STUDIA INFORMATICA 2007
Volume 28 Number 4 (74)

Marcin GORAWSKI, Michał THIELE
Politechnika Śląska, Instytut Informatyki

MODELE KOSZTOWE DLA INDEKSU STCAT

Streszczenie. Przedstawiono modele kosztowe służące do estymacji liczby dostępów do węzłów podczas realizacji zapytań zakresowych, agregacyjnych oraz *k*NN na indeksie STCAT (ang. *Spatio-Temporal Cup Aggregate Tree*). Zostały one opracowane na podstawie istniejących modeli kosztowych dla indeksów przestrzennych. Modele zaimplementowano, przetestowano i porównano z modelami dla R-drzewa.

Słowa kluczowe: indeksy przestrzenno-czasowe, modele kosztowe, zapytanie zakresowe, zapytanie agregacyjne, zapytanie *k*NN, STCAT, R-drzewo

COST MODELS FOR STCAT INDEX

Summary. The paper proposes cost models for STCAT (Spatio-Temporal Cup Aggregate Tree) index which lets us estimate number of node accesses during executing range, aggregate and *k*NN queries. It is based on existing cost models for spatial indices. The models was implemented, tested and compared with cost models for R-tree.

Keywords: spatio-temporal indices, cost models, range query, aggregate range query, *k*NN query, STCAT, R-tree

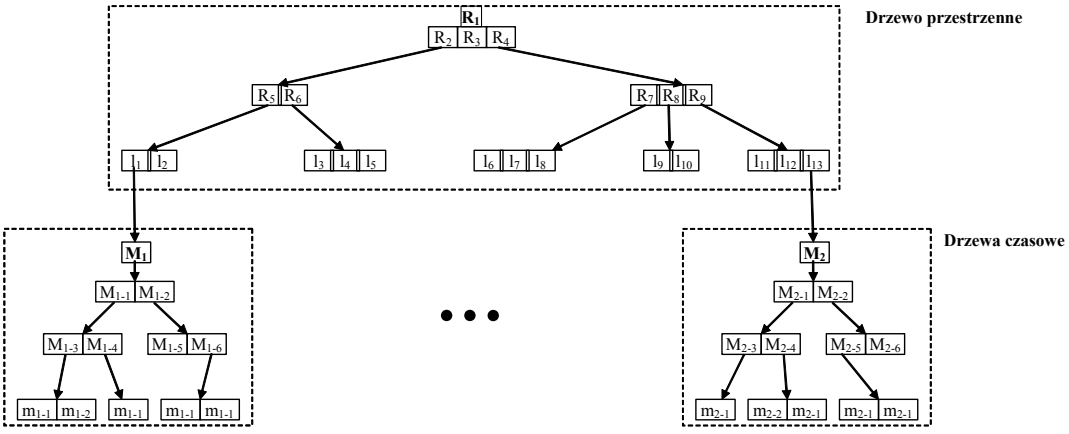
1. Wprowadzenie

We współczesnych systemach informatycznych gromadzących dane możliwość przetwarzania i analizy dużej ilości informacji staje się coraz częściej kluczowym czynnikiem, decydującym o ich przydatności. Jako przykład mogą tu posłużyć systemy nawigacji, sieci telefonii komórkowych oraz będący główną motywacją tej pracy system telemetryczny [1]. Zastosowanie w tych przypadkach, choćby nawet najlepszych tradycyjnych rozwiązań bazodanowych, często okazuje się niewystarczające. Naprzeciw tym wyzwaniom wychodzą specjalizowane systemy hurtowni danych (HD), które dzięki zastosowaniu zoptymalizowanych

pod kątem konkretnych zadań schematów baz danych, przetwarzaniu rozproszonemu oraz specjalizowanym narzędziom i algorytmom umożliwiają szybki i przekrojowy dostęp do olbrzymiej liczby danych. Stąd HD stanowi dobrą podstawę dla wszelkich systemów analitycznych, w tym systemów wspomagających podejmowanie decyzji. Aby szybki dostęp do HD był możliwy, dane muszą być wstępnie uporządkowane i pogrupowane hierarchicznie. Uporządkowanie to jest realizowane poprzez zastosowanie indeksów, najczęściej implementowanych jako struktury drzewiaste, tablice mieszające lub bitmapy. O wyborze konkretnej struktury indeksu decyduje głównie postać danych, ich wymiarowość oraz typ operacji, jakie będą na nich realizowane. W artykule zostaną zaproponowane modele kosztowe przestrzenno-czasowego drzewa agregacji kubełkowej STCAT [1, 2], pozwalające przewidzieć liczbę dostępów do jego węzłów podczas realizacji zapytań zakresowych, agregacyjnych i *k*NN. Modele kosztowe odgrywają dużą rolę w optymalizatorach baz danych, ponieważ pozwalają z wyprzedzeniem ocenić korzyści, jakie przyniesie zastosowanie konkretnego indeksu, a tym samym wybrać najkorzystniejszy wariant. Precyzyjny model kosztowy powinien więc być nierozłączną częścią każdego dobrego indeksu.

2. Indeks STCAT

Drzewo STCAT (ang. *Spatial-Temporal Cup Aggregate Tree*) to przestrzenno-czasowe kubełkowe drzewo agregacyjne opracowane przez autorów publikacji [1, 2]. W każdym liściu przestrzennego drzewa wielowymiarowego znajduje się lista korzeni do drzew czasowych, czyli jest to struktura listy drzew czasowych o jednym wymiarze podczepianych do liści drzewa wielowymiarowego (rys. 1).



Rys. 1. Przykładowa budowa drzew indeksu STCAT
Fig. 1. Sample STCAT structure

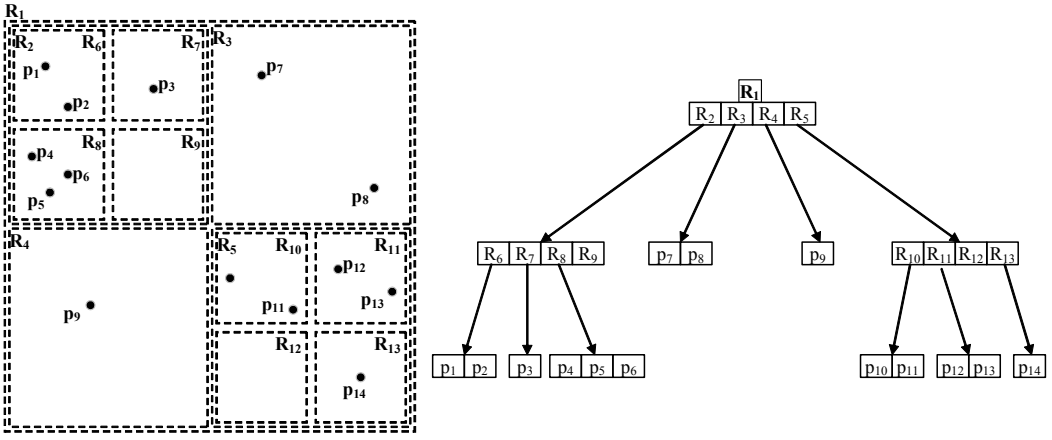
Omawiana tu wersja STCAT posiada agregaty w każdym węźle drzewa przestrzennego oraz w węzłach drzew czasowych. Indeks ten jest wykorzystany w będącym motywacją tej pracy systemie przestrzennej hurtowni danych telemetrycznych służącym do zdalnego zbierania pomiarów z liczników mediów (wody, prądu, gazu) wyposażonych w interfejs komunikacji bezprzewodowej GSM/GPRS. W liściach drzewa przestrzennego przechowywane są liczniki o ustalonych współrzędnych w przestrzeni dwu- lub trójwymiarowej. Każdemu licznikowi odpowiada drzewo czasowe przechowujące jego pomiary.

2.1. Wielowymiarowe drzewo przestrzenne

Drzewo przestrzenne STCAT jest wielowymiarowym drzewem, które w każdym węźle pośrednim dzieli obszar równomiernie na p_n^d części, gdzie p_n jest liczbą części podziału wzdłuż pojedynczego wymiaru, a *d* jest liczbą wymiarów. Obszar korzenia odpowiada całej przestrzeni danych. Parametrami drzewa przestrzennego indeksu STCAT są:

- liczba części podziału wzdłuż pojedynczego wymiaru (p_n) – decyduje o liczbie podwęzłów, jaką mogą zawierać węzły pośrednie, która wynosi p_n^d, gdzie d jest liczbą wymiarów,
- pojemność węzłów-liści (c_l)– maksymalna liczba wpisów (liczników, korzeni drzew czasowych), jaką mogą zawierać węzły-liście.

Rysunek 2 prezentuje przykładowe drzewo przestrzenne dla przestrzeni dwuwymiarowej.



Rys. 2. Drzewo przestrzenne STCAT dla przykładowych danych punktowych 2d
Fig. 2. Example of spatial STCAT tree

Drzewo posiada możliwość rozbudowywania obszaru podczas wstawiania nowych danych. Gdy współrzędne nowego punktu (licznika) przekraczają obszar drzewa (korzenia), to dokłada się na górę nowy korzeń, który posiada podwojony zakres w każdym z wymiarów, tak by obejmował nowy punkt.

