STUDIA INFORMATICA 2008
Volume 29 Number 3A (78)

Marcin GORAWSKI, Marcin BUGDOL
Politechnika Śląska, Instytut Informatyki

MODEL KOSZTOWY X-BR-DRZEWA W PRZESTRZENNYCH BAZACH DANYCH

Streszczenie. W artykule przestawiono model kosztowy x-BR-drzewa dla zapytań realizowanych w przestrzennych bazach danych. Model wyznacza koszt dla zapytań przestrzennych w bazach danych, rozumiany jako liczba dostępów do węzłów lub odczytów z dysku. Zaprezentowano wyniki testów, które pokazują dokładność analitycznych estymacji w porównaniu z rzeczywistymi wynikami.

Słowa kluczowe: przestrzenne bazy danych, x-BR-drzewo, estymacja kosztów

COST MODEL FOR X-BR-TREE IN SPATIAL DATABASES

Summary. The paper proposes the cost model for spatial databases based on x-BR-tree index. The model evaluates the cost for spatial queries in database, meant as a number of node accesses or disc reads. In addition, experimental results are presented, which shows the accuracy of analytical estimation compared with actual results.

Keywords: spatial database, x-BR-tree, cost estimation

1. Wstęp

Nieustanny rozwój systemów baz danych wymaga opracowywania coraz to nowszych metod dostępu. Szczególną popularnością w ostatnich latach cieszą się dane przestrzenne, w związku z czym położono nacisk na usprawnienie metod dostępu do tego typu danych [1, 2].

Z powodu ogromnej liczby rozwiązań wspierających przetwarzanie danych przestrzennych obecne badania skupiły się na opracowywaniu modeli analitycznych, które umożliwiają predykcję kosztów dostępu.

W pozycjach [3] oraz [4] przedstawiono przybliżone estymatory kosztu dla indeksów z rodziny R-drzew. Ich wyniki są dość dobre zarówno dla danych rozlokowanych w przestrzeni równomiernie, jak i nierównomiernie. Estymatory te stanowiły podstawę do rozważań analitycznych w kilku kluczowych pracach [5, 6] ze względu na wszechstronne możliwości.

Modele te nie są jednak doskonałe, co wyraźnie widać przy szacowaniu kosztów przy dostępie do struktur, wykorzystujących indeksy, opierające się na hierarchicznym podziale przestrzeni. Wynika to bezpośrednio z faktu, że zakładają one, iż rozmiar węzłów struktury indeksującej zależy tylko od rozmieszczenia obiektów w przestrzeni. Jest to zgodne ze specyfiką indeksów z rodziny R-drzew. Drzewa czwórkowe i indeksy bazujące na nich dzielą przestrzeń regularnie, przez co wynikowa struktura nie jest ściśle dopasowana do rozmieszczenia obiektów przestrzennych.

2. Indeks X-BR-drzewo

X-BR-drzewo (ang. *External Balanced Regular Tree*), zaprezentowany w [2], wywodzi się z idei drzew czwórkowych. Jest to struktura, która opiera się na hierarchicznym, równomiernym podziale indeksowanej przestrzeni. Dzięki modyfikacjom, które rozszerzają jego możliwości, jest strukturą o wiele wydajniejszą i bardziej funkcjonalną, aniżeli drzewa czwórkowe. Indeks x-BR–drzewo jest strukturą zrównoważoną, liście znajdują się na tym samym poziomie i odpowiadają stronom na dysku. Możliwość przechowywania w węzłach pośrednich większej liczby wpisów niż 4 (co ma miejsce dla drzew czwórkowych) zmniejsza wysokość drzewa, a poszczególne węzły mogą być bardziej efektywnie wykorzystane.

2.1. Budowa

Drzewo x-BR składa się z dwóch rodzajów węzłów. Pierwszym z nich są węzły zewnętrzne, zawierające obiekty przestrzenne, których liczbę określa pojemność węzła C. Często nazywa się je także liśćmi. Podział liścia następuje w wyniku jego przepełnienia. Proces ten zaczyna się od rekurencyjnego podziału obszaru liścia na cztery równe części. Następnie wybierana jest ćwiartka zawierającą najwięcej obiektów. Proces ten przebiega tak długo, jak długo wynikowe obszary będą zawierały więcej niż xC lub mniej niż $(1–x)C$ obiektów, gdzie $x \in (0{,}5;1)$ jest współczynnikiem podziału [2]. Wartość tego współczynnika wpływa na liczbę koniecznych podziałów. W miarę zbliżania się tej wartości do 0,5 musi zostać wykonana coraz większa liczba podziałów.

