

Dariusz R. AUGUSTYN
Politechnika Śląska, Instytut Informatyki

SZACOWANIE SELEKTYWNOŚCI ZAPYTAŃ Z WYKORZYSTANIEM METOD ESTYMACJI JEDNOWYMIAROWYCH ROZKŁADÓW PRAWDPODOBIENSTWA

Streszczenie. Artykuł opisuje sposoby obliczania selektywności – parametru wyznaczanego w celu wstępnego oszacowania rozmiaru wyniku zapytania. Wartość tego parametru wykorzystywana jest przez optymalizator zapytań serwera bazy danych w procesie uzyskania możliwie najefektywniejszego sposobu realizacji zapytania. Artykuł prezentuje techniki wyznaczania selektywności, bazujące na histogramowej estymacji jednowymiarowego rozkładu wartości atrybutu.

Słowa kluczowe: optymalizacja planu wykonania zapytania, selektywność zapytań, nieparametryczna estymacja rozkładu wartości atrybutu, histogram equi-height, histogram equi-width, histogram skompresowany, histogram MaxDiff, histogram V-optymalny

THE QUERY SELECTIVITY APPROXIMATION USING ESTIMATION METHODS OF SINGLE-DIMENSIONAL PROBABILITY DISTRIBUTIONS

Summary. The paper presents methods of a selectivity value estimation, which is required for a preliminary approximation of a query result size. The selectivity value is used by database management system query optimizer for obtaining the most effective query execution plan. The paper shows selectivity approximation methods based on a histogram estimation of a single-dimensional attribute value distribution.

Keywords: query execution plan optimization, query selectivity, non-parametric estimation of a attribute value distribution, equi-height histogram, equi-width histogram, compressed histogram, MaxDiff histogram, V-optimal histogram

1. Selektowność w procesie wyznaczania planu realizacji zapytania – wprowadzenie

1.1. Model realizacji prostych zapytań SQL

Realizacja zapytania SQL – tzw. faza wykonania (ang. *execute phase*) poprzedzona jest fazą przygotowania zapytania (ang. *prepare phase*). Faza przygotowania obejmuje m.in.: odebranie zapytania przesłanego do serwera b. d., sprawdzenie syntaktyczne zdania wyrażonego w języku SQL, sprawdzenie istnienia odpowiednich obiektów bazy danych, do których zapytanie się odnosi (tablic, perspektyw, kolumn, funkcji i in.), opierając się na słowniku bazy danych (tzw. metaschemacie), kontrolę uprawnień dostępu do obiektów wynikających z kontekstu połączenia, w ramach którego zapytanie przesyłane do serwera b. d., podstawienie definicji obiektów pośrednich (perspektyw) i wyznaczenie optymalnego sposobu realizacji zapytania (również opierając się na danych pochodzących z metaschematu).

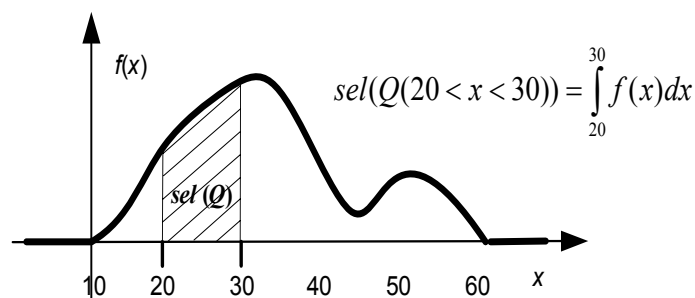
Za wykonanie tego ostatniego zadania, tzn. wyznaczenia możliwie najlepszego pod względem efektywnościowym planu realizacji, odpowiedzialny jest moduł optymalizatora w ramach programu serwera b. d. Optymalizator wyznacza kilka możliwych tzw. ścieżek realizacji zapytania SQL (ang. *access path*), pozwalających osiągnąć rezultat zapytania, przy różnym koszcie, mierzonym liczbą operacji: dyskowych, CPU, transferów sieciowych. Sposób realizacji zapytania SQL, wyrażony poprzez tzw. plan wykonania, może być dostępny do analizy dla użytkownika/administratora bazy danych wyposażonego w odpowiednie narzędzia programowe (na ogół specyficzne dla konkretnej implementacji serwera b. d.).

Podstawowym kryterium oceny efektywności sposobu realizacji zdań SQL (głównie zapytań SQL) jest ilość operacji pobrania/zapisu danych, mierzona liczbą pobrań/zapisów podstawowych jednostek alokacji danych w bazie danych, tzn. stron – bloków o stałym, predefiniowanym rozmiarze.

Sposób realizacji zapytania może znacząco się różnić. Przykładowo, w sytuacji prostego zapytania i obecności odpowiedniej struktury danych typu indeks będzie istnieć możliwość realizacji zapytania albo przez przegląd całej tablicy, albo dostęp od razu do właściwego elementu tablicy podczas przejścia przez drzewo indeksowe. Ten drugi sposób będzie na ogół sposobem efektywniejszym, chyba że wynik zapytania miałby dotyczyć wszystkich lub prawie wszystkich wierszy tablicy (wówczas pełny przegląd będzie efektywniejszy). Stąd wynika potrzeba wstępnego oszacowania rozmiaru rezultatu zapytania przez wyznaczenie parametru zwanego selektywnością.

1.2. Selektywność w ujęciu statystycznym

Pojęcie selektywności będzie wyjaśnione z wykorzystaniem prostej klasy zapytań SQL – zapytań z warunkiem na przedział danych (zakres), w odniesieniu do pojedynczej tablicy i do pojedynczego atrybutu tejże tablicy (ang. *range query with single-table predicate*). Przyjęto dodatkowo, że rozkład wartości atrybutu X można opisać przez $f(x)$ – funkcję gęstości rozkładu prawdopodobieństwa (f.g.p.). Przykładową f.g.p. pokazano na rys. 1.



