geometrycznych, przetwarzają w realnych zastosowaniach dużą ilość danych. Analiza każdego obiektu wiąże się z odczytaniem jego reprezentacji w postaci Well-Known Binary (WKB). Czas takiego przetwarzania może być proporcjonalny w stosunku do liczby wierzchołków obiektu. Przykładowo, wyliczanie pola powierzchni dowolnego wielokąta odbywa się przez podział tej figury na składowe trójkąty, których pola wylicza się ze wzoru na powierzchnię trójkąta, a następnie sumuje się je.

Warto sprawdzić, jaki wpływ ma generalizacja na czasy wykonywania zapytań. W tym celu sporządzono model obiektu o zarysie trójkąta. Do owego modelu stopniowo dodano na bokach kolejne redundantne punkty, tak aby nowa figura miała więcej wierzchołków, ale zachowała tę samą powierzchnię. Schemat ilustrujący figurę przedstawiono na rys. 4.



Rys. 4. Przykładowy obiekt geometryczny

Fig. 4. An example of spatial object

Dla tego obiektu dane są definiowane zgodnie ze standardem OGC SQL, frazą:

```
-- trojkat, 3 punkty
GeomFromText('POLYGON((0 0, 0 6, 6 0, 0 0))')

-- trojkat, 4 punkty,
GeomFromText('POLYGON((0 0, 0 3, 0 6, 6 0, 0 0))')
...

-- trojkat, 9 punktow
GeomFromText('POLYGON((0 0, 0 2, 0 3, 0 4, 0 6, 6 0, 4 0, 3 0, 2 0, 0 0))');
```

Kolejność dodawanych punktów jest zaznaczona na rys.1 liczbami w kółkach.

Dane przechowuje się w tabeli Obiekty, zawierającej tysiąc lub sto tysięcy wierszy (zależnie od badanego przypadku). Schemat tabeli Obiekty przedstawia rys. 5.

Obiekty	
PK	<u>id_obiektu</u>
	obszar

Rys. 5. Schemat tabeli Obiekty

Fig. 5. Schema of Objects table

W ramach badań przedstawiono przykładowe zapytania:

- Z1 – suma powierzchni obiektów (zgodne z OpenGIS OGC i SQL/MM):

```
SELECT SUM(Area(obszar)) FROM Obiekty;
```

- Z2 – powierzchnia sumy obiektów (zgodne z SQL/MM):

```
SELECT Area(ST_Union(dane)) FROM Obiekty;
```

Zapytania te wykonano w różnych środowiskach sprzętowych, dla różnych platform systemowych.

Pierwsze zapytanie wykonano na komputerze klasy serwerowej pod kontrolą systemu Linux. Zainstalowanym systemem zarządzania bazą danych było oprogramowanie MySQL 5.0.67 (z rozszerzeniem Spatial [9]). Serwer był wyposażony w procesor Intel Xeon 1.6 GHz (2 rdzenie), 4 GB RAM DDR1 333 MHz, tabela Obiekty zawierała sto tysięcy wierszy.

Zapytanie drugie zostało wykonane na komputerze PC o charakterze stacji roboczej, z zainstalowanym systemem zarządzania bazą danych PostgreSQL 8.4.6 [10] oraz rozszerzeniem przestrzennym Postgis 1.5.0 [11]. Komputer posiadał procesor Intel Pentium IV 2,8 GHz (Hyper-threading), 1 GB RAM DDR1 333MHz, HDD Seagate 7200 obr/min. Tabela Obiekty zawierała tysiąc wierszy.

Warto nadmienić, iż brak implementacji SQL/MM oraz częściowa implementacja OpenGIS OGC nie pozwalały w systemie MySQL Spatial wykonać drugiego zapytania [12].