Algorytm, w postaci pseudokodu, dokonujący podziału liścia przedstawia się następująco:

```
leafSplit()
BEGIN
    DO
        podziel region zawierający najwięcej elementów na cztery równe części
        wyszukaj część zawierającą najwięcej elementów
    WHILE (liczba elementów w wybranej części < x*C && liczba elementów w
           wybranej części > (1-x)*C)
    przenieś odpowiednie elementy do nowego liścia
    zaktualizuj wpisy w rodzicu
END
```

Zauważono, iż sposób doboru współczynnika *x* nie może być dowolny. Przykład pokaże sytuację, w której nieodpowiedni dobór współczynnika podziału może spowodować niepoprawne działanie algorytmu podziału liści. Przyjęto następujące założenia: pojemność liści wynosi 100, współczynnik podziału *x* równa się 0,7, a obiekty rozmieszczone są równomiernie. Jeżeli do liścia wstawi się 100 obiektów, to w każdym regionie, będącym ćwiartką regionu początkowego, liczba wpisów wynosi 25. Wstawienie kolejnego elementu spowoduje przepełnienie liścia i konieczność jego podziału. Nowy węzeł nie powinien posiadać więcej niż 70 (0,7*100), i mniej niż 30 ((1–0,7)*100) wpisów. Można zauważyć, iż algorytm nie dokona podziału węzła według przedstawionych założeń, ponieważ liczba elementów w każdym z nowych regionów nie przekroczy 26. Co więcej, działanie algorytmu spowoduje błąd wykonania z powodu pętli nieskończonej lub przepełnienia stosu w przypadku wywołań rekurencyjnych. Na podstawie powyższego rozumowania wyznaczono zależność na wartość współczynnika *x,* którą można wyrazić jako:

$$\lfloor (1-x)*C \rfloor < \frac{1}{4}C + 1 \tag{1}$$

gdzie: C – pojemność węzła, x – współczynnik podziału węzła, $x \in (0{,}5;\ 1)$.

Drugim rodzajem węzłów są węzły pośrednie. Zawierają one wpisy w postaci [*adres*; *wskaźnik*], gdzie *adres* zawiera symbole kierunkowe obszaru obejmowanego przez potomka, natomiast pole *wskaźnik* jest odnośnikiem do dziecka. Adres zapisany jest w postaci zbioru symboli kierunkowych, który zawiera następujące elementy: NW, NE, SW, SE oraz *. Oznaczają one kolejne ćwiartki danego regionu, natomiast symbol * oznacza cały pozostały region.

Adres umożliwia zapisanie większej liczby danych w węźle pośrednim, ponieważ potomek może teraz obejmować nie tylko jedną ćwiartkę przestrzeni rodzica, ale również o wiele mniejsze regiony. Wpisy rozpatrywane są według ich kolejności w węźle pośrednim. Obszar potomka zależy nie tylko od jego adresu, ale również od adresów wpisów znajdujących się przed nim. Obszar ten jest określany jako różnica pomiędzy przestrzenią obejmowaną adresem danego potomka a przestrzenią obejmowaną przez adresy wpisów znajdujących się przed nim. W przypadku przepełnienia węzła pośredniego następuje jego podział, który wykonywany jest według poniższych kroków:

- budowa drzewa czwórkowego na podstawie wpisów w przepełnionym węźle,
- wyszukanie najlepszego miejsca podziału,
- przeniesienie wpisów do nowo utworzonego węzła,
- aktualizacja wpisów w rodzicu, a w przypadku jego braku utworzenie nowego węzła pośredniego.

Szczegóły tych kroków zostały omówione wraz z przykładami w [2].

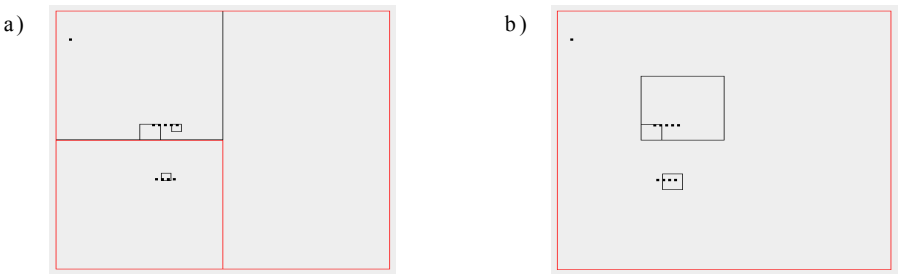
2.2. Problemy z oszacowaniem liczby węzłów

Rozmiar i kształt węzłów w indeksie x-BR-drzewo wynika ze sposobu podziału przestrzeni roboczej. Podziału tego nie można niestety dokładnie przewidzieć. Zależy on od zbioru danych, ale również od kolejności wstawiania elementów. Poniższy przykład ilustruje wspomniany problem.

Przykład 1

Wstawiono kilka punktów do indeksu o następujących parametrach:

- pojemność liści – 3;
- pojemność węzłów pośrednich – 4;
- próg podziału – 0,7 (podział liścia będzie następował tak długo, dopóki nie będzie on zawierał maksymalnie 2 punktów).