Rys. 1. Obliczanie selektywności z użyciem funkcji gęstości prawdopodobieństwa
Fig. 1. Selectivity calculation for given probability density function

Selektywność $sel(Q)$ zapytania Q można opisać jako liczbę z przedziału $[0, 1]$, będącą stosunkiem liczby wierszy spełniających kryteria zapytania do liczby wszystkich wierszy tablicy. W ujęciu statystycznym selektywność oznacza prawdopodobieństwo wylosowania wiersza spełniającego kryterium zapytania ze zbioru wszystkich wierszy (rys. 1). Na przykład, w odniesieniu do rozpatrywanej klasy zapytań zakresowych $Q = Q(a < X < b)$, działających na atrybucie X relacji R selektywność zapytania wynosi: $sel(Q) = \int_a^b f(x) dx$.

Wynika stąd, że selektywność można oszacować korzystając z nieparametrycznych metod estymacji rozkładu prawdopodobieństwa, np. histogramów estymujących f.g.p.

1.3. Przykład wykorzystania selektywności w procesie wyznaczania kosztu realizacji zapytania

Logiczna budowa bazy danych umożliwia manipulowanie na tablicach, perspektywach, wierszach, natomiast fizyczna budowa zakłada jedynie manipulowanie na stronach bazy danych. Strona (blok) danych o stałym, określonym inicjalnie rozmiarze (np. będącym małą wielokrotnością 1KB) jest podstawową jednostką alokacji pamięci dyskowej dla serwera b.d.

Strony różnych typów przechowują różne rodzaje danych, np.: zawartości tablic (wierszy), fragmenty dużych obiektów binarnych (ang. BLOB), węzły drzew indeksowych (np. dla indeksów typu B^+ -tree) i inne. Ponieważ operacje dyskowe są operacjami najbardziej czasochłonnymi, koszt realizacji zapytania SQL jest głównie mierzony szacowaną liczbą pobrań stron bazy danych.

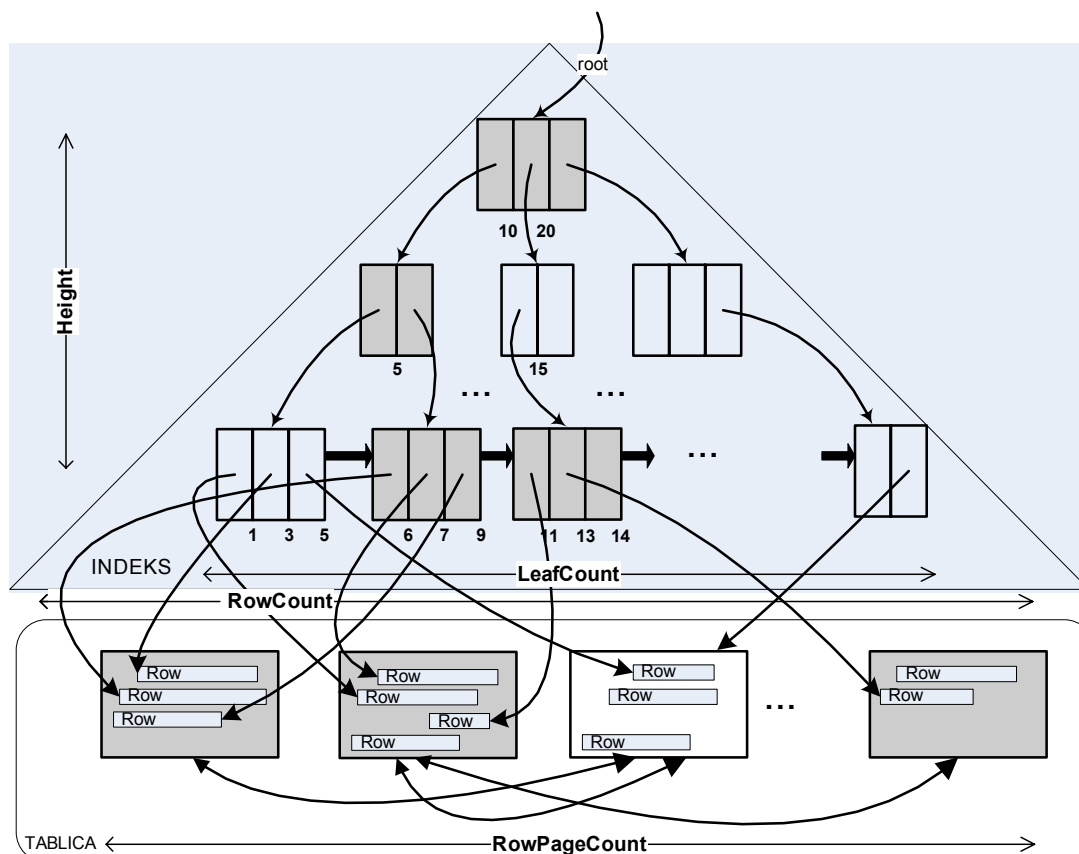
Wyznaczanie kosztu przykładowego zapytania SQL będzie zilustrowane na przykładzie sposobu działania optymalizatora serwera b. d. SQLBase firmy GUPTA Corp. [3], co nie zmienia ogólności przykładu. W implementacji SQLBase każda strona posiada swój unikalny numer strony (*PageNumber*) – odległość w bajtach liczona od początku fizycznego pliku bazy danych.

Wiersze każdej tablicy bazy danych przechowywane są na tzw. stronach danych (ang. *row page*). Wiersze tej samej tablicy mogą mieć różny rozmiar, stąd też liczba i położenie wierszy na stronie nie jest stałe. Każdy wiersz trafia do tzw. szczeliny (ang. *slot*), opisanej w nagłówku strony przez wartość przesunięcia początku i końca szczeliny, liczonych w bajtach od początku strony. Strony danych przechowują kolumny wierszy typów standardowych, kolumny typu BLOB są reprezentowane jako wskaźnik do listy innego rodzaju stron. Przy pewnych uproszczeniach (pomijających problem wersjonowania wierszy, ale nie wpływających na poprawność omawianych zagadnień), można stwierdzić, że każdy wiersz każdej tablicy posiada swój unikalny numer (*RowID*), będący złożeniem numeru strony (*PageNumber*) i numeru szczeliny na stronie (*SlotNumber*).