2.2. Drzewo czasowe

Część czasowa indeksu STCAT jest jednowymiarową odmianą części przestrzennej tego indeksu. Każdy węzeł pośredni dzieli swój zakres równomiernie na *r* części przyporządkowanych synom.

Parametrami drzewa czasowego indeksu STCAT są:

- liczba części podziału węzła pośredniego (r_n),
- pojemność węzłów-liści (r_l) – maksymalna liczba wpisów (danych), jaką mogą zawierać węzły-liście,
- początkowy rozmiar korzenia – w przeciwieństwie do części przestrzennej nieznany jest tu początkowy zakres drzewa. Dlatego jest on podawany tym parametrem i każde nowe drzewo czasowe jest tworzone z takim początkowym zakresem czasu,
- minimalny zakres (rozmiar) węzła – określa dolną granicę rozmiaru węzła, po przekroczeniu której węzeł nie jest dzielony na podwęzły, a liczba jego wpisów może przekraczać pojemność węzłów.

2.3. Zapytania na indeksie STCAT

Ze względu na to, że drzewo czasowe STCAT możemy traktować jak jednowymiarową wersję drzewa przestrzennego, to sposób realizacji zapytań dla nich obu jest podobny. Poniżej omówiono krótko trzy podstawowe typy zapytań wspierane przez indeks STCAT.

2.3.1. Zapytanie zakresowe

Zapytanie zakresowe (ang. *Range Query*) zadajemy, gdy interesują nas wartości wszystkich danych z określonego obszaru i dla zadanego przedziału czasu. Zakładamy, że obszarem przestrzennym zapytania może być jedynie hiperprostopadłościan, a interwał czasu definiujemy, podając czas początkowy i końcowy interesującego nas przedziału. Chcąc wykonać zapytanie na obszarze przestrzennym o kształcie innym niż hiperprostopadłościan, musimy przybliżyć go serią hiperprostopadłościanów, a następnie dla każdego z nich zadać osobne zapytanie. Zapytanie to odpowiada warunkom filtracji frazy WHERE języka SQL na wymiarach przestrzennym i czasowym. Realizowane jest ono tak, że rozpoczynając od korzenia i schodząc w dół drzewa, w każdym węźle odrzucane są podwęzły nieprzecinające zakresu zapytania (dzięki czemu znacznie ograniczamy przestrzeń poszukiwań i czas realizacji), a zapytanie jest przekazywane do tych węzłów, które przecinają zapytanie, po czym następuje scalanie wyników cząstkowych podwęzłów, a wynik jest przekazywany w górę.

2.3.2. Zapytanie agregacyjne

Zapytanie agregacyjne (ang. *Aggregate Range Query*) zadajemy, gdy nie interesują nas wartości poszczególnych pomiarów z zadanego obszaru i zakresu czasowego, a jedynie ich podsumowanie w postaci liczby pomiarów, ich sumy, wartości minimalnej, maksymalnej lub średniej. Zapytanie to odpowiada frazom COUNT, MIN, MAX, SUM, AVG stosowanym w zapytaniu zakresowym języka SQL. W obecnej wersji STCAT obsługiwane są jedynie wymienione powyżej podzielne funkcje agregacyjne (nie ma wsparcia dla funkcji holistycznych, jak np.: MEDIANA). Ich realizacja przebiega podobnie do zapytań zakresowych, z tą różnicą że nie musimy wchodzić do węzłów, których obszar całkowicie mieści się w zakresie zapytania, a odczytujemy jedynie przechowywane w nich wartości agregatów. Dzięki temu redukujemy znacznie liczbę dostępów do węzłów drzewa.

2.3.3. Zapytanie o k najbliższych sąsiadów

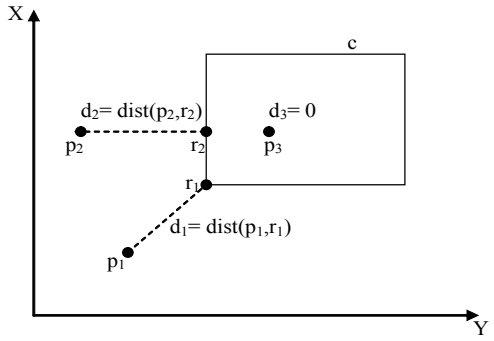
Zapytanie o *k* najbliższych sąsiadów (*k*NN) dotyczy jedynie przestrzennej części indeksu. Zadajemy je, gdy chcemy otrzymać zbiór *k* najbliższych obiektów względem punktu będącego parametrem zapytania. Zapytanie realizowane jest metodą „best fit". Jest to najwydajniejszy i najbardziej uniwersalny algorytm ze względu na to, że umieszcza już przeglądnięte wpisy (zarówno węzły, jak i wyniki) na liście posortowanej według ich odległości od punktu zapytania, dzięki czemu gwarantuje, co najwyżej jednokrotne odwiedzanie każdego węzła. W każdej iteracji pobierany jest wpis z początku listy (najbliższy punktowi zapytania), gdy jest to licznik, to dodajemy go do wyniku końcowego, gdy węzeł, to przeglądane są jego wpisy i umieszczane na odpowiednich pozycjach listy. Przetwarzanie jest kontynuowane do czasu uzyskania *k* wyników lub wyczerpania wszystkich elementów zbioru. Poniżej zaprezentowano algorytm realizacji zapytania *k*NN:

```
FUNCTION KNNQUERY
INPUT: punkt p, liczba wyników k, drzewo tree
OUTPUT: lista wynikowa res
lista sList;                  // posortowana według odległości od p lista wpisów
wpis node := tree.root;       // aktualnie przetwarzany wpis
WHILE res.count < k AND res.count < root.count AND node <> NULL DO
   IF node jest węzłem pośrednim THEN
         dodajemy wszystkie niepuste wpisy podległe node na listę sList;
   ELSE IF node jest węzłem-liściem THEN
         dodajemy wszystkie liczniki podległe node na listę sList;
   ELSE                       // node jest licznikiem
         dodajemy licznik do listy wynikowej res;
   END-IF
   node := sList.first;       // Zdejmujemy pierwszy wpis z listy sList
END-WHILE
RETURN res;
```

Zakładamy, że odległość dwóch punktów jest wyznaczana metryką euklidesową. Natomiast odległość pomiędzy punktem *p* a hiperprostopadłościanem *c* jest obliczana algorytmem MINDIST, który do wyznaczenia odległości wybiera z hiperprostopadłościanu *c* punkt leżący

najbliżej punktu *p*, a następnie zwraca ich wzajemną odległość. Jego działanie prezentuje poniższy pseudokod oraz rys. 3.

```
FUNCTION MINDIST
INPUT: punkt p, hiperprostopadłościan c
OUTPUT: odległość d
punkt r := p;    // r jest wyznaczanym punktem z regionu c najbliższym p
FOR każdy wymiar przestrzeni i DO
   IF p.pos[i] <= c.pos[i] THEN
          r.pos[i] := c.pos[i];
   ELSE IF p.pos[i] >= (c.pos[i] + c.size[i]) THEN
          r.pos[i] := c.pos[i] + c.size[i];
   ELSE
          r.pos[i] := p.pos[i];//Gdy p wewnątrz c to odległość = 0
   END-IF
END-FOR
RETURN d := dist(p,r); // Euklidesowa Odległość punktów
```



Rys. 3. Wyznaczanie odległosci algorytmem MINDIST (przykład dla przestrzeni 2d)
Fig. 3. Example of spatial STCAT tree

3. Modele kosztowe zapytań

Modele kosztowe wykorzystują wiedzę na temat budowy zbiorów danych w celu predykcji liczby dostępów do węzłów podczas realizacji zapytań, zanim jeszcze dana struktura zostanie stworzona. Modele analityczne mogą zostać wykorzystane na trzy sposoby:

- pozwalają lepiej zrozumieć zachowanie struktur danych dla różnorodnych wejściowych zbiorów danych o różnych rozmiarach,
- stanowią obiektywny sposób porównywania indeksujących struktur danych,
- mogą zostać użyte przez optymalizatory zapytań do utworzenia najlepszego planu realizacji złożonych zapytań.

Dobry model kosztowy indeksu drzewiastego pozwala przewidzieć liczbę dostępów do węzłów (a tym samym kosztownych czasowo odczytów z dysku) ze średnim błędem mniejszym niż 15% bez potrzeby realizacji zapytania, a jedynie opierając się na znajomości parametrów drzewa oraz ogólnej wiedzy o danych. Podczas analizy kosztów realizacji zapytań wygodnie jest założyć jednostkową przestrzeń danych $[0,1]^d$, gdyż pozwala to

sprowadzić rozmiary obiektów przestrzennych wprost do prawdopodobieństw. Tabela 1 opisuje główne symbole wykorzystywane w dalszej analizie.