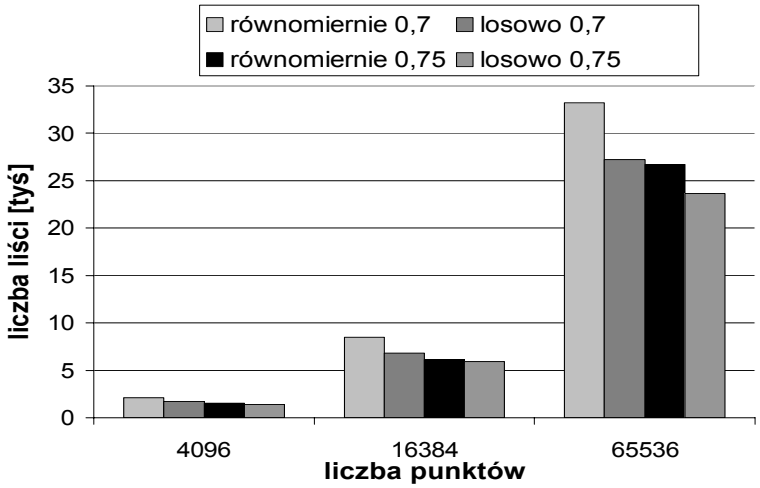
Rys. 1. Wpływ kolejności wstawienia wpisów na strukturę x-BR-drzewa: a)wstawianie losowe, b) wstawianie uporządkowane
Fig. 1. The influence of the entries inserts order on the x-BR-tree structure: a) random insertion, b) ordered insertion

Jak widać na powyższych ilustracjach rys. 1, struktury wynikowe w obu przypadkach są różne. W a) zarówno liczba liści, jak i węzłów pośrednich jest większa niż w przypadku b), a ponadto liście są o wiele mniejsze. Indeks ten w pierwotnej postaci nie przewiduje łączenia węzłów, więc nie ma możliwości, by struktura węzłów była bardziej efektywna. Z przykładu wynika, że liczba liści oraz ich rozmiar nie zależą tylko od zbioru danych, lecz w dużej mierze również od kolejności wstawiania elementów.

Przykład 2

Porównanie liczby węzłów w zależności od sposobu wstawienia, wartości współczynnika podziału oraz liczności zbioru danych:

- liczba wstawionych punktów: 4096, 16384 oraz 65536;
- rozmiar przestrzeni: 512×512 punktów;
- pojemność liści: 4;
- pojemność węzłów pośrednich: 4;
- współczynnik podziału: 0,7 oraz 0,75 (podział liścia przebiega tak długo, aż każdy region będzie zawierać co najwyżej dwa wpisy dla parametru 0,7 i co najwyżej trzy wpisy dla wartości 0,75);
- sposób wstawiania punktów – równomiernie od „lewej” do „prawej” i od „góry” do „dołu”, w drugim przypadku ten sam zbiór punktów wstawiono losowo.



Rys. 2. Liczba liści w zależności od liczby wstawionych punktów
Fig. 2. Number of leaves depending on the number of inserted points

Na podstawie rys. 2 można zauważyć, że liczba węzłów w x-BR-drzewie bardzo silnie zależy od sposobu wstawiania, a także od współczynnika podziału.

Dla współczynnika podziału o wartości 0,75 różnice przy mniejszych zbiorach danych nie są znaczne. Jednak dla zbioru danych o liczności 65536 elementów liczba liści przy wstawianiu równomiernym jest o ponad 12% mniejsza niż przy umieszczaniu wpisów losowo. Dla wartości współczynnika podziału równej 0,7 różnice są jeszcze większe. Liczba węzłów w każdym przypadku różni się o co najmniej 20%.

Jak widać na podstawie powyższego eksperymentu, liczba węzłów, a w szczególności liści, jest bardzo trudna do oszacowania. Wartość ta jest podstawą w przedstawionym modelu kosztowym, co sprawia, że opracowanie dokładnej zależności nastręcza pewnych problemów.

2.3. Model kosztów

Model kosztów dla zapytań przestrzennych z użyciem indeksu x-BR-drzewa można przedstawić jako zależność probabilistyczną. Przedstawione rozważania będą dążyły do wyrażenia kosztu zapytania przestrzennego z użyciem informacji o liczności zbioru danych. Na chwilę obecną model będzie się ograniczał tylko do poziomów liści, aby nie komplikować rozważań. Przestrzeń, w której będzie omawiany model, jest jednostkowa.

Zapytanie przestrzennej selekcji odwołuje się do węzłów znajdujących się w jego obszarze. Dostęp do węzła następuje zarówno, gdy zawiera się on w całości w oknie zapytania, jak i gdy zapytanie przecina węzeł.