Tablica jest dwukierunkową listą stron danych. W metaschemacie bazy danych dla każdej tablicy zapisane są dane dotyczące liczby wierszy (*RowCount*) i liczby stron danych (*RowPageCount*), gdzie $RowPageCount \leq RowCount$.

Podstawową strukturą danych umożliwiającą efektywne wyszukiwanie lub porządkowanie danych jest struktura zrównoważonego drzewa indeksowego pokazana na rys. 2. Węzły drzewa indeksowego zawierają wartości kluczy i albo wskaźniki na węzły kolejnego, niższego poziomu drzewa, albo, w przypadku węzłów ostatniego poziomu (poziom liści), wskaźniki na wiersze tablicy zawierające wartości klucza. Drzewo indeksowe jest budowane na podstawie wartości wyrażenia indeksowego, którym w najprostszym przypadku są wartości pojedynczego atrybutu tablicy. Wartości kluczy występujące w indeksie dzielą dziedzinę wyrażenia indeksowanego na podprzedziały. Dla danej wartości klucza w poddrzewie po lewej stronie znajdują się węzły z wartościami mniejszymi lub równymi tej wartości klucza (w prawym poddrzewie występują węzły z wartościami kluczy większymi od danej wartości). Zrównoważenie drzewa (B-tree od ang. *balanced tree*) sprowadza się do zapewnienia przez proces serwera bazy danych własności stałej wysokości drzewa (*Height*). Zrównoważenie drzewa powoduje, że liczba kluczy w węźle (a stąd liczba wskaźników wyprowadzanych z węzła) może być różna dla każdego węzła. Jeżeli przyjąć średnią („po wszystkich węzłach”) liczbę wyprowadzeń z węzła jako d , to wysokość drzewa indeksowego można przybliżyć następująco $Height = \log_d(RowCount)$. Ostatni poziom węzłów-liści zawiera wszystkie możliwe wartości klucza. Węzły-liście są połączone w jednokierunkową listę o długości *LeafCount* (cecha indeksów typu B-plus). Wartości kluczy przeglądane wg tej listy są uporządkowane niemalejąco.

Fizycznie węzeł drzewa indeksowego jest zaimplementowany w postaci strony bazy danych (strona indeksowa). Węzły zwykle zawierają wskaźniki (*PageNumber*) na strony z innymi węzłami indeksowymi. Węzły-liście zawierają wskaźniki na strony danych, a dokładniej na konkretne wiersze tablic ($RowID = (PageNumber, SlotNumber)$)).



Rys. 2. Realizacja przykładowego, prostego, zakresowego zapytania SQL z wykorzystaniem indeksu typu B⁺-drzewo

Fig. 2. The example of realization of simple range SQL-query using the B⁺-tree index

Analiza możliwości wykorzystania drzewa indeksowego zostanie przeprowadzona na przykładzie rozważań dotyczących wyboru planu realizacji prostego zapytania Q_1 :

```
select Y from TABLICA where 6 ≤ X ≤ 13.
```

Przyjęto, że selektywność zapytania została oszacowana i jej wartość wynosi sel (Q_1).

Najprostszą metodą realizacji zapytania jest pełny przegląd tablicy (ang. *full table scan*), wykorzystujący fakt, że strony tablicy są połączone w listę. Koszt takiego sposobu wykonania, liczony liczbą pobranych stron, wynosi *RowPageCount*.

W przypadku użycia indeksu może nastąpić przejście od korzenia drzewa indeksowego do liścia na podstawie warunku dotyczącego lewego krańca zakresu ($6 \leq X$). Koszt operacji jest równy wysokości drzewa i wynosi *Height*. Korzystając z faktu uporządkowania kluczy w ramach listy węzłów-liści, może nastąpić przejście od lewej w kierunku prawej części listy (rys. 2). Przy przejściu, w ramach każdego węzła-liścia, dla każdego klucza w tym liściu

nastąpi sprawdzenie warunku prawego krańca ($X \leq 13$) i w przypadku jego spełnienia będzie zrealizowane przejście do odpowiedniej strony danych i wiersza (przez *RowID* związany z tym kluczem w liściu). Pierwsze niespełnienie warunku prawego krańca kończy przegląd listy węzłów-liści. Na rys. 2 strony biorące udział w planowanej realizacji zapytania Q_1 zaznaczono kolorem szarym. Koszt realizacji zapytania z wykorzystaniem indeksu wynosi: $Height + sel(Q_1)LeafCount + sel(Q_1)RowCount$ ¹. Optymalizator rozstrzyga, czy jest on mniejszy od wspomnianego kosztu dla pełnego przeglądu, tzn. *RowPageCount*, a wartość parametru $sel(Q_1)$ ma tu istotne znaczenie.