Tabela 1

Czasy wykonania zapytania Z1, MySQL 5.0.67 [s]

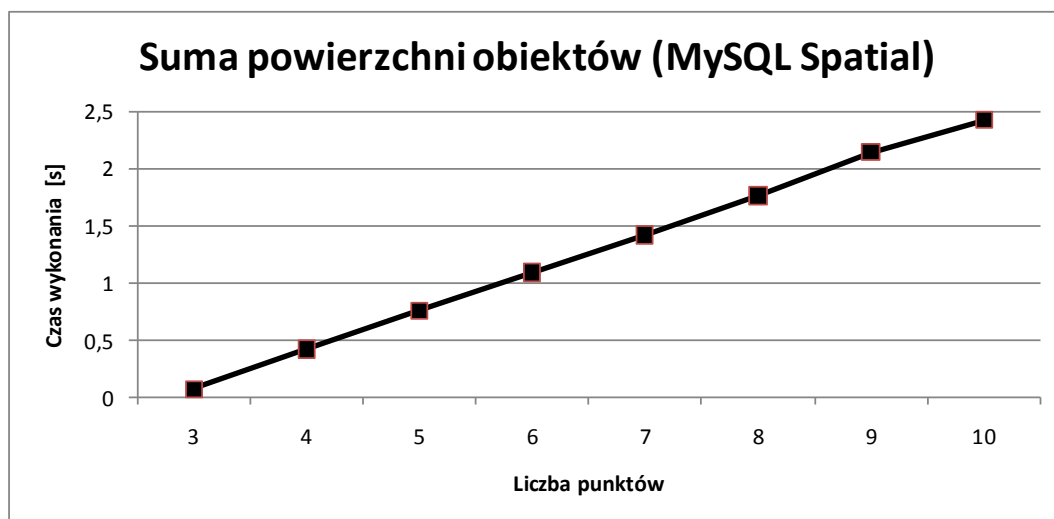
Liczba punktów	3	4	5	6	7	8	9	10
MySQL Spatial, Z1	0,07	0,42	0,76	1,09	1,42	1,76	2,14	2,42

Tabela 2

Czasy wykonania zapytania Z2, PostGIS 1.5.0 [s]

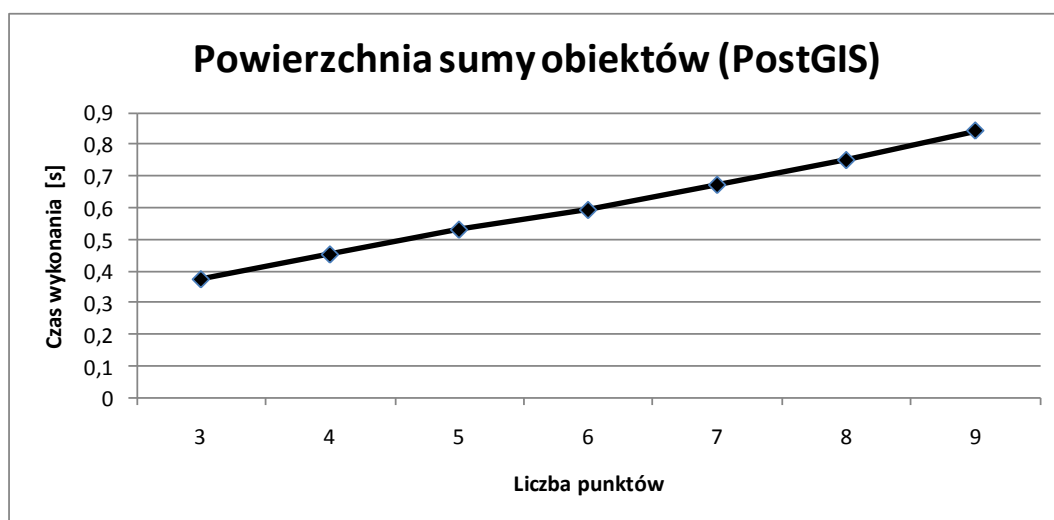
Liczba punktów	3	4	5	6	7	8	9
PostGIS, Z2	0,375	0,453	0,531	0,593	0,672	0,75	0,843

Za czasy przetwarzania poszczególnych zapytań przyjęto wartości minimalne (powtarzalne) z serii zapytań. Serie te cechowały niewielkie odchylenia standardowe. Wyniki zebrano w tabelach 1 i 2 i przedstawiono na wykresach (rys. 6 i rys. 7).