Prawdopodobieństwo zawierania się obszaru danych s w oknie zapytania q w przestrzeni d-wymiarowej wynosi:

$$P_{cont} = \prod_{i=1}^{d} (q_i - s_i) \tag{2}$$

Prawdopodobieństwo przecinania się obszaru danych s w oknie zapytania q w przestrzeni d-wymiarowej wynosi:

$$P_{cross} = \prod_{k=1}^{d} (q_k + s_k) - \prod_{k=1}^{d} (q_k - s_k) \tag{3}$$

Poprzez koszt wykonania zapytania selekcji z użyciem indeksu będziemy rozumieć liczbę odwołań do liści x-BR-drzewa *LA_total*, co odpowiada liczbie odczytów z pamięci zewnętrznej. Wartość ta będzie zależała od rozmiaru pojedynczego liścia oraz od rozmiaru zapytania. Wzór na liczbę przeciętych liści w zależności od rozmiaru zapytania można przedstawić następująco:

$$LA\_total(q) = N_1 * (P_{cont} + P_{cross}) = N_1 * \prod_{k=1}^{d} (q_k + s_k) \tag{4}$$

gdzie: P_{cont} – prawdopodobieństwo zawierania się obiektu danych w zapytaniu, P_{cross} – prawdopodobieństwo przecinania się obiektu danych z zapytaniem, d – liczba wymiarów przestrzeni, N_1 – liczba liści, q_k – długość boku zapytania w k-tym wymiarze, s_k – średnia długość boku liścia w k-tym wymiarze.

Najprostszy model bazuje na zależności (4). Liczbę liści w indeksie N_I można uzyskać na podstawie zebranych statystyk drzewa. Rozmiar pojedynczego liścia można wyznaczyć, dzieląc jeden wymiar przestrzeni roboczej przez pierwiastek d-tego stopnia, gdzie d to liczba wymiarów przestrzeni:

$$s_k = 1 / \sqrt[d]{N_1} \tag{5}$$

gdzie: s_k – długość boku danych w k-tym wymiarze, d – liczba wymiarów, N_1 – liczba liści.

Przyjęto, że przestrzeń robocza jest dwuwymiarowa oraz że każdy z wymiarów przestrzeni posiada taką samą długość (kwadratowa przestrzeń). Podstawiając równanie (5) do (4), otrzymujemy:

$$LA\_total(q) = N_1 * (q + \frac{1}{\sqrt{N_1}})^2 \tag{6}$$

gdzie: *LA_total* – całkowita liczba dostępów do liści, N_1 – liczba liści w drzewie, q – długość boku zapytania.

Powyższy model nie bierze pod uwagę algorytmu tworzenia indeksu, lecz, bazując na liczbie liści, uśrednia ich rozmiar i te wartości wykorzystuje do predykcji kosztu. Poniżej podjęto próbę bardziej dokładnego oszacowania liczby liści oraz ich rozmiaru. Poczyniono następujące założenia przy wyprowadzaniu tej zależności:

- przestrzeń jest dwuwymiarowa,
- każdy z wymiarów posiada taką samą długość (kwadrat),
- liczba wstawionych punktów wynosi 2^{2m},
- długość każdego boku przestrzeni jest potęgą liczby 2, aby podział przestrzeni zawsze dawał w wyniku liczby naturalne,
- wypełnienie węzłów jest bliskie maksimum.

Mając dane: pojemność liścia C oraz liczbę wstawionych elementów n, rozpatrzono, jak będzie się przedstawiać zależność liczby liści (N_1) od C i n.

$$\begin{cases} n \le C & N_1 = 1 \\ \frac{3n}{4} \le C < n & N_1 = 2 = 2^1 \\ \frac{2n}{4} \le C < \frac{3n}{4} & N_1 = 3 = 3 * 2^0 \\ \frac{n}{4} \le C < \frac{2n}{4} & N_1 = 4 = 2^2 \\ \frac{3n}{16} \le C < \frac{n}{4} \Leftrightarrow \frac{3n}{4^2} \le C < \frac{n}{4^1} & N_1 = 8 = 2^3 \\ \frac{2n}{16} \le C < \frac{3n}{16} \Leftrightarrow \frac{2n}{4^2} \le C < \frac{3n}{4^2} & N_1 = 12 = 3 * 2^2 \\ \frac{n}{16} \le C < \frac{2n}{16} \Leftrightarrow \frac{n}{4^2} \le C < \frac{2n}{4^2} & N_1 = 16 = 2^4 \\ \frac{3n}{64} \le C < \frac{n}{16} \Leftrightarrow \frac{3n}{4^3} \le C < \frac{n}{4^2} & N_1 = 32 = 2^5 \\ \frac{2n}{64} \le C < \frac{3n}{64} \Leftrightarrow \frac{2n}{4^3} \le C < \frac{3n}{4^3} & N_1 = 48 = 3 * 2^4 \\ \frac{n}{64} \le C < \frac{2n}{64} \Leftrightarrow \frac{n}{4^3} \le C < \frac{2n}{4^3} & N_1 = 64 = 2^6 \\ \vdots & \\ \vdots & \end{cases}$$