2. Przyjęte oznaczenia i definicje

Omówienie konstrukcji niektórych histogramów i sposobu oszacowywania selektywności zostanie przedstawione z wykorzystaniem następujących oznaczeniach i założeń (głównie zaczerpniętych z [4]):

- X – atrybut/kolumna relacji/tablicy R ,
- $V = (v_1, \dots, v_N)$ – zbiór wartości atrybutu X , na podstawie których będzie konstruowany histogram; zakłada się, że V jest uporządkowany, tzn. $i < j \Rightarrow v_i < v_j$,
- $f(v)$ – częstość, liczba krotek/wierszy $t \in R$, spełniających warunek $t.X = v$ (termin „częstość” jest kalką językową z j. angielskiego, sugerującą, że mamy do czynienia z wielkością względną; użyty zostanie powszechnie przyjęty termin „częstość”, chociaż lepszym, polskim odpowiednikiem byłoby określenie „liczność”),
- $F = (f_1, \dots, f_N)$ – wektor częstotliwości wartości X , gdzie $f_i = f(v_i)$,
- B – liczba podprzedziałów histogramu (ang. *bucket*); zakłada się, że $B \leq N$, na ogół zachodzi warunek $B \ll N$,
- I_i – podprzedział o zadanym początku i końcu (b_i, e_i); $I_i = \{v_j : b_i \leq v_j \leq e_j\}$, gdzie $\overline{I_i} = \overline{\{v_j : b_i < v_j \leq e_j\}}$ jest liczbą wartości uwzględnionych w danym podprzedziale histogramu,
 $I = (I_1, \dots, I_B)$ – wektor rozłącznych podprzedziałów, rozdzielających wektor V ,

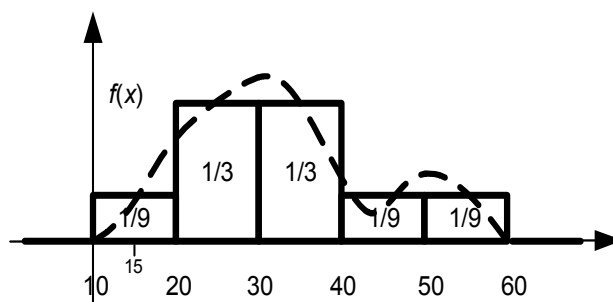
¹ W formule zastosowano uproszczenie, polegające na pesymistycznym założeniu, że podczas przeglądu kolejnych liści i odniesieniu do odpowiedniego wiersza, następuje za każdym razem przejście do innej strony danych, stąd *RowCount* w formule. W przypadku optymistycznym, podczas przeglądu kolejnych liści, potrzebne strony danych są pobierane tylko raz, stąd w formule zamiast *RowCount* wystąpiłoby *RowPageCount*. W przypadku ogólnym w formule wystąpi parametr *ClusterCount* $\in [PageCount, RowPageCount]$, wyrażający stopień integracji indeksu w kontekście rozmieszczenia wierszy tablicy, określony jako liczba pobrań stron danych przy przeglądzie całej listy liści.

- $H = (h_1, \dots, h_B)$ – histogram, wektor aproksymujący F o długości B , gdzie h_i jest aproksymacją częstości w podprzedziale I_i (np. średnia częstość w podprzedziale, tzn. $h_i = \sum_{v_j \in I_i} (f(v_j) / \overline{I_i})$),
- $R_H = (r_1, \dots, r_N)$ – aproksymacyjna rekonstrukcja F na podstawie H i I , R_H – wektor aproksymowanych częstości r_i dla każdej wartości v_i z V , tzn. $r_i = h_j$, gdzie $j: v_i \in I_j$.

Dla sprawdzenia jakości i zapewnienia możliwości porównywania metod histogramowej estymacji f.g.p. wprowadza się metrykę błędu aproksymacji. Na ogół jest to odległość (ang. *distance metric*) pomiędzy rekonstrukcją R_H i F (np. średni błąd kwadratowy $\sum_{i=1}^N (r_i - f_i)^2$).

3. Histogram equi-width

Najprostszym histogramem pod względem sposobu konstrukcji, wykorzystywanym przy wyznaczaniu oszacowania selektywności, jest histogram o stałej szerokości podprzedziałów (ang. *equi-width histogram*). Rysunek 3 przedstawia histogram typu equi-width, o pięciu podprzedziałach, estymujący f.g.p. pokazaną na rys. 1. Posługując się oznaczeniami z podrozdziału 2 podstawowe własności histogramu equi-width (przechowującego wartości bezwzględne, a nie wartości o sumie znormalizowanej do 1 jak na rys. 3) określone są następująco $\forall_{I_i, I_j \in I} e_i - b_i = e_j - b_j = const$, $h_i = \sum_{v_j \in I_i} f(v_j)$.



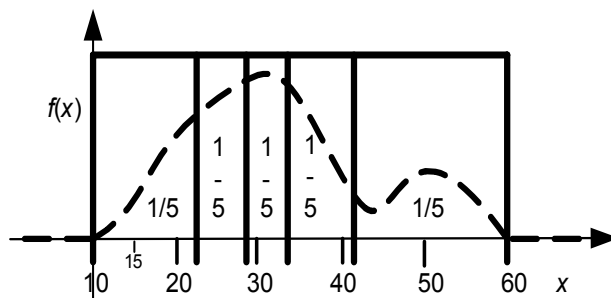
Rys. 3. Histogram equi-width jako estymator funkcji gęstości prawdopodobieństwa
Fig. 3. The equi-width histogram in estimation of a probability density function

Przykładowo, dla zapytania z warunkiem selekcji $20 < X \leq 30$ szacowana wartość selektywności wynosi: $sel(Q(20 < X \leq 30)) = 1/3$. Przy oszacowywaniu selektywności przyjmuje się pewne przybliżenie, tzn. zakłada się, że rozkład wartości zmiennej wewnątrz podprzedziału jest równomierny. Przy takim założeniu możliwe jest oszacowanie selektywności zapytań, dla których krańce zakresów nie pokrywają się z granicami podprzedziałów, np.:

$$\begin{aligned} sel(Q(15 < X \leq 30)) &= (1/2) sel(Q(10 < X \leq 20)) + sel(Q(20 < X \leq 30)) = \\ &= (1/2)(1/9) + 1/3 \approx 0,389. \end{aligned}$$

4. Histogram equi-height

Jeszcze częściej (ze względu na mniejszy średni błąd estymacji f.g.p., przy tej samej liczbie podprzedziałów histogramu) niż histogram equi-width przy oszacowywaniu selektywności w komercyjnych systemach zarządzania b. d. (np. ORACLE, MS SQLServer, Sybase Adaptive Server) wykorzystywany jest histogram o stałej wysokości (ang. *equi-height* lub *equi-depth histogram*) – histogram kwantylowy. Dla histogramów equi-height granice podprzedziałów wyznaczane są w ten sposób, aby prawdopodobieństwa przynależności wartości do danego podprzedziału były równe (lub prawie równe). Rysunek 4 przedstawia histogram typu equi-height, o pięciu podprzedziałach, konstruowany na podstawie przykładowej f.g.p. z rys. 1.