Tabela 1

Główne symbole wykorzystywane w analizie

Symbol	Opis
p_n	liczba części podziału węzła pośredniego drzewa przestrzennego wzdłuż pojedynczego wymiaru
c_l	pojemność liści drzewa przestrzennego
d	liczba wymiarów przestrzeni
b	pojemność węzłów pośrednich drzewa przestrzennego
h_p	wysokość drzewa przestrzennego
n_l	liczba liści drzewa przestrzennego
N_j	liczba węzłów na *j*-tym poziomie drzewa przestrzennego
S_j	średni rozmiar węzłów na *j*-tym poziomie drzewa przestrzennego (we wszystkich wymiarach)

Analiza mająca na celu stworzenie modeli kosztowych zapytań dla STCAT będzie zaprezentowana dla jego przestrzennej wielowymiarowej części i oparta jest na analizie głównie dla R-drzew przedstawionej w publikacjach [5, 6, 7, 8, 9]. Dla części czasowej rozumowanie jest podobne, a nawet uproszczone, gdyż dotyczy tylko jednego wymiaru, którym jest czas.

Znając liczbę węzłów M drzewa oraz ich rozmiary L_j, możemy wyznaczyć liczbę dostępów do węzłów drzewa przestrzennego podczas realizacji zapytania q z poniższej zależności [8]:

$$NA(q) = \sum_{j=1}^{M} P_C(L_j, q) \tag{1}$$

gdzie: P_C jest prawdopodobieństwem tego, że podczas realizacji zapytania zostanie zrealizowane odwołanie do danego węzła.

W praktyce, nie możemy znać dokładnie liczby węzłów, a tym bardziej ich rozmiarów, zanim nie zbudujemy drzewa, dlatego też przybliżamy rezultat opierając się na znajomości wysokości drzewa h, liczby N_j węzłów na j-tym poziomie (0 to poziom liści) oraz średnich rozmiarów S_j węzłów na każdym poziomie [5]:

$$NA(q) = \sum_{j=0}^{h-1} \left[N_j \cdot P_C(S_j, q) \right] \tag{2}$$

Znając dobrze strukturę drzewa oraz stosowane algorytmy wartości h oraz N_j, można dość dokładnie estymować. Poniżej zostaną przedstawione wyprowadzenia formuł modeli kosztowych dla indeksu STCAT dla obsługiwanych rodzajów zapytań.

STCAT w części przestrzennej jest indeksem, który na każdym poziomie (w węźle) dzieli przestrzeń równomiernie na $b = p_n^d$ jednakowych części, gdzie p_n jest parametrem indeksu

określającym podział każdego wymiaru w węźle. Dlatego poniższe parametry łatwo wyznaczyć, traktując indeks jako równomierne drzewo *d*-wymiarowe.
Liczbę liści (poziom 0) wyznaczamy ze wzoru:

$$n_l = \frac{N}{c_l} \tag{3}$$

gdzie: N jest liczebnością zbioru danych, a c_l pojemnością liści. Wysokość drzewa obliczamy jako:

$$h = \lceil \log_b n_l \rceil \tag{4}$$

Liczba węzłów na *j*-tym poziomie:

$$N_j = b^{h-j-1} \tag{5}$$

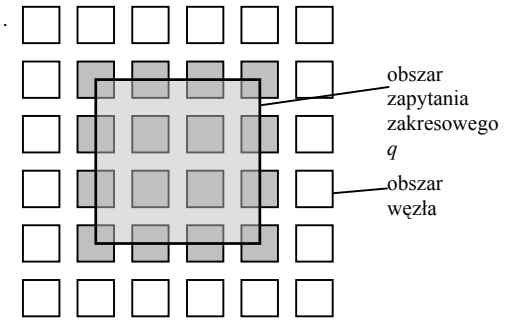
Średni rozmiar węzłów na *j*-tym poziomie (we wszystkich wymiarach, zakładając przestrzeń jednostkową):

$$S_j = \frac{1}{p_n^{\,h-j-1}} \tag{6}$$

Wyznaczone w ten sposób powyższe wartości wykorzystamy w formułach estymujących liczbę dostępów do węzłów podczas realizacji poszczególnych rodzajów zapytań.

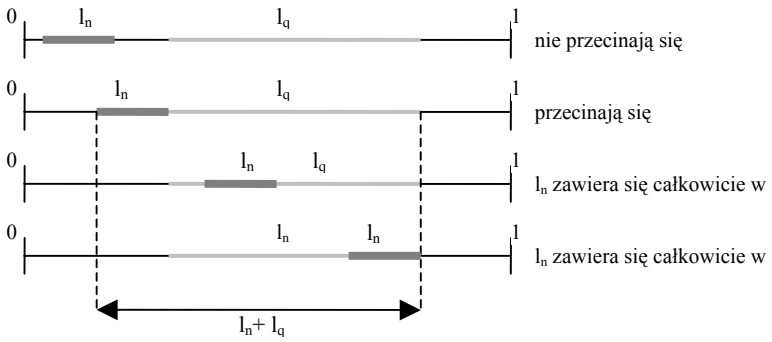
3.1. Przestrzenne zapytanie zakresowe

Dla zapytań zakresowych prawdopodobieństwo odwołania do węzła podczas realizacji zapytania P_C jest prawdopodobieństwem przecięcia obszaru węzła obszarem zapytania i będziemy je oznaczać jako P_{INTR}. Wynika to z tego, że odwołanie nastąpi do wszystkich węzłów, z którymi zapytanie się częściowo pokrywa lub które zawierają się w obszarze zapytania (rys. 4).



Rys. 4. Odwołania do węzłów (szare) podczas realizacji zapytania zakresowego
Fig. 4. Accessed nodes (grey) while executing range query

Aby wyznaczyć prawdopodobieństwa odwołania, rozważmy w jednostkowej przestrzeni jednowymiarowej możliwe położenia odcinka *q* reprezentującego obszar zapytania względem odcinka *l* reprezentującego obszar węzła, prezentuje to rys. 5.



Rys. 5. Wzajemne położenie odcinka zapytania i węzła w 1d przestrzeni jednostkowej (l_n oznacza długość odcinka węzła, a l_q oznacza długość odcinka zapytania)
Fig. 5. Mutual positions of query (q) and node segment (l) in 1d data space (ln and lq denote lengths of segments) for range query

Jak więc widać na powyższym rysunku, jeżeli pominiemy efekty graniczne, to dla jednego wymiaru prawdopodobieństwo przecięcia odcinka węzła odcinkiem zapytania jest równe sumie ich długości. Aby rozszerzyć powyższą analizę na dowolną liczbę wymiarów, wystarczy pomnożyć prawdopodobieństwa przecięcia w każdym z wymiarów, dzięki czemu otrzymujemy [5]. Gdy prawdopodobieństwo przecięcia dla któregoś z wymiarów przekroczy wartość 1, należy ograniczyć je w tym wymiarze do jedności.

$$P_{INTR}(n,q) = \prod_{i=0}^{d-1}(l_{ni} + l_{qi}) \tag{7}$$

Podstawiając do powyższego wzoru w miejsce $n.l_i$ średnie rozmiary S_j węzłów na j-tym poziomie drzewa z równania (6) oraz całość podstawiając do formuły (2), otrzymujemy ostatecznie formułę na estymację liczby dostępów do węzłów drzewa przestrzennego podczas realizacji zapytania zakresowego:

$$NA(q) = 1 + \sum_{j=0}^{\lceil \log_b n_l \rceil - 1} \left\{ b^{h-j-1} \cdot \prod_{i=0}^{d-1} \left(\frac{1}{p_n^{\,h-j-1}} + q.l_i \right) \right\} \tag{8}$$

Jedynka na początku formuły wynika z dostępu do korzenia drzewa, który musi być zawsze zrealizowany. Ponieważ końcowym celem jest również estymacja liczby dostępów do węzłów części czasowej, należy zmodyfikować powyższy model tak, aby umożliwiał estymacje osobno liczby dostępów do liści oraz do węzłów pośrednich. Modyfikacja sprowadziła się jedynie do tego, że liczba dostępów do liści odpowiada liczbie wyliczonej w pierwszej iteracji (dla poziomu j=0) i jest ona osobno zapamiętywana, natomiast suma uzyskana z pozostałych poziomów odpowiada liczbie dostępów do węzłów pośrednich.

Dla przykładu dokonamy estymacji liczby dostępów do węzłów podczas realizacji zapytania zakresowego dla indeksu STCAT o następujących parametrach:

- podział każdego wymiaru w węźle pośrednim $p_n = 5$,
- pojemność węzłów-liści $c_l = 5$,

- liczebność zbioru danych $N = 100000$,
- liczba wymiarów $d = 2$,
- zapytanie q o długości $q_{l1} = q_{l2} = 0.1$ w każdym z dwóch wymiarów.

Obliczamy:

Pojemność węzłów pośrednich $b = p_n{}^d = 5^2 = 25$

Liczba liści: $n_l = \frac{N}{c_l} = \frac{100000}{5} = 20000$

Wysokość drzewa: $h = \lfloor \log_{b_n} n_l \rfloor = \lceil \log_{25} 20000 \rceil = 5$

Węzły mają jednakowe rozmiary we wszystkich wymiarach, więc $l_{j0} = l_{j1}$ dla wszystkich poziomów j.