Zaprezentowany podział na przedziały można zapisać ogólnie, w zależności od parametru k, będącego dowolną liczbą naturalną. Liczba liści również zależy od tego parametru i przedstawia się następująco:

$$\begin{cases} n \le C & N_1 = 1 \\ \frac{3n}{4^k} \le C < \frac{n}{4^{k-1}} & N_1 = 2^{2k-1} \\ \frac{2n}{4^k} \le C < \frac{3n}{4^k} & N_1 = 3 * 2^{2(k-1)} \\ \frac{n}{4^k} \le C < \frac{2n}{4^k} & N_1 = 2^{2k} \end{cases} \qquad k = 1,2,\ldots \tag{7}$$

Wiadomo, że C będzie należeć do dokładnie jednego z tych przedziałów dla dokładnie jednego k. Chcąc wyznaczyć wartość k, która jest niezbędna do oszacowania liczby liści, należy przekształcić wzór (7) w następujący sposób:

$$\frac{3n}{4^k} \le C < \frac{n}{4^{k-1}} \Leftrightarrow \qquad k = 1,2,\ldots$$

$$\Leftrightarrow \left(\frac{3n}{4^k} \le C \quad \wedge \quad C < \frac{n}{4^{k-1}}\right) \Leftrightarrow$$

$$\Leftrightarrow \left(4^k \ge \frac{3n}{C} \quad \wedge \quad 4^{k-1} < \frac{n}{C}\right) \Leftrightarrow$$

$$\Leftrightarrow \left(\log_4 4^k \ge \log_4 \frac{3n}{C} \quad \wedge \quad \log_4 4^{k-1} < \log_4 \frac{n}{C}\right) \Leftrightarrow$$

$$\Leftrightarrow \left(k \ge \log_4 \frac{3n}{C} \quad \wedge \quad k - 1 < \log_4 \frac{n}{C}\right) \Leftrightarrow$$

$$\Leftrightarrow \left(k \ge \log_4 \frac{3n}{C} \quad \wedge \quad k < \log_4 \frac{n}{C} + 1\right) \Leftrightarrow$$

$$\Leftrightarrow \left(k \ge \log_4 \frac{3n}{C} \quad \wedge \quad k < \log_4 \frac{n}{C} + \log_4 4\right) \Leftrightarrow$$

$$\Leftrightarrow \left(k \ge \log_4 \frac{3n}{C} \quad \wedge \quad k < \log_4 \frac{4n}{C}\right) \Leftrightarrow$$

$$\Leftrightarrow \log_4 \frac{3n}{C} \le k < \log_4 \frac{4n}{C}$$

Analogicznie można przekształcić pozostałe przedziały, uzyskując:

$$\begin{cases} n \le C & N_1 = 1 \\ \log_4 \frac{3n}{C} \le k < \log_4 \frac{4n}{C} & N_1 = 2^{2k-1} \\ \log_4 \frac{2n}{C} \le k < \log_4 \frac{3n}{C} & N_1 = 3 * 2^{2(k-1)} \\ \log_4 \frac{n}{C} \le k < \log_4 \frac{2n}{C} & N_1 = 2^{2k} \end{cases} \tag{8}$$

Ponieważ, jak wspomniano powyżej, istnieje dokładnie jedno k, spełniające dokładnie jedną z powyższych nierówności, zatem w celu wyznaczenia wartości k należy sprawdzić, czy sufit wyrażenia mniejszego lub równego k jest mniejszy od wyrażenia większego od k. Jeśli tak, wówczas sufit ten jest szukanym parametrem k. Jeśli nie, należy sprawdzić w ten sam sposób kolejną nierówność. Następnie za pomocą k wyznacza się liczbę liści w indeksie. Pomimo że przyjęto, iż wypełnienie węzłów będzie maksymalne, to na podstawie eksperymentów oraz literatury można stwierdzić, że w praktyce wypełnienie węzłów osiąga wartość ok. 70%. Wynika z tego, że przewidywana liczba liści będzie przechowywać ok. 70% wszystkich wpisów. Można zatem przyjąć, że odpowiednio większa liczba liści przechowuje informacje o całym zbiorze danych. Ostateczna liczba liści wyraża się wzorem:

$$N_1\_total = \frac{1}{f} N_1 \tag{9}$$

gdzie: N_1_total – liczba liści, f – współczynnik wypełnienia węzłów, N_1 – przewidziana liczba liści; zgodnie ze wzorem (8).

Na tej podstawie N_1_total oblicza się średnią długość boku liścia przy użyciu zależności (5):

$$s_k = 1/\sqrt[d]{N_1\_total} \tag{10}$$

Ostatecznie z równań (4), (9), (10) otrzymujemy:

$$LA\_total(q) = \frac{1}{f} N_1 * (q + \frac{1}{\sqrt{\frac{1}{f} N_1}})^2 \tag{11}$$

gdzie: LA_total – całkowita liczba dostępów do liści, N_1 – liczba liści w drzewie; zgodnie ze wzorem (8), q – długość boku zapytania, f – współczynnik wypełnienia liścia.