Rys. 4. Zastosowanie histogramu equi-height do estymacji rozkładu prawdopodobieństwa
Fig. 4. The estimation of a probability distribution using the equi-height histogram

Dla histogramu equi-height spełnione jest założenie o równości selektywności dla całych podprzedziałów $sel(Q(b_i < x \leq e_i)) = 1/B$ dla każdego $i = 1, \dots, B$. Wykorzystując oznaczenia z podrozdziału 2, podstawową własność histogramu equi-height (przechowującego wartości bezwzględne, a nie wartości o sumie znormalizowanej do 1 jak na rys. 4) określa wyrażenie $h_i = N/B$. Podobnie jak poprzednio, przy wykorzystaniu tego rodzaju histogramu do wyznaczania selektywności będzie obowiązywać założenie, dotyczące równomiernego rozkładu wartości w podprzedziałach.

Uwzględniając powyższe założenia, oszacowanie selektywności przykładowego zapytania można wyznaczyć przez wyrażenie:

$$sel(Q(15 < X \leq 30)) = sel(Q(15 < X \leq e_1)) + sel(Q(b_2 < X \leq e_2)) + sel(Q(b_3 < X \leq 30)).$$

Przyjmując (zgodnie z rys. 4, na podstawie stosunku długości odpowiednich odcinków)

$$\frac{e_1 - 15}{e_1 - b_1} = \frac{3}{5} \text{ i } \frac{30 - b_3}{e_3 - b_3} = \frac{1}{4}, \text{ oszacowania selektywności składowych wynoszą odpowiednio:}$$

$$sel(Q(15 < X \leq e_1)) = (3/5)(1/5), \quad sel(Q(b_3 < X \leq 30)) = (1/4)(1/5).$$

Stąd ostatecznie szukane oszacowanie selektywności zapytania przykładowego wynosi

$$sel(Q(15 < X \leq 30)) = (3/5)(1/5) + (1/5) + (1/4)(1/5) \approx 0,37.$$

5. Wyznaczenie selektywności dla przykładowego zapytania na podstawie techniki B-tree scan

Selektywność zapytania zakresowego może być oszacowana za pomocą techniki nazywanej płytkim wejściem w drzewo indeksowe (ang. *B-tree scan*). W ramach tej metody zakłada się, że wartości klucza w węźle dzielą dziedzinę atrybutu na podprzedziały o równym prawdopodobieństwie przynależności – podział kwantylowy (podobnie do histogramu equi-height). Założenie to jest tylko w przybliżeniu prawdziwe, bo liczba wyprawdzeń z węzłów może być i na ogół jest różna.

W uproszczeniu algorytm metody szacowania można opisać następująco: Jeżeli kraniec zakresu (prawy albo lewy) z warunku zapytania wystąpi w kluczu węzła, to selektywność wyliczana jest na podstawie liczby kluczy w węźle i algorytm zostaje zakończony. Jeśli nie wystąpi, to selektywność dotychczas wyznaczona jest uszczegóławiana (pomniejszana) w ramach przejścia do niższego poziomu węzłów. Dokładniej, selektywność jest iloczynem dotychczasowej selektywności i selektywności cząstkowej, wyznaczanej z poddrzewa o korzeniu w aktualnym węźle.

Dla indeksu z rys. 2 oszacowanie selektywności na podstawie omówionej metody, dla kilku poniżej przedstawionych przykładowych zapytań, wynosi:

- $sel(Q(X \leq 10)) = 1/3$ – wyznaczone jedynie na podstawie korzenia drzewa indeksowego,
- $sel(Q(X \leq 5)) = sel(Q(X \leq 10)) sel(Q((X \leq 5)|_{x \leq 10})) = (1/3)(1/2)$ – wyznaczone przy przejściu od korzenia do drugiego poziomu węzłów (gdzie $sel(Q((X \leq 5)|_{x \leq 10}))$ oznacza selektywność cząstkową, liczoną w ramach analizy poddrzewa o korzeniu w węźle z kluczem o wartości 5; w tym wypadku ograniczając się tylko do tego korzenia),
- $sel(Q(X < 4)) = (1/3)(1/2)(2/3)$ – wyznaczone przy przejściu do poziomu węzłów-liści,
- $sel(Q(X > 15)) = sel(Q(10 < X \leq 20)) sel(Q((X > 15)|_{10 < X \leq 20})) + sel(Q(X > 20)) = (1/3)(1/2) + 1/3$.

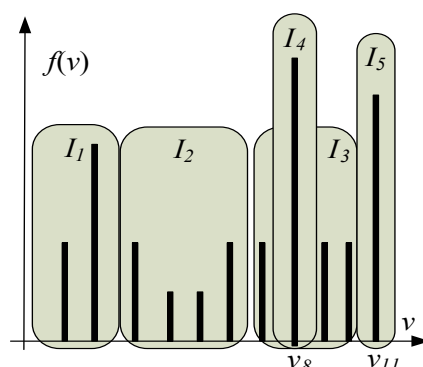
Oszacowana wartość selektywności dla przykładowego zapytania Q_1 z poprzedniego podrozdziału wg omówionej metody wynosi :

$$sel(Q_1(6 \leq X \leq 13)) = (1/3)(1/2)(3/3) + (1/3)(1/2)(2/3).$$

6. Histogram skompresowany

W estymacji rozkładu wartości atrybutu tablicy w komercyjnych zastosowaniach (IBM DB2) wykorzystywany jest histogram typu skompresowanego (ang. *compressed*), w którym specjalnie wyodrębnione są podprzedziały dla kilku największych częstości, tzw. singletony. Pozostałe częstości „składane” są zgodnie z koncepcją tworzenia histogramu equi-height (rzadziej equi-width). Nazwę histogramu można uzasadnić obserwacją, że kompresja (redukcja danych koniecznych do przechowywania) dotyczy par wartość-częstość (v_i, f_i) , niezawierających się w singletonach.