Tabela 2

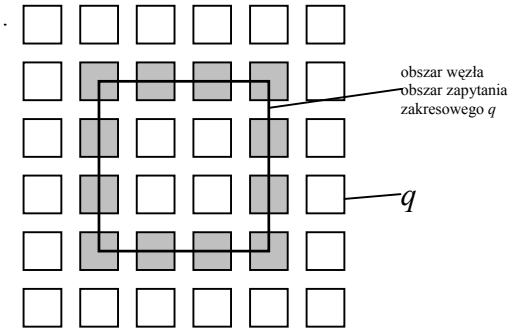
Przykład wyznaczania liczby dostępów do węzłów dla zapytania zakresowego

Poziom j	$N_j = b_n{}^{h-j-1}$	Wymiar i	$l_{j,i} = S_j = \frac{1}{p_n{}^{h-j-1}}$	q_{li}	$P_{PART}(n,q) = \prod_{i=0}^{d-1}(q.l_i + n.l_i)$	$NA(j) = N_j \cdot P_{PART}(j)$
0	15625	0	0,008	0,1	0,011664	**182** (liście)
		1	0,008	0,1		
1	625	0	0,04	0,1	0,0196	12,25
		1	0,04	0,1		
2	25	0	0,2	0,1	0,09	2,25
		1	0,2	0,1		
3	1	0	1	0,1	1,21 -> 1,0	1
		1	1	0,1		
					$NA(q) = \sum_{j=0}^{h-1}[NA(j)]$	**16** (w. pośrednie)

Gdy prawdopodobieństwo PPART(j) przekracza wartość 1.0, to zastępujemy je liczbą 1.0. Ostateczny wynik estymacji to 182 dostępy do węzłów-liści oraz 16 dostępów do węzłów pośrednich daje 198 dostępów do węzłów ogółem.

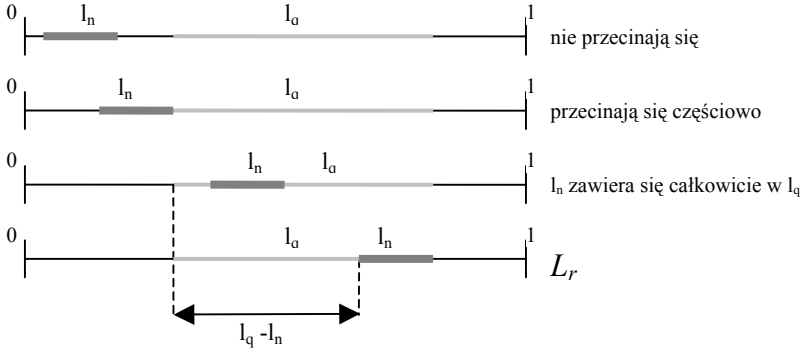
3.2. Przestrzenne zapytanie agregacyjne

Dla zapytań agregacyjnych prawdopodobieństwo odwołania do węzła podczas realizacji zapytania P_C jest prawdopodobieństwem częściowego przecięcia obszaru węzła obszarem zapytania (ale nie całkowitego zawierania się węzła w obszarze zapytania). Będziemy je oznaczać jako P_{PART}. Odwołanie nastąpi do wszystkich węzłów, z którymi zapytanie się częściowo pokrywa, ale nie do tych, które zawierają się w obszarze zapytania (rys. 6). Analiza ta zbliżona jest do tej prezentowanej w publikacji [5].



Rys. 6. Odwołania do węzłów (szare) podczas realizacji zapytania agregacyjnego
Fig. 6. Accessed nodes (grey) while executing aggregate range query

Aby wyznaczyć prawdopodobieństwa odwołania, podobnie jak w poprzednim przypadku, rozważmy w jednostkowej przestrzeni jednowymiarowej możliwe położenia odcinka q reprezentującego obszar zapytania agregacyjnego względem odcinka l reprezentującego obszar węzła, dla przypadku gdy zawiera się on całkowicie w obszarze zapytania.



Rys. 7. Wzajemne położenie odcinka zapytania i węzła w 1D przestrzeni jednostkowej (l_n oznacza długość odcinka węzła, a l_q oznacza długość odcinka zapytania)
Fig. 7. Mutual positions of query (q) and node segment (l) in 1d data space (l_n and l_q denote lengths of segments) for aggregate range query

Dla jednego wymiaru prawdopodobieństwo zawierania się odcinka węzła w odcinku zapytania jest równe różnicy ich długości. Aby rozszerzyć powyższą analizę na dowolną liczbę wymiarów, wystarczy pomnożyć prawdopodobieństwa przecięcia w każdym z wymiarów, dzięki czemu otrzymujemy [9]:

$$P_{CONT}(n,q) = \prod_{i=0}^{d-1} (l_{qi} - l_{ni}) \tag{9}$$

Dostęp następuje jedynie do tych węzłów, które częściowo się przecinają z zapytaniem [9]:

$$P_{PART}(n,q) = P_{INTR}(n,q) - P_{CONT}(n,q) \tag{10}$$

stąd po podstawieniu otrzymujemy [9]:

$$P_{PART}(n,q) = \prod_{i=0}^{d-1}(l_{qi} + l_{ni}) - \prod_{i=0}^{d-1}(l_{qi} - l_{ni}) \tag{11}$$

Podobnie jak dla zapytań zakresowych, podstawiając do powyższego wzoru w miejsce $n.l_i$ średnie rozmiary S_j węzłów na j-tym poziomie drzewa z równania (6) oraz całość podstawiając do formuły (2), otrzymujemy ostatecznie formułę na estymację liczby dostępów do węzłów drzewa przestrzennego podczas realizacji zapytania agregacyjnego:

$$NA(q) = \sum_{j=0}^{\lfloor \log_f n_l \rfloor - 1} \left\lceil b^{h-j-1} \cdot \left(\prod_{i=0}^{d-1}(q.l_i + \frac{1}{p_n{}^{h-j-1}}) - \prod_{i=0}^{d-1}(q.l_i - \frac{1}{p_n{}^{h-j-1}}) \right) \right\rceil \tag{12}$$

3.3. Zapytanie kNN

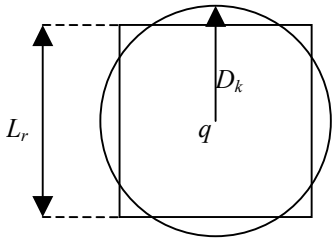
Wyprowadzenia zawarte w tej sekcji zostały zaczerpnięte z [7], gdzie autorzy zaprezentowali ulepszony model zapytań *k*NN dla R-drzewa. W przypadku zapytań *k*NN mamy do czynienia z pewnym punktem centralnym zapytania q oraz nieznanym z góry otaczającym go obszarem będącym hiperkulą $\Theta(q, D_k)$ o promieniu równym odległości D_k, będącej odległością punktu q do punktu k-tego sąsiada liczonej według określonej metryki (w dalszej analizie przyjmiemy metrykę euklidesową) a więc obszarem zawierającym k sąsiadów punktu q. Zakładając jednostkową przestrzeń danych oraz ich równomierny rozkład, prawdziwa jest zależność:

$$Vol(\Theta(q, D_k)) = k / N \tag{13}$$

gdzie: $Vol(\Theta(q, D_k))$ oznacza objętość hiperkuli zapytania, k jest liczbą najbliższych sąsiadów, a N jest liczbą wszystkich obiektów. Uwzględniając to, że obszar ten może częściowo wychodzić poza obszar przestrzeni danych U, jego wyznaczenie jest dość kłopotliwe, gdyż wymaga obliczenia średniej wartości jego objętości ze wszystkich możliwych pozycji punktu zapytania w przestrzeni U, co sprowadza się do czasochłonnego wyznaczenia wartości poniższej całki:

$$E[Vol(\Theta(q, D_k) \cap U)] = \int_{p \in U} Vol(\Theta(p, D_k) \cap U) dp \tag{14}$$

Aby uprościć obliczenia, przybliżamy obszar hiperkuli $\Theta(q, D_k)$ hipersześcianem $R_V(q, L_r)$, którego środek ciężkości q pokrywa się ze środkiem hiperkuli, a jego rozmiary są takie, aby posiadał on objętość równą objętości hiperkuli (rys. 8).