Rys. 6. Czasy wykonania zapytań o sumę powierzchni obiektów

Fig. 6. Execution times for sum of objects area queries



Rys. 7. Czasy wykonania zapytań o powierzchnię sumy obiektów

Fig. 7. Execution times for area of objects sum queries

Eksperymentalne testy wykazały, zarówno w pierwszym, jak i w drugim przypadku, liniową zależność czasu wykonania wybranych zapytań w stosunku do liczby punktów, z których składały się obiekty. Każdy punkt w pierwszym przypadku zwiększał czas wykonania zapytania o ok. 300 ms, natomiast w drugim przypadku o ok. 80 ms. Uzasadniony jest zatem wniosek, iż redukcja jak największej liczby punktów w danych przestrzennych jest istotna dla optymalizacji zapytań przetwarzających te dane.

4. Optymalna generalizacja danych przestrzennych w bazie danych

W poprzedniej sekcji została eksperymentalnie uzasadniona minimalizacja liczby punktów danego obiektu przestrzennego. Celowe zatem jest określenie metody minimalizacji i zakresu, w jakim można taką technikę zastosować. Można udowodnić, że przedstawiony problem optymalizacji, przy ograniczeniach związanych z powierzchnią obiektów (rys. 3), jest zagadnieniem NP-trudnym.

Szkic dowodu

W odniesieniu do bazy danych przestrzennych, najszybsze przetwarzanie zapytań uzyskuje się przy minimalnej liczbie punktów. Dla ciągu punktów opisujących figurę wypukłą można przyjąć dodatkowy zapis (1), określający, czy dany punkt p_i został zredukowany czy nie (2):

$$(p_1, p_2, \dots, p_n) \quad (1)$$

$$\begin{cases} p_i = 0, \text{ punkt zredukowany} \\ p_i = 1, \text{ punkt obecny} \end{cases} \quad (2)$$

gdzie n oznacza liczbę punktów obiektu.

Redukcja liczby punktów redundantnych jest oczywistym krokiem podczas optymalizacji. Problemowe jest natomiast ustalenie, które punkty (i ile) można usunąć, tak by zachować uogólnione cechy obiektu. Dość dobrym ograniczeniem jest reguła usuwania punktów, tak by nie utracić zadanej części wartości powierzchni (wartość stała lub wartość względna, np. 5%).

Ciągowi punktów (1) nie da się jednak przypisać odpowiadającego mu ciągu utraty pola powierzchni kształtu, jaki ma miejsce w przypadku redukcji, ponieważ utrata pola przy redukcji dwóch sąsiednich punktów nie jest równoważna sumie utraty pól w przypadku pojedynczych redukcji tychże punktów. Dlatego do opisu sytuacji redukcji punktów sąsiadujących można wykorzystać podciągi (3):

$$(y_1, y_2, \dots, y_m), \quad (3)$$

gdzie m to liczba podciągów, y_i odpowiada podciąg punktów (4):

$$(p_{a+1}, p_{a+2}, \dots, p_{a+c}), \quad (4)$$

a c to liczba redukowanych punktów w podciągu.

Liczbę redukowanych punktów dla każdego podciągu można zapisać również w postaci ciągu (5):

$$(c_1, c_2, \dots, c_m). \quad (5)$$

Podobnie pola powierzchni, jakie traci obiekt w wyniku redukcji podciągu punktów, można zapisać w postaci ciągu (6):

$$(w_1, w_2, \dots, w_m). \quad (6)$$

Wprowadzana jest zmienna decyzyjna x_i , która podobnie jak p_i (2) określa, czy dany podciąg będzie użyty w generalizacji (7):

$$(x_1, x_2, \dots, x_m) \quad (7)$$

$$\begin{cases} x_i = 0, \text{ podciąg punktów pozostających} \\ x_i = 1, \text{ podciąg punktów zredukowanych} \end{cases} \quad (7a)$$