Przykład histogramu typu skompresowanego pokazuje rys. 5 (f występuje tu w znaczeniu częstości, zgodnie z oznaczeniami z podrozdziału 2). W przykładzie przyjęto: $M = 2$ jako liczbę największych częstotliwości do wyodrębnienia, $B = 5$ jako liczbę wszystkich podprzedziałów. Wartości v_8 i v_{11} , o największych wartościach f , przechowywane są w oddzielnych podprzedziałach I_4 i I_5 , pozostałe podprzedziały: I_1, I_2, I_3 wyznaczone są zgodnie z definicją histogramu equi-height, ale z wyłączeniem wartości v_8 i v_{11} .



Rys. 5. Histogram skompresowany użyty do estymacji rozkładu prawdopodobieństwa
Fig. 5. The estimation of a probability distribution using the compressed histogram

Wyznaczenie selektywności z wykorzystaniem omawianego histogramu bazuje na wyznaczeniu aproksymowanych częstości w sposób właściwy dla histogramów equi-height, dodatkowo powiększonych o częstości wynikające z uwzględnienia odpowiednich singletonów.

7. Histogram MaxDiff

W histogramie MaxDiff granice podziału określające podprzedziały są wyznaczane na podstawie wielkości różnicy „sąsiednich częstości”. Koncepcja MaxDiff sprowadza się do rozdzielania do różnych, sąsiednich podprzedziałów, najbardziej różniących się „sąsiednich

częstości”. W konstrukcji histogramu wykorzystuje się wyrażenie $freq_gap(i) = |f_{i+1} - f_i|$ (wielkość „przepaści”), wyznaczające moduł różnicy częstości dla sąsiednich wartości v_i i v_{i+1} . Wartość funkcji $freq_gap(i)$ oblicza się dla $i = 1, \dots, N - 1$, wybiera się $B - 1$ największych wartości $freq_gap(j)$, a uzyskany zbiór indeksów j wykorzystuje się do wyznaczenia granic podziału V na B podprzedziałów.

8. Histogram z optymalną sumaryczną wariancją częstości w podprzedziałach (histogram V-optymalny)

Jednym ze sposobów konstrukcji histogramu estymującego f.g.p. jest podejście polegające na takim wyznaczeniu granic podprzedziałów histogramu, by sumaryczna wariancja częstotliwości w podprzedziałach była najmniejsza ([4]). Im bardziej będą zbliżone do siebie wartości częstości w danym podprzedziale, tym bardziej wariancja częstości będzie mniejsza (bliższa zeru). Stąd też przyjęto następującą miarę jakości aproksymacji w podprzedziale o indeksie początku podprzedziału b i indeksie końca podprzedziału e :

$$SQERR(b, e) = \sum_{i=b}^e (f(i) - AVG(b, e))^2, \quad AVG(b, e) = (\sum_{i=b}^e f(i)) / (e - b + 1),$$

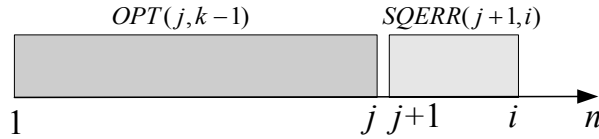
gdzie AVG jest średnią częstością w podprzedziale.

Tworzenie histogramu V-optymalnego opiera się na koncepcji programowania dynamicznego [2]. Koncepcja opiera się na zasadzie optymalności Bellmana, która orzeka, że decyzja podejmowana na pewnym etapie jest wyłącznie funkcją stanu układu (systemu) na początku tego etapu i nie jest zależna od sposobu dojścia do tego stanu. Stąd strategia optymalna nie zależy od historii dojścia do pewnego stanu i może być jednoznacznie określona na podstawie stanu układu w danej chwili.

Na potrzeby algorytmu tworzenia histogramu V-optymalnego zdefiniowany został wskaźnik jakości $OPT(i, k)$, wyrażający optymalną (najmniejszą) wartość sumy $SQERR$ dla histogramu zbudowanego z k podprzedziałów oraz obejmującego początkowy podciąg częstości o długości równej i (czyli ciąg częstości: f_1, \dots, f_i). Zgodnie z zasadą optymalności wartość wskaźnika OPT dla histogramu z k podprzedziałami można wyznaczyć na bazie znajomości wartości OPT wyznaczonych dla histogramu z $k - 1$ podprzedziałami w następujący sposób: $OPT(i, k) = \min_{1 \leq j \leq i-1} (OPT(j, k - 1) + SQERR(j + 1, i))$.

Zagadnienie tworzenia histogramu V-optymalnego o rozmiarze k podprzedziałów może być zredukowane do zagadnienia o rozmiarze $k - 1$, przez analizę wszystkich możliwych (dla $j = 1, \dots, i - 1$) lewych granic ostatniego z prawej, k -tego podprzedziału (rys. 6). Odwołując się do wspomnianej zasady optymalności, można stwierdzić, że przy wyznaczeniu właściwego

(optymalnego) ostatniego podprzedziału (obejmującego indeksy od $j + 1$ do i) nie jest istotny sposób wyznaczenia $OPT(j, k - 1)$, czyli sposób podziału na pozostałe podprzedziały.