Rys. 8. Zastąpienie hiperkuli hipersześcianem o jednakowej objętości (na przykładzie 2d)
Fig. 8. Replacing hyper-sphere by hyper-rectangle in 2d

Objętość hiperkuli $\Theta(q, D_k)$ można wyznaczyć z zależności:

$$Vol(\Theta(q, D_k)) = \frac{\sqrt{\pi^d}}{\Gamma(d/2+1)} \cdot D_k^d \tag{15}$$

gdzie: $\Gamma(x)$ jest funkcją gamma ($\Gamma(x+1) = x \cdot \Gamma(x),\ \Gamma(1) = 1,\ \Gamma(1/2) = \pi/2$), a d oznacza liczbę wymiarów przestrzeni. Natomiast objętość hipersześcianu $R_V(q, L_r)$ obliczamy jako:

$$Vol(R_V(q, L_r)) = {L_r}^d \tag{16}$$

Przyrównując oba powyższe wzory (13) i (16), możemy łatwo wyznaczyć rozmiar krawędzi hipersześcianu:

$$\begin{aligned} &Vol(\Theta(q, D_k)) = Vol(R_V(q, L_r)) \\ &\frac{\sqrt{\pi^d}}{\Gamma(d/2+1)} \cdot D_k^d = {L_r}^d \\ &L_r = \frac{\sqrt{\pi}}{[\Gamma(d/2+1)]^{1/d}} \cdot D_k \end{aligned} \tag{17}$$

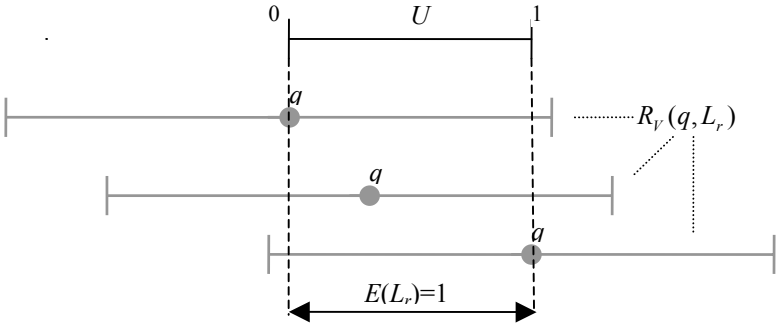
lub oznaczając stałą część równania jako C_v:

$$L_r = C_V \cdot D_k \quad \text{gdzie} \quad C_V = \frac{\sqrt{\pi}}{[\Gamma(d/2+1)]^{1/d}} \tag{18}$$

Obliczenie średniej wartości objętości ze wszystkich możliwych pozycji punktu zapytania w przestrzeni U dla obszaru $R_V(q, L_r)$ nie wymaga już kosztownego całkowania po całej przestrzeni U. Całkę

$$E[Vol(R_V(q, L_r) \cap U)] = \int_{p \in U} Vol(R_V(p, L_r) \cap U) dp \tag{19}$$

można w odróżnieniu do (14) obliczyć stosując następującą analizę. Na początek rozważmy przykład przestrzeni jednowymiarowej. Mając odcinek reprezentujący przestrzeń U o długości 1 oraz odcinek reprezentujący długość obszaru $R_V(q, L_r)$, można wyróżnić dwa przypadki. Pierwszy, gdy $L_r \geq 2$, wtedy bez względu na położenie odcinka $R_V(q, L_r)$ w przestrzeni, jego długość zawsze będzie ograniczona do wartości równej 1 (rys. 9).

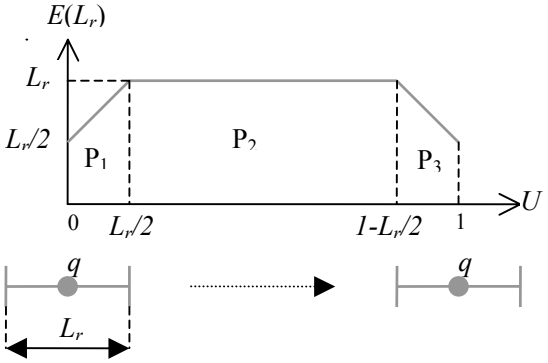


Rys. 9. Wzajemne położenie odcinka $R_V(q, L_r)$ o długości R_V oraz odcinka przestrzeni U dla przestrzeni 1D, gdy $L_r \geq 2$

Fig. 9. Position of $R_V(q, L_r)$ and U in 2d when $L_r \geq 2$

Dla drugiego przypadku ($L_r < 2$) oczekiwana wartość długości obszaru $R_V(q, L_r)$ zmienia się wraz z położeniem środka q odcinka od wartości $L_r/2$, gdy q znajduje się w punkcie 0, następnie rośnie aż do osiągnięcia wartości L_r gdy q znajdzie się w położeniu odpowiadającym $L_r/2$, pozostaje stała aż q osiągnie położenie 1- $L_r/2$, po czym spada do wartości $L_r/2$, gdy q ostatecznie pokrywa się z punktem 1. Prezentuje to rys. 10. Wartość oczekiwana długości $E(L_r)$ odpowiada polu pod wykresem podzielonemu przez długość przedziału, która wynosi 1. Pole to możemy obliczyć dzieląc je na składowe:

$$E[Vol_{1D}(R_V(q,L_r)\cap U)] = P_C = P_1 + P_2 + P_3$$
$$= (1-L_r)\cdot L_r + 2\cdot\frac{(L_r/2+L_r)\cdot L_r/2}{2} = L_r - L_r{}^2/4 \tag{20}$$



Rys. 10. Wzajemne położenie odcinka $R_V(q, L_r)$ o długości R_V oraz odcinka przestrzeni U dla przestrzeni 1d gdy $L_r < 2$

Fig. 10. Position of $R_V(q, L_r)$ and U in 2d when $L_r < 2$

Dla d-wymiarowej przestrzeni oczekiwaną średnią objętość wyznaczamy jako:

$$avgVol(R) = \prod_{i=1}^{d} avgLen(R_i) \tag{21}$$

stąd po uwzględnieniu (5) do (15) oraz obu przypadków otrzymujemy:

$$E[Vol(\Theta(q,D_k)\cap U)] \approx \begin{cases} (L_r - L_r{}^2/4)^d = (C_V \cdot D_k - C_V^2 \cdot D_k^2/4)^d, & L_r < 2 \\ 1 & otherwise \end{cases} \tag{22}$$

Gdy przyrównamy powyższy wynik do (13), otrzymamy:

$$Vol(\Theta(q,D_k)) = k/N = (C_V \cdot D_k - C_V^2 \cdot D_k^2/4)^d$$

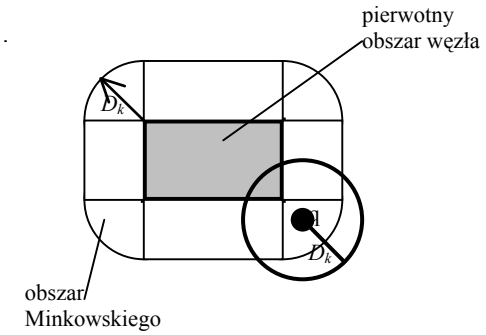
skąd ostatecznie odległość D_k, pomiędzy punktem zapytania a k-tym sąsiadem estymujemy jako:

$$D_K \approx \frac{2}{C_V}\left[1 - \sqrt{1 - \left(\frac{k}{N}\right)^{1/d}}\right] \tag{23}$$

Znając odległość D_k możemy przystąpić do estymacji liczby dostępów do węzłów drzewa. Przypomnijmy wzór na średnie rozmiary węzłów na j-tym poziomie (6):

$$S_j = \frac{1}{p_n{}^{h-j-1}} \tag{24}$$

Prawdopodobieństwo tego, że obszar M węzła przecina obszar $\Theta(q,D_k)$ możemy zastąpić prawdopodobieństwem tego, że obszar Minkowskiego $\Xi(q,D_k)$ (powstały poprzez powiększenie obszaru węzła o D_k jak na rysunku 11) zawiera punkt centralny zapytania q.



Rys. 11. Obszar Minkowskiego dla obszarów węzła
Fig. 11. Minkowski region

Jednak obliczenie prawdopodobieństwa dla obszaru Minkowskiego wymaga sporego nakładu obliczeniowego, dlatego też tak utworzony obszar Minkowskiego przybliżamy za pomocą hipersześcianu (o jednakowym środku ciężkości i jednakowej objętości). Objętość obszaru Minkowskiego możemy wyznaczyć ze wzoru :

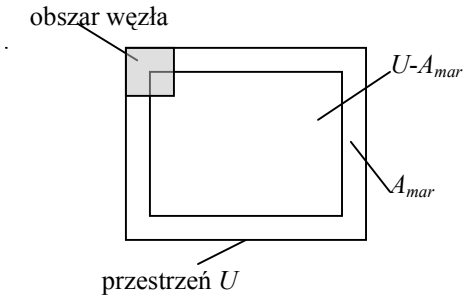
$$Vol(\Xi(q,D_k)) = \sum_{i=0}^{d}\left(\binom{d}{i} \cdot s_M^{d-i} \cdot \frac{\sqrt{\Pi^i}}{\Gamma(i/2+1)} \cdot D_K^i\right) \tag{25}$$

Przyrównując ją do objętości hipersześcianu wyznaczanej jako $Vol(\Xi(q,D_k)) = L_D^d$, możemy obliczyć rozmiary hipersześcianu w każdym z d wymiarów:

$$L_D = \left[\sum_{i=0}^{d} \left(\binom{d}{i} \cdot s_M^{d-i} \cdot \frac{\sqrt{\Pi^i}}{\Gamma(i/2+1)} \cdot D_K^i \right) \right]^{1/d} \tag{26}$$

gdzie s_M odpowiada rozmiarom węzła w każdym wymiarze (równanie (24)).

Prawdopodobieństwo dostępu P_{NAj} do węzła na j-tym poziomie podczas realizacji zapytania kNN odpowiada średniej wartości objętości obszaru $\Xi(q, D_k)$ dla wszystkich możliwych pozycji środka ciężkości węzła w obrębie przestrzeni U pomniejszonej o margines A_{mar} o szerokości $s_j/2$. Margines ten wynika z faktu, że pierwotne MBR węzłów muszą się w całości zawierać w U (rys. 12).