3. Eksperymentalna ocena estymatora kosztu

Przeprowadzono dwa eksperymenty mające na celu sprawdzenie dokładności rozwiązań przedstawionych w rozdziale 2.3. Wykorzystano zbiory danych o liczności 40 oraz 60 tysięcy punktów. Przestrzeń robocza miała rozmiar 512×512 punktów. Ustawienia x-BR-drzewa to:

- pojemność liści i węzłów pośrednich – 4;
- współczynnik podziału – 0,7 oraz 0,75.

Punkty załadowano do przestrzennej bazy danych losowo. Ich rozkład w jednym przypadku był równomierny, a w drugim normalny (Gaussa).

3.1. Błąd predykcji dostępów do liści

W tym doświadczeniu badano średni błąd względny predykcji obu przedstawionych modeli przy szacowaniu kosztów dla podanych zapytań. Średni błąd względny predykcji zdefiniowano jako:

$$\delta = \frac{1}{k} \sum_{i=1}^{k} \frac{|x_i - v_i|}{v_i} \tag{12}$$

gdzie: δ – średni błąd względny predykcji, k – liczba zapytań, x_i – przewidywana liczba odwiedzonych liści dla i-tego zapytania, v_i – rzeczywista liczba odwiedzonych liści dla i-tego zapytania.

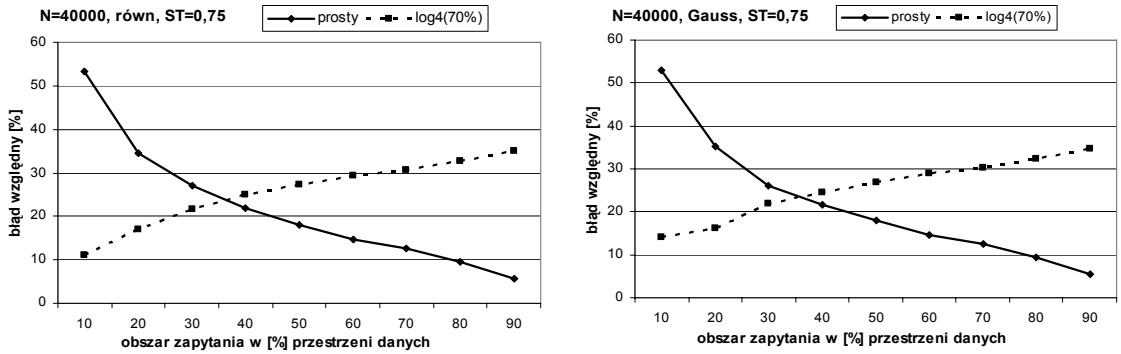
Dla każdego rozmiaru zapytania (od 10% do 90% obszaru przestrzeni roboczej, co 10%) wygenerowano 200 losowych zapytań. Błąd mierzono dla każdego zapytania, a wyniki uśredniono. Oznaczenia na wszystkich wykresach są następujące:

N – oznacza liczność zbioru danych,

„równ.", „Gauss" – rozkład zbioru danych (równomierny oraz normalny),

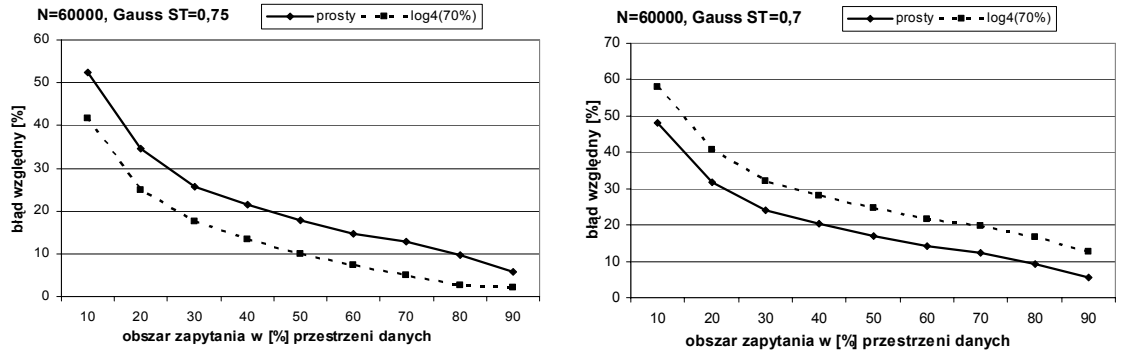
ST – „split treshold" współczynnik podziału.

Przebieg oznaczony jako „prosty" to wyniki otrzymane z wykorzystaniem zależności (6), a przebieg „log" to wyniki uzyskane na podstawie zależności (11).