Rys. 6. Wybór wartości j wyznaczającej optymalny podział i -elementowej części wektora F dla zadanej liczby podprzedziałów k (krok algorytmu bazującego na koncepcji programowania dynamicznego)

Fig. 6. Choosing value of j to obtain optimal buckets boundaries for the subvector of F consisting of i elements and for k – given number of buckets (a step of the algorithm based on the dynamic programming idea)

Znalezienie histogramu V -optymalnego sprowadza się do wyznaczenia wyrażenia $OPT(N, B)$, czyli wielokrotnych obliczeń $OPT(i, k)$ dla kolejno zwiększającego się $k = 1.. B$, tzn. rozmiaru histogramu oraz zwiększającego się $i = 2.. N$, tzn. podciągu obejmującego coraz więcej elementów wektora częstości F .

Na przykład, dla pewnego ciągu częstości podział na podprzedziały histogramu V -optymalnego jest następujący: $I = (\{ 1, 2, 3, 1 \} \{ 6, 6 \} \{ 1, 2 \})$, gdzie znaki ‘{’ i ‘}’ rozdzielają ciąg częstości, wyznaczając granice podprzedziałów, a optymalna wartość wskaźnika jakości wynosi $OPT(8, 3) = 3,25$.

W omówionym sposobie optymalizacji wskaźnika OPT wykorzystano formułę błędu średniokwadratowego $SQERR$ jako miary jakości aproksymacji w danym podprzedziale. Jednak prezentowany sposób znajdowania minimum OPT (bazujący na zasadzie Bellmana) może być również wykorzystywany, gdy użyta zostanie inna metryka błędu aproksymacji w podprzedziale, np. oparta na pojęciu entropii [5].

Entropia informacyjna $Ent(p_1, \dots, p_n) = -\sum_{j=1}^n p_j \log_2(p_j)$ rozkładu zmiennej dyskretnej V

osiąga wartość największą dla rozkładu równomiernego, tzn. $Ent(p_1, \dots, p_n) \leq Ent(\frac{1}{n}, \dots, \frac{1}{n}) = \log_2(n)$, gdzie p_j oznacza prawdopodobieństwo $P(V = v_j)$, a wyrażenie $\log_2(n)$ to wartość entropii dyskretnego rozkładu równomiernego o n wartościach.

Dla wyznaczenia zaproponowanej metryki jakości bazującej na pojęciu entropii wprowadza się w [5] wektor częstości względnej w i -tym podprzedziale

$$RF(I_i) = \{(v_j, p_j) : p_j = f(v_j) / \bar{I}_i \wedge v_j \in I_i\}.$$

Entropia rozkładu częstości względnych w podprzedziale I_i wynosi

$$Ent(RF(I_i)) = -\sum_{j: v_j \in I_i} p_j \log_2(p_j).$$

Jako miarę podobieństwa rozkładu $RF(I_i)$ do rozkładu równomiernego w tym podprzedziale można przyjąć wielkość różnicy entropii rozkładu $RF(I_i)$ i entropii rozkładu równomiernego, tzn. $\log_2(\bar{I}_i) - Ent(RF(I_i))$. Uwzględniając różną licznosc częstości w podprzedziałach, wskaźnik jakości wykorzystujący entropię informacyjną, określony jako ważona suma wartości odchyłek entropii wewnątrzprzedziałowej od entropii rozkładu równomiernego, można wyrazić następująco:

$$\sum_{j=1}^B ENTERR(j) = \sum_{j=1}^B (\bar{I}_j [\log_2(I_j) - Ent(RF(I_j))]).$$

Stąd, w algorytmie optymalizacyjnym wyznaczającym histogram V-optymalny, w wyrażeniu na wartość wskaźnika OPT można zastąpić SQERR przez ENTERR, uzyskując $OPT(i, k) = \min_{1 \leq j \leq i-1} (OPT(j, k-1) + ENTERR(k))$, gdzie ENTERR(k) oznacza odchyłkę (zmodyfikowaną o wagę podprzedziału) od entropii rozkładu równomiernego dla k -tego podprzedziału, zbudowanego na częstościach z F o indeksach od $j+1$ do i .

9. Podsumowanie

W niniejszym, przeglądowym artykule pominięto opis metod bazujących na: estymacji jądrowej f.g.p. (ang. *kernel estimation*) [7, 9, 10], własnościach dyskretnej transformaty kosinusowej (ang. DCT – *discrete cosine transform*) [6] oraz własnościach dyskretnej transformaty falkowej (ang. DWT – *discrete wavelet transform*) [8]. Metody stworzone dla estymacji rozkładów wielowymiarowych, chociaż opracowane teoretycznie i dokładniejsze dla niektórych klas warunków zapytań, na ogół nie są zaimplementowane w komercyjnych serwerach b.d. (choć np. nowsze wersje MS SQLServer obsługują mechanizm multicolumn statistics). Niektóre z tych metod (np. jądrowa, DCT, DWT) mogłyby być skutecznie wykorzystywane również przy estymacji rozkładów jednowymiarowych, których omówieniem zajmuje się niniejszy artykuł.

Artykuł prezentuje podstawowe techniki bazujące głównie na histogramowych metodach estymacji rozkładu. Omawiane techniki, w związku z ich realnym lub potencjalnym zastosowaniem, rozważa się i porównuje w różnych kategoriach, np. średniego błędu estymacji, kosztu (czasowego) tworzenia histogramu, kosztu utrzymania histogramu (koszt aktualizacji po zmianie zawartości b. d.), rozmiaru pamięci wymaganej do przechowywania histogramu w b.d., kosztu (czasowego) użycia histogramu (złożoność wyliczania selektywności). Kategorie te są analizowane w kontekście: zadanej liczby podprzedziałów, rozmiaru danych F , cech statystycznych estymowanego rozkładu (np. wielomodalność f.g.p., skośność rozkładu itp.). Częściowo o tej tematyce traktują wspomniane prace [4, 5, 6, 7, 8]. W badaniach ilości-