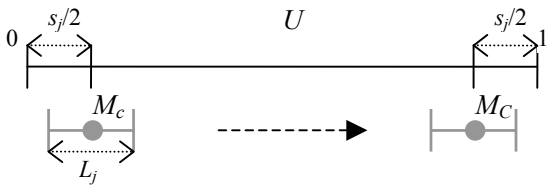


Rys. 12. Możliwe pozycje środka ciężkości węzła w obrębie przestrzeni U
Fig. 12. The query region localization in the U data space

Prawdopodobieństwo to wyznaczamy za pomocą całki:

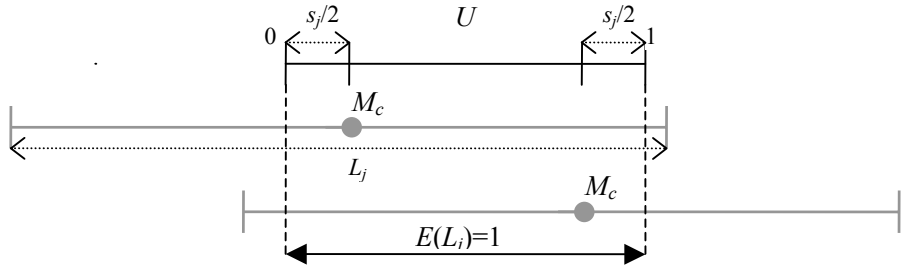
$$P_{NAj} = E[Vol(\Xi(q, D_k) \cap U)] \approx \int_{p \in U - A_{mar}} Vol(R_{MINK}(M, L_r) \cap U) dM_C$$

Zamiast rozwiązywać tę całkę numerycznie, można podobnie jak dla (19) obliczyć ją stosując poniższą analizę. Ponownie rozpoczniemy analizę od przestrzeni jednowymiarowej. Mając odcinek reprezentujący przestrzeń U o długości 1, odcinek reprezentujący długość obszaru MBR s_j oraz odcinek odpowiadający długości obszaru Minkowskiego L_j, można wyróżnić trzy przypadki. Pierwszy, gdy $s_j/2 \geq L_j/2$, czyli gdy bez względu na położenie odcinek obszaru $\Xi(q, D_k)$ zawsze mieści się w całości w U-A_{mar}, a więc jego długość zawsze wynosi L_j (rys. 13).



Rys. 13. Położenie odcinka $\Xi(q, D_k)$ o długości L_j w przestrzeni U-A_{mar} dla przestrzeni 1d, gdy $s_j/2 \geq L_j/2$
Fig. 13. Localization of segment $\Xi(q, D_k)$ (wchich width is L_j) in U-A_{mar} space in 1d for $s_j/2 \geq L_j/2$

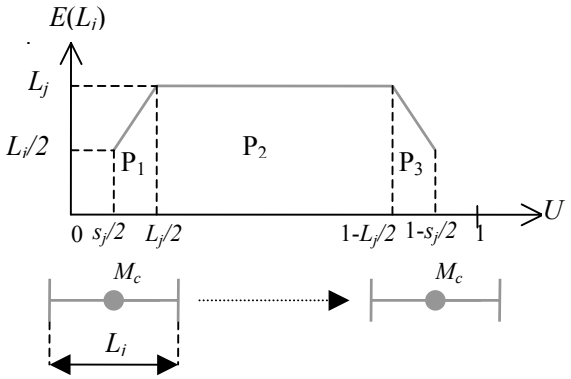
Drugi przypadek występuje, gdy $s_j/2 + L_j/2 \geq 1,$ czyli gdy bez względu na położenie odcinek obszaru $\Xi(q, D_k)$ zawsze zawiera obszar przestrzeni *U*, więc jego długość zawsze wynosi 1 (rys. 14).



Rys. 14. Położenie odcinka $\Xi(q, D_k)$ o długości L_j w przestrzeni U-A_{mar} dla przestrzeni 1d, gdy $s_j/2 + L_j/2 \geq 1$

Fig. 14. Localization of segment $\Xi(q, D_k)$ (wchich width is L_j) in U-A_{mar} space in 1d for $s_j/2 + L_j/2 \geq 1$

Dla trzeciego przypadku ($s_j < s_j/2 + L_j/2 < 1$) wynikowa długość obszaru $\Xi(q, D_k)$ zmienia się wraz z położeniem środka odcinka od wartości $s_j/2 + L_j/2$, gdy środek znajduje się w punkcie $s_j/2$, następnie rośnie aż do osiągnięcia wartości L_j, gdy środek znajdzie się w położeniu odpowiadającym $L_r/2$, pozostaje stała aż osiągnie on położenie 1- $L_j/2$, po czym spada do wartości $s_j/2 + L_j/2$, gdy środek ostatecznie pokrywa się z punktem 1-$s_j/2$. Prezentuje to rys. 15.



Rys. 15. Położenie odcinka $\Xi(q, D_k)$ o długości L_j w przestrzeni U-A_{mar} dla przestrzeni 1d, gdy $s_j < s_j/2 + L_j/2 < 1$

Fig. 15. Localization of segment $\Xi(q, D_k)$ (wchich width is L_j) in U-A_{mar} space in 1d for $s_j < s_j/2 + L_j/2 < 1$

Wartość oczekiwana długości $E(L_j)$ odpowiada polu pod wykresem podzielonemu przez długość przedziału, a więc $1 - s_j$. Pole to możemy obliczyć dzieląc je na składowe:

$$E[Vol_{1D}(\Xi(q,D_k)\cap U)] = \frac{P_C}{1-s_j} = \frac{P_1+P_2+P_3}{1-s_j}$$
$$= \frac{(1-L_j)\cdot L_j + 2\cdot\frac{(L_j+s_j+L_j/2)\cdot(L_j/2-s_j/2)}{2}}{1-s_j} = \frac{L_j-(L_j/2+s_j/2)^2}{1-s_j} \tag{27}$$

Ostatecznie po uwzględnieniu wszystkich przypadków i *d*-wymiarowej przestrzeni oczekiwaną średnią objętość wyznaczamy (korzystając z (21)) jako:

$$P_{NAj} = E[Vol_{1D}(\Xi(q,D_k)\cap U)] \approx \begin{cases} \left(\frac{L_j-(L_j/2+s_j/2)^2}{1-s_j}\right)^d, & L_j+s_j<2 \\ 1 & otherwise \end{cases} \tag{28}$$

Podstawiając do (2), liczbę dostępów do węzłów estymujemy jako:

$$NA(k) = \sum_{j=0}^{h-1}\left\{ b^{h-j-1}\cdot\left(\frac{L_j-(L_j/2+S_j/2)^2}{1-S_j}\right)^d\right\} \tag{29}$$

gdzie: L_j uzyskujemy z równania (26), a S_j z równania (24).

Podobnie jak dla poprzednich modeli, powyższy model można również zmodyfikować tak, aby umożliwiał estymacje osobno liczby dostępów do liści oraz do węzłów pośrednich. Modyfikacja sprowadza się jedynie do tego, że liczba dostępów do liści odpowiada liczbie wyliczonej w pierwszej iteracji (dla poziomu j=0) i jest ona osobno zapamiętywana, natomiast suma uzyskana z pozostałych poziomów odpowiada liczbie dostępów do węzłów pośrednich.

3.4. Zapytanie zakresowe przestrzenno-czasowe

Jak wspomniano wcześniej, drzewo czasowe indeksu STCAT możemy traktować jako jednowymiarową odmianę drzewa przestrzennego. Dlatego też modele kosztowe zapytań dla części czasowej są analogiczne do części przestrzennej. Jako przykład zaprezentowany zostanie model kosztowy zapytania zakresowego przestrzenno-czasowego. Model ten składa się z dwóch części: omówionej poprzednio estymacji liczby dostępów do liści drzewa przestrzennego, którą tu oznaczymy jako NA_{Sl}(q) oraz estymacji liczby dostępów do węzłów NA_T(q) drzew czasowych. W praktyce, bazując na otrzymanej liczbie dostępów do liści przestrzennych, określamy w przybliżeniu, na podstawie średniego ich zapełnienia *f*, liczbę liczników, których dotyczy zapytanie i mnożymy przez nią uzyskane z estymacji dla drzewa czasowego wartości określające liczby dostępów do węzłów czasowych w celu uzyskania całkowitej liczby dostępów do węzłów drzew czasowych oznaczonej jako TNA_T(q):

$$TNA_T(q) = NA_T(q)\cdot NA_{Sl}(q)\cdot f \tag{30}$$

Czasowa część indeksu STCAT jest jednowymiarowym odpowiednikiem przestrzennej części STCAT. Podstawową różnicą jest to, że o ile dla części przestrzennej znaliśmy obszar przestrzeni danych (zakładaliśmy, że jest to przestrzeń jednostkowa), to dla części czasowej nie jest on z góry znany. Musimy więc albo podać ten zakres czasu jako dodatkowy parametr estymacji, albo odczytać go ze zbudowanego indeksu, dla którego przeprowadzamy estymacje, o ile taki istnieje. Następnie wykorzystując ten przedział należy normalizować zakresy czasowe węzłów do przedziału [0; 1], aby odpowiadały bezpośrednio prawdopodobieństwu.