Rys. 3. Błąd względny modelu kosztowego w zależności od stopnia pokrycia przez zapytanie dla zbioru 40000

Fig. 3. Relative error of the cost model depending on the query size for data set 40000



Rys. 4. Błąd względny modelu kosztowego w zależności od stopnia pokrycia przez zapytanie dla zbioru 60000

Fig. 4. Relative error of the cost model depending on the query size for data set 60000

Model „prosty" jest o wiele mniej dokładny dla małych zapytań (nawet do 50%), jednak błąd ten maleje wraz ze wzrostem obszaru zapytania. Wynika to z faktu, że bazuje on na średnim rozmiarze liścia, wyliczonym na podstawie liczby liści, która jest odczytywana ze statystyk drzewa. Im mniejszy obszar zapytania, tym bardziej faktyczny rozmiar liści odbiega od średniego. Wzrost rozmiaru zapytania powoduje, iż przeciętych jest więcej liści o większym rozmiarze, przez co średnia rozmiarów odwiedzonych liści zbliża się do wyznaczonej analitycznie.

Jak widać, na większości przebiegów wykresy błędu dla obu modeli przecinają się dla zapytań o stopniu pokrycia 30-40%. Można zatem spróbować połączyć oba modele. Dla zapytań o stopniu pokrycia do 35% należy korzystać z zależności (11), natomiast dla większych obszarów zapytań oszacowanie będzie dokładniejsze przy zastosowaniu zależności (6). W ten sposób błąd modelu „połączonego" nie powinien przekroczyć 30%, co wydaje się dobrym wynikiem przy pierwszym podejściu do tego zagadnienia.

3.2. Estymowana liczba liści

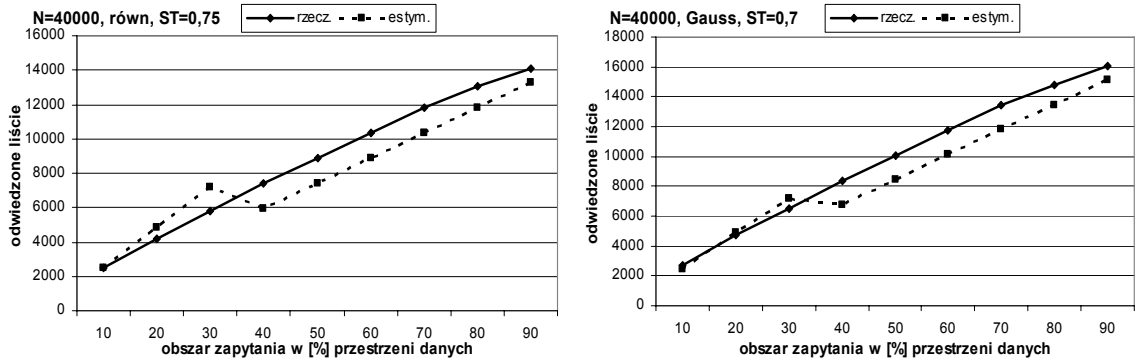
Eksperyment miał na celu porównanie liczby odwiedzonych liści podczas przetwarzania zapytania z wartością estymowaną. Liczba odwiedzonych liści była badana w zależności od rozmiaru zapytania. Rozmiar ten zmieniał się w zakresie od 10% do 90% z krokiem co 10%. Dla każdej wartości obszaru wygenerowano 200 losowych zapytań. Estymacji dokonywano dla każdego zapytania osobno, a wyniki uśredniono. Estymacja liczby odwiedzonych liści oparta była na modelu będącym połączeniem obu rozwiązań przedstawionych w rozdziale 2.3. Zgodnie z wnioskami przedstawionymi w poprzednim rozdziale (3.1), dla zapytań o rozmiarze poniżej 35% wykorzystano równanie (25), natomiast dla zapytań o większym obszarze korzystano z równania (30). Oznaczenia na wszystkich wykresach są następujące:

N – oznacza liczność zbioru danych,

„równ.", „Gauss" – rozkład zbioru danych (równomierny oraz normalny),

ST – „split treshold" współczynnik podziału.

Przebiegi oznaczone jako „rzecz." prezentują rzeczywistą liczbę dostępów do liści, natomiast przebiegi oznaczone jako „estym." przedstawiają wartość estymowaną.