Funkcja celu prowadzi do jak największej liczby zredukowanych punktów, zatem zdefiniowana jest następująco (8):

$$F = \sum_{i=1}^m c_i x_i \rightarrow \max. \quad (8)$$

Zważając na przyjęte ograniczenia, na zmianę powierzchni przy dokonywanych redukcjach, nieprzekraczającą pewnej wartości stałej Q dla danego obiektu, należy wprowadzić ograniczenia (9):

$$\sum_{i=1}^m w_i x_i \leq Q. \quad (9)$$

Przyjmując założenia osłabiające problem, dotyczące występowania każdego punktu p_i w tylko jednym podciągu y_j ze zbioru wszystkich podciągów, możemy stwierdzić, iż ten szczególny przypadek jest równie trudny obliczeniowo jak problem plecakowy, zdefiniowany przez warunki analogiczne do (7–9). Zatem proponowana metoda generalizacji obiektu należy do klasy problemów NP-trudnych. ■

5. Wnioski

Niniejszy artykuł przedstawia cel generalizacji obiektów w bazach danych przestrzennych, jakim jest przyspieszenie wykonywania zapytań odwołujących się do tego typu danych. Eksperymentalnie dowiedziono widoczny wpływ obecności każdego punktu w figurze na czas przetwarzania. Zaproponowano postępowanie w celu optymalizacji problemu, jednocześnie wykazując NP-trudny charakter obliczeniowy problemu. W dalszych badaniach warto przetestować istniejące algorytmy upraszczające geometrie stosowane w kartografii oraz zaproponować algorytmy heurystyczne do tego problemu i zbadać ich wpływ na przyspieszenie zapytań.

BIBLIOGRAFIA

1. OGC – The Open Geospatial Consortium, <http://www.opengeospatial.org/>.
2. OpenGIS Implementation Specification for Geographic information – Simple feature access – SQL option, <http://www.opengeospatial.org/standards/sfs>.
3. ISO/IEC 13249-3:1999, Information technology – Database languages – SQL Multimedia and Application Packages – Part 3: Spatial, International Organization For Standardization, 2000.
4. McMaster R.B., Shea K.S.: Generalization in Digital Cartography. AAG, Washington DC, 1992.
5. Brassel K., Weibel R.: A review and conceptual framework of automated map generation. *Journal of Geographical Information Systems*, Vol. 2, Issue 3.
6. McMaster R.B., Shea K.S.: Cartographic Generalization in A Digital Environment. 9 International Symposium Computer Assisted Cartography, Baltimore 1989.
7. Ratajski L.: *Metodyka kartografii społeczno-gospodarczej*. PPWK, Warszawa-Wrocław 1989.
8. Douglas D.H., Peucker T.K.: Algorithms for the reduction of the number of Points Required to Represent a Digital Line or its Caricature. *The Canadian Cartographer*, Vol. 10, Issue 2.
9. MySQL Spatial, <http://dev.mysql.com/doc/refman/5.1/en/spatial-extensions.html>.
10. PostgreSQL Home Page, <http://www.postgresql.org/>.
11. PostGIS Home Page, <http://postgis.refrains.net/>.
12. Piórkowski A.: Mysql Spatial And Postgis – Implementations Of Spatial Data Standards, *EJPAU*, Vol. 14(1), No. 03, 2011.

Recenzenci: Dr inż. Piotr Bajerski
Dr inż. Dariusz Mrozek

Wpłynęło do Redakcji 18 stycznia 2011 r.

Abstract

This article describes the issue of generalization of spatial objects in spatial databases. The generalization issue is detailed presented. Graphical examples make the introduction to the scope of optimization. There are presented examples of operations (the sum of surface

objects, surface objects total) in which the reduction of redundant nodes strongly accelerates query. Queries are executed in popular open source database managing systems – MySQL and PostgreSQL. The issue of generalization of convex geometric objects is highlighted. It has been proven NP-difficulty of the operation the largest reduction in the number of nodes at a given accuracy.

Adresy

Adam PIÓRKOWSKI: Akademia Górniczo-Hutnicza, Katedra Geoinformatyki i Informatyki Stosowanej, al. Mickiewicza 30, 30-059 Kraków, Polska, pioro@agh.edu.pl.

Artur KRAWCZYK: Akademia Górniczo-Hutnicza, Katedra Ochrony Terenów Górniczych, Geoinformatyki i Geodezji Górniczej, al. Mickiewicza 30, 30-059 Kraków, Polska, art-kraw@agh.edu.pl.