Podobnie jak poprzednio estymacja dla części czasowej również bazuje na formule (2):

$$NA_T(q) = \sum_{j=0}^{h-1} \left[N_j \cdot P_{INTR} \right]$$

Czasowa część STCAT jest indeksem, który na każdym poziomie (w węźle) dzieli zakres czasu równomiernie na b jednakowych części. Liczbę liści (poziom 0) wyznaczamy ze wzoru:

$$n_l = \frac{N}{b} \tag{31}$$

Wysokość drzewa obliczamy jako:

$$h = \lceil \log_b n_l \rceil + 1 \tag{32}$$

Liczba węzłów na *j*-tym poziomie:

$$N_j = b^{h-j-1} \tag{33}$$

Rozmiar węzłów na *j*-tym poziomie:

$$S_j = \frac{S_{tot}}{p_n^{h-j-1}} \tag{34}$$

gdzie: S_{tot} jest całkowitym zakresem czasu indeksu.

Prawdopodobieństwo przecięcia węzła zakresem zapytania zakresowego obliczamy poprzez sumę znormalizowanych zakresów:

$$P_{INTR} = \frac{s_j}{S_{tot}} + \frac{q_{range}}{S_{tot}} \tag{35}$$

gdzie: q_{range} jest zakresem czasowym zapytania. Ostatecznie podstawiając do (2) wartości z (33) i (35), otrzymujemy oczekiwaną liczbę dostępów do węzłów:

$$NA_T(q) = \sum_{j=0}^{h-1} \left[b^{h-j-1} \cdot \left(\frac{s_j}{S_{tot}} + \frac{q_{range}}{S_{tot}} \right) \right] \tag{36}$$

Ostatecznie całkowitą liczbę dostępów do wszystkich węzłów czasowych indeksu STCAT otrzymamy poprzez pomnożenie uzyskanej wartości przez wyznaczoną liczbę dostępów do liści części przestrzennej:

$$TNA_T(q) = NA_T(q) \cdot NA_{Sl}(q) = \sum_{j=0}^{h-1} \left[b^{h-j-1} \cdot \left(\frac{s_j}{S_{tot}} + \frac{q_{range}}{S_{tot}} \right) \right] \cdot NA_{Sl}(q) \cdot f \tag{37}$$

4. Testy

Testy były wykonywane na komputerze z procesorem Athlon XP 1800+ (1.54 GHz), 1.5 GB pamięci RAM pracującym pod kontrolą systemu Windows XP Professional SP2. Indeksy i ich modele były pisane w języku JAVA oraz skompilowane i uruchamiane na maszynie wirtualnej w wersji 1.6.02.

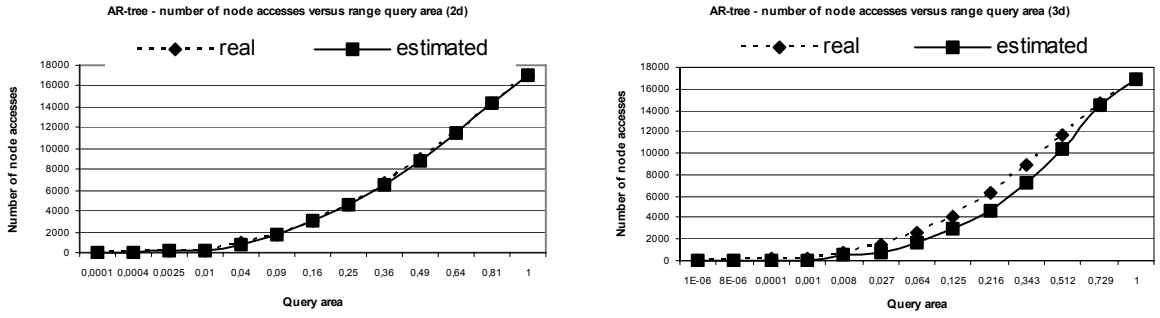
Aby ułatwić ocenę jakości opracowanych modeli dla indeksu STCAT, zaimplementowano modele dla aR-drzewa opierając się na publikacje [5, 6, 7].

Poniższe testy modeli kosztowych części przestrzennych indeksów były wykonywane na losowych zbiorach danych, zawierających 100 tys. liczników. Pojemność węzłów pośrednich oraz liści aR-drzewa była ustawiona na 10, a współczynnik minimalnego zapełnienia na 0.5. Dla indeksu STCAT parametr określający liczbę podziałów na wymiar wynosił dla węzłów 5, a pojemność liści została ustawiona na 10.

Dla każdego zestawu parametrów były zadawane dwa zapytania: jedno zakresowe oraz jedno o estymacje kosztu. Na wykresach przedstawiono porównanie całkowitej liczby dostępów (do węzłów i liści) wykonanych podczas realizacji zapytania z liczbą dostępów, będącą wynikiem estymacji.

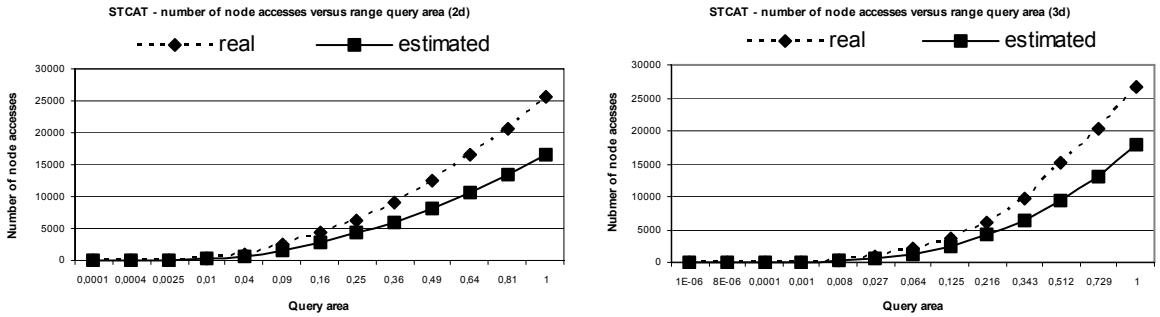
4.1. Przestrzenne zapytanie zakresowe

Pierwszy eksperyment ma na celu zbadanie dokładności modeli kosztowych zapytań zakresowych dla różnych obszarów zapytań. Rysunki 16 i 17 przedstawiają wyniki badań dla aR-drzewa oraz STCAT w przestrzeni 2- i 3-wymiarowej.



Rys. 16. Wyniki eksperymentu estymacji liczby dostępów do węzłów aR-drzewa w funkcji obszaru zapytania zakresowego dla przestrzeni 2d i 3d

Fig. 16. Range query cost evaluation versus query area in 2d and 3d data space for aR-tree



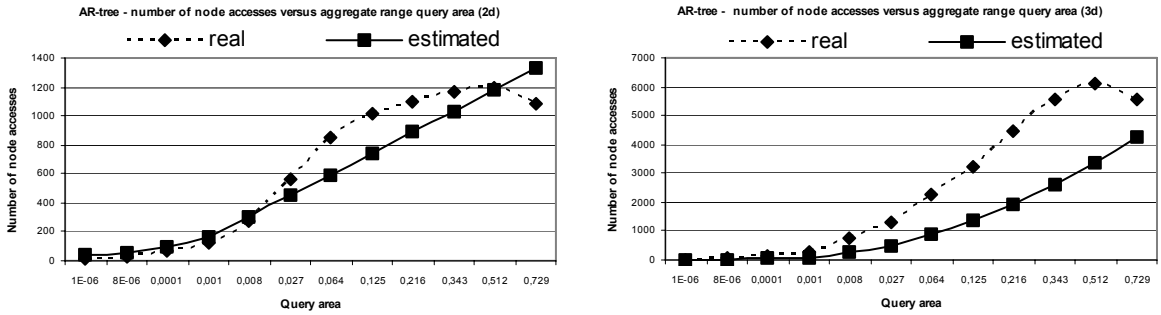
Rys. 17. Wyniki eksperymentu estymacji liczby dostępów do węzłów STCAT w funkcji obszaru zapytania zakresowego dla przestrzeni 2d i 3d

Fig. 17. Range query cost evaluation versus query area in 2d and 3d data space for STCAT

Dla zapytań zakresowych model kosztowy aR-drzewa jest dość precyzyjny. Model dla indeksu STCAT jest nieco mniej precyzyjny, zwłaszcza przy dużych zakresach przestrzennych zapytań. Wraz ze wzrostem liczby wymiarów przestrzeni estymacja staje się coraz mniej dokładna dla obu indeksów.