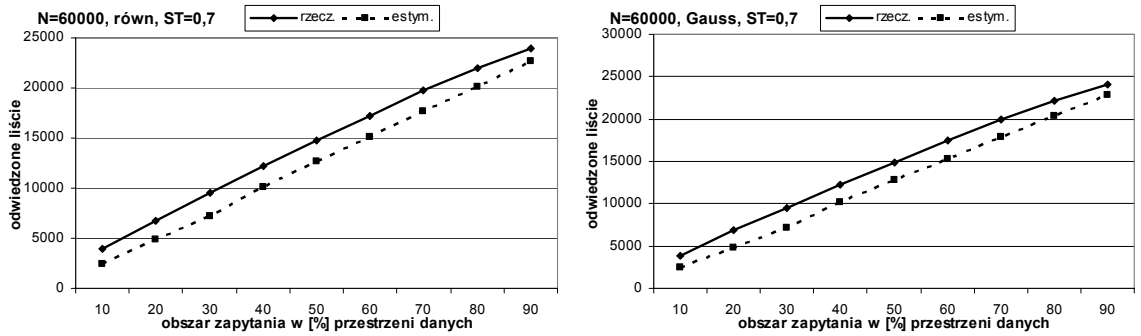


Rys. 5. Rzeczywista i estymowana liczba odwiedzonych liści w zależności od stopnia pokrycia przez zapytanie dla zbioru 40000

Fig. 5. Actual and estimated number of leaves accesses depending on the query size for data set 40000

Opierając się na wynikach powyższego doświadczenia, można zauważyć, że wartość estymowana liczby dostępów do liści przy przetwarzaniu zapytań przestrzennych nie odbiega

znacząco od wartości rzeczywistej. Dla zbioru o liczności 40 tysięcy punktów (rys. 4) widać miejsce, w którym następuje przecięcie przebiegów wartości rzeczywistej i estymowanej.



Rys. 6. Rzeczywista i estymowana liczba odwiedzonych liści w zależności od stopnia pokrycia przez zapytanie dla zbioru 60000

Fig. 6. Actual and estimated number of leaves accesses depending on the query size for data set 60000

Pokrywa się ono z rozmiarem zapytania, będącym wartością graniczną dla zależności (6) i (11). W przypadku zbioru o liczności 60 tysięcy punktów (rys. 5) przebiegi nie przecinają się, a różnica wartości estymowanej i rzeczywistej jest stała. Przekłada się to na zmniejszanie się średniego błędu względnego wraz ze wzrostem rozmiaru zapytania.

4. Podsumowanie

Zaproponowano model kosztowy dla zapytań przestrzennej selekcji z użyciem indeksu x-BR-drzewa. Opracowano dwa rozwiązania, które opierają się w dużej części na probabilistyce i estymacji. Wyniki badań wykazały, że każde z tych rozwiązań ma pewien zakres obszaru zapytania, w którym daje dokładniejsze wyniki. Co więcej, zakresy te są różne dla obu rozwiązań. Połączenie obu modeli skutkuje dokładniejszą estymacją w całym zakresie obszaru zapytania. Takie rozwiązanie jest obarczone pewnym błędem, jednak tylko w nielicznych przypadkach przekracza on 30%. Trudności związane z uzyskaniem dokładnych oszacowań wynikają z właściwości x-BR-drzewa i jego algorytmu podziału.

Dalsze prace powinny skupić się na rozwinięciu przedstawionego modelu, aby uwzględniał również koszt dostępu do wyższych poziomów struktury indeksującej.

LITERATURA

1. Gorawski M., Malczok R.: On Efficient Storing and Processing of Long Aggregate Lists. Proceedings of the 7th International Conference Data Warehousing and Knowledge Discovery (DaWak2005, LNCS 3589), Copenhagen, Denmark 2005.

2. Vassilakopoulos M., Manolopoulos Y.: External Balanced Regular (x-BR) Trees: New Structure for Very Large Spatial Databases. Technical Report TR99-13.
3. Faloutsos C., Kamel I.:Beyond Uniformity and Independence: Analysis of R-trees Using the Concept of Fractal Dimension, In Proceedings of the 13th ACM Symposium on Principles of Database Systems (PODS), 1994.
4. Theodoridis Y., Sellis T.: A model for the Prediction of R-tree Performance. Proc. Symp. Principles of Database Systems, 1996.
5. Yu S., Atluri V.,. Adam N. R.: Selective View Materialization in a Spatial Data Warehouse. DaWaK 2005: s. 157÷167.
6. Dellis E., Seeger B., Vlachou A.: Nearest Neighbor Search on Vertically Partitioned High-Dimensional Data. DaWaK 2005: s. 243÷253.

Recenzent: Dr hab. inż. Maciej Zakrzewicz

Wpłynęło do Redakcji 30 października 2007 r.

Abstract

The paper proposes the cost model for spatial databases based on x-BR-tree index. For this solution mathematical formulas were created, that express the cost of selection queries using x-BR-tree.

The model evaluates the cost for spatial queries in database, meant as a number of node accesses or disc reads. In addition, experimental results are presented, which shows the accuracy of analytical estimation compared with actual results.

Adresy

Marcin GORAWSKI: Politechnika Śląska, Instytut Informatyki, ul. Akademicka 16, 44-100 Gliwice, Polska, M.Gorawski@polsl.pl.
Marcin Bugdol: Politechnika Śląska, Instytut Informatyki, ul. Akademicka 16, 44-100 Gliwice, Polska, M.Bugdol@polsl.pl.