4.2. Przestrzenne zapytanie agregacyjne

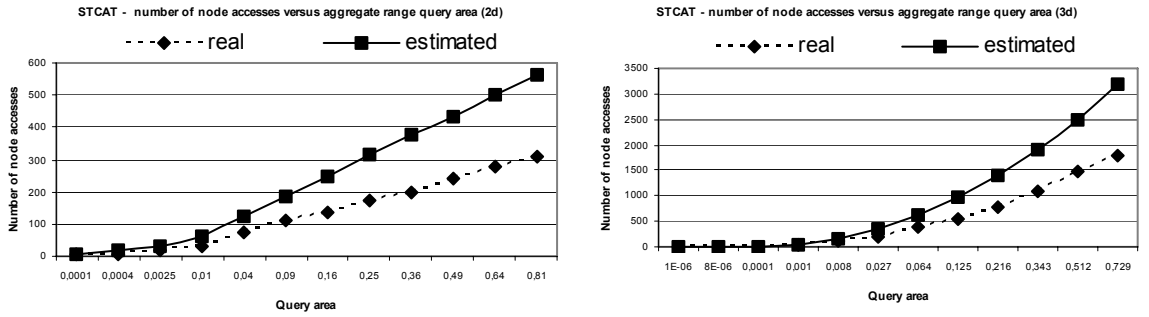
Drugi eksperyment bada dokładność modeli kosztowych dla zapytań agregacyjnych dla różnych obszarów zapytań. Rysunki 18 i 19 przedstawiają wyniki badań dla aR-drzewa i STCAT w przestrzeni 2- i 3-wymiarowej.



Rys. 18. Wyniki eksperymentu estymacji liczby dostępów do węzłów aR-drzewa w funkcji obszaru zapytania agregacyjnego dla przestrzeni 2d i 3d

Fig. 18. Aggregate range query cost evaluation versus query area in 2d and 3d data space for aR-tree

Dla zapytań agregacyjnych modele kosztowe zarówno aR-drzewa, jak i STCAT są mniej dokładne niż modele zapytań zakresowych. Model kosztowy aR-drzewa dokonuje przeszacowania liczby dostępów do węzłów, a model kosztowy STCAT – niedoszacowania.

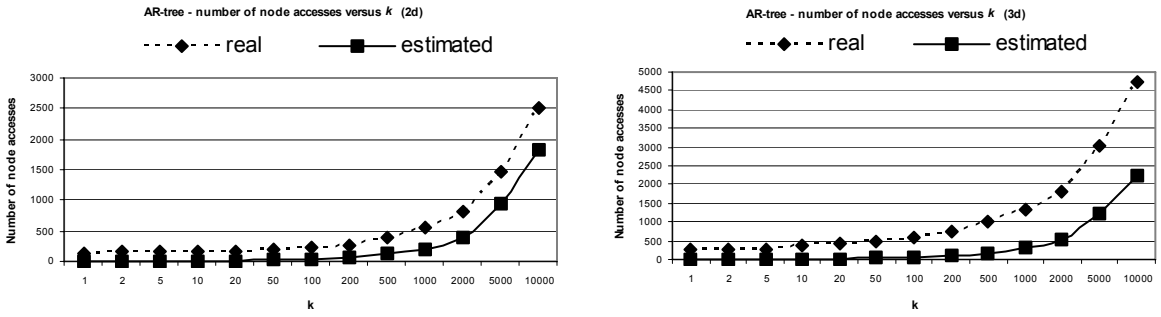


Rys. 19. Wyniki eksperymentu estymacji liczby dostępów do węzłów STCAT w funkcji obszaru zapytania agregacyjnego dla przestrzeni 2d i 3d

Fig. 19. Aggregate range query cost evaluation versus query area in 2d and 3d data space for STCAT

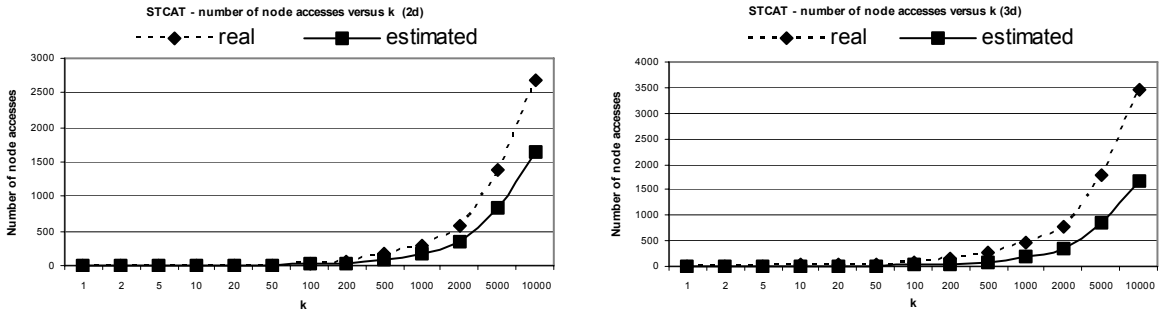
4.3. Przestrzenne zapytanie kNN

Kolejny eksperyment sprawdza dokładność modeli kosztowych dla zapytań o k najbliższych sąsiadów przy różnych wartościach parametru k. Rysunki 20 i 21 przedstawiają wyniki badań dla aR-drzewa i STCAT w przestrzeni 2- i 3-wymiarowej.



Rys. 20. Wyniki eksperymentu estymacji liczby dostępów do węzłów aR-drzewa w funkcji parametru k zapytań kNN dla przestrzeni 2d i 3d

Fig. 20. kNN query cost evaluation versus k in 2d and 3d data space for aR-tree



Rys. 21. Wyniki eksperymentu estymacji liczby dostępów do węzłów STCAT w funkcji parametru k zapytań kNN dla przestrzeni 2d i 3d

Fig. 21. kNN query cost evaluation versus k in 2d and 3d data space for STCAT

Modele kosztowe zapytań *k*NN są najbardziej skomplikowanymi z prezentowanych tu modeli. Podobnie jak modele dla zapytań agregacyjnych są one mniej dokładne niż modele zapytań zakresowych. Modele dla STCAT zapytań *k*NN wykazują większą dokładność niż modele dla aR-drzewa, zwłaszcza przy mniejszych wartościach parametru *k*.

5. Podsumowanie

Opierając się na wiedzy z prac [5, 6, 7], przeanalizowano i zaimplementowano modele kosztowe dla aR-drzewa estymujące liczbę dostępów do węzłów drzewa podczas realizacji zapytań zakresowych, agregacyjnych oraz o *k* najbliższych sąsiadów. Wzorując się na modelach dla aR-drzewa, zostały od podstaw opracowane i zaimplementowane modele kosztowe dla indeksu STCAT. Przeprowadzone testy porównujące liczbę rzeczywistych dostępów do węzłów obu indeksów z liczbą estymowaną za pomocą wdrożonych modeli kosztowych wykazały, że konieczne są dalsze badania, mające na celu zwiększenie dokładności modeli kosztowych zapytań dla indeksu STCAT. Jednak już obecnie modele te mogą być bardzo użyteczne w zastosowaniach, w których nie jest istotny dokładny wynik estymacji, a jedynie zgrubne oszacowanie kosztu realizacji zapytań w celu porównania różnych metod.

LITERATURA

1. Gorawski M., Gorawski J., M.: Balanced Spatio-Temporal Data Warehouse with R-MVB, STCAT and BITMAP Indexes PARELEC (2006) 5-th International Symposium on Parallel Computing in Electrical Engineering, Poland, IEEE Computer Society, s. 43÷48 (2006).
2. Gorawski M., Gorawski J.,M., Bańkowski S.: Selection of Indexing Structures in Grid Data Warehouses with Software Agents. International Journal of Computer Science & Applications, vol.4, No.1, s. 39÷52, ISSN 0972-9038 (2007).
3. Guttman A.: R-Trees. a Dynamic Index Structure For Spatial Searching, ACM SIGMOD Conference on Management of Data, s. 47÷57 (1984).
4. Simonas Saltenis, Christian S. Jensen : R-tree Based Indexing of General Spatio-Temporal Data, TimeCenter Technical Report (1999).
5. Yannis Theodoridis, Timos Sellis : A Model for the Prediction of R-tree Performance, ACM Conf. ISBN 0-89791-781-2, s. 161÷ 71 (1996).
6. C. Boehm : A cost Model for Query Processing in High Dimensional Data Spaces, ACM Conf. ISSN:0362-5915, s. 129÷178 (2000).

7. Yufei Tao, Jun Zhang, Dimitris Papadias, Nikos Mamoulis: An Efficient Cost Model for Optimization of Nearest Neighbor Search in Low and Medium Dimensional Spaces, IEEE TKDE, ISSN 1041-4347, s. 1169÷1184 (2004).
8. C. Faloutsos, I. Kamel : Beyound Uniformity and Indepedence: Analysis of R-trees Using the Concept of Fractal Dimension, ACM Conf. ISBN:0-89791-642-5, pp. 4÷13 (1994).
9. Yufei Tao, Dimitris Papadias, Jun Zhang : Aggregate Processing of Planar Points, EDBT, ISBN 3-540-43324-4, s. 682÷700 (2002).

Recenzent: Dr inż. Tomasz Koszlajda

Wpłynęło do Redakcji 3 października 2007 r.

Abstract

The paper proposes new cost models for STCAT index. The models lets us estimate number of node accesses during executing range, aggregate and *k*NN queries. Proposed cost models follow existing models for R-tree. Both STCAT and R-tree models was implemented, tested and compared. Performed tests confirm that models for STCAT are useful for assessing index efficiency and they can be used to choose between different indices.

Adresy

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