wych, opisanych w tych pracach, uwzględniane zostały zbiory danych syntetycznych (tworzonych z użyciem generatorów pseudolosowych dla zadanych rozkładów prawdopodobieństwa), jak i zbiory danych „rzeczywistych” (zbiory referencyjne, dostępne w Internecie). W przybliżeniu, dla analizowanych zbiorów danych, metody histogramowe można uszeregować pod względem rosnącego błędu estymacji (błędu średniokwadratowego liczonego wg założeń z podrozdziału 2) następująco: V-optimal, MaxDiff, compressed, equi-height, equi-width. Kolejność uszeregowania będzie na ogół odwrotna przy uwzględnieniu kryterium rosnącego kosztu (złożoności) konstrukcji histogramu. Najbardziej kosztowna obliczeniowo jest konstrukcja histogramu V- optymalnego, nawet po uwzględnieniu dwóch wariantów algorytmu wprowadzonych w pracy [4], tzn.: efektywniejszego algorytmu optymalnego (bazującego na własności $SQERR(i, j) \geq SQERR(i, k) + SQERR(k + 1, j)$) oraz algorytmu w przybliżeniu optymalnego.

Chociaż prezentowany artykuł ma charakter przeglądowy, to analiza istniejących rozwiązań nasuwa pewien wniosek natury jakościowej. W zasadzie bardziej zaawansowane techniki histogramowe (oprócz equi-height i equi-width) nie znalazły jeszcze szerokiego zastosowania komercyjnego, tzn. nie są implementowane w serwerach b.d. W serwerach b.d. istnieją odpowiednie mechanizmy tworzące i aktualizujące odpowiednie struktury histogramowe, ale są to mechanizmy ukierunkowane na użycie jednego, konkretnego typu histogramu. Można byłoby sobie wyobrazić, że w serwerze b.d. zaimplementowano obsługę kilku typów histogramów, również i tych kosztownych pod względem tworzenia i aktualizacji, ale dokładniejszych. Te ostatnie mogą być użyte np. dla statystycznego opisu danych podlegających bardzo rzadkim zmianom (dane słownikowe).

Można byłoby sobie również wyobrazić, że aktualizacja histogramu sprowadzałaby się nie tylko do uwzględnienia kolejnego przyrostu, tzn. nowego zbioru danych, ale prowadziłaby do jakościowej zmiany sposobu estymacji rozkładu. Przykładowo, pod wpływem analizy cech statystycznych danych, w ramach procesu aktualizacji, mógłby nastąpić automatyczny dobór innych wartości parametrów estymacji (np.: liczba podprzedziałów histogramu, wartość parametru wygładzania dla estymatora jądrowego) albo nawet zmianę sposobu estymacji, np. typu histogramu.

Doceniając wagę problemu obliczania selektywności, niektórzy producenci serwerów baz danych przewidują możliwość rozszerzania funkcjonalności systemów zarządzania bazą danych przez umożliwienie implementacji własnych mechanizmów definiowania selektywności przez zaawansowanych użytkowników – tzw. user-defined selectivity mechanism (np. w serwerze b.d. ORACLE 10g, w ramach pakietu ODCIStat – Oracle Data Cartridge Interface Statistics).

LITERATURA

1. Possala V., Ioannidis Y.E.: Selectivity Estimation without the Attribute Value Independence Assumption. Proc. of the 23rd Int. Conf. on Very Large Databases, The VLDB Journal, Athens 1997.
2. Feldbaum A. A.: Podstawy teorii optymalnych układów sterowania automatycznego. PWN, Warszawa 1967.
3. Boring J., Gerber D.: SQLBase Advanced Topic Guide. SQL Design, Denmark, Viby 1995.
4. Jagadish H. V., Koudas N., Muthukrishnan S., Possala V., Sevcik K., Suel T.: Optimal Histogram with Quality Guarantees. Proc. of the 24rd Int. Conf. on Very Large Databases, The VLBD Journal, New York 1998.
5. Giannella C., Sayrafi B.: An Information Theoretic Histogram for Single Dimensional Selectivity Estimation. Proc. of the 2005 ACM Symposium on Applied Computing. ACM, New York 2005.
6. Lee. J., Deok-Hwan K., Chin-Wan Ch.: Multi-dimensional Selectivity Estimation Using Compressed Histogram Estimation Information. Proc. of ACM SIGMOD Int. Conf. on Management of Data. ACM, Philadelphia 1999.
7. Blohsfeld B., Korus D., Seeger B.: A Comparison of Selectivity Estimators for Range Queries on Metric Attributes. Proc. of ACM SIGMOD Int. Conf. on Management of Data. ACM, Philadelphia 1999.
8. Matias Y., Vitter J. S., Wang M.: Wavelet-Based Histograms for Selectivity Estimation. Proc. of ACM SIGMOD Int. Conf. on Management of Data. ACM, Washington 1998.
9. Gunopulos D., Kollios G., Tsotras V.J.: Approximating Multi-Dimensional Aggregate Range Queries Over Real Attributes. ACM SIGMOD 2000, Dallas 2000.
10. Augustyn D. R.: Metoda estymacji jądrowej w szacowaniu selektywności zapytań. Studia Informatica. Vol 23, no. 4, Gliwice 2002.

Recenzent: Dr inż. Maciej J. Bargielski

Wpłynęło do Redakcji 6 grudnia 2007 r.

Abstract

The paper contains a survey of database query selectivity methods. The selectivity is a parameter required for the estimation of a query result set size. The selectivity for a simple

single-table query is a fraction, where the numerator is the number of rows satisfying the given query criteria and the denominator is the total number of table rows. The selectivity for a single-table query can be considered as a probability of a random selection of a single row satisfying the query criteria from the whole set of table rows.

The calculated selectivity value is used by the query optimizer module to obtain the most effective query execution plan. A query execution according to the optimal plan should cause performing the minimal number of database read operations.

Selectivity calculations are based on an estimation of the probability density function (PDF). Histograms as non-parametric estimators PDF are considered in this paper. This article concentrates on single-dimensional estimators. The paper describes several types of histograms: the equi-width, equi-height, compressed one, which is used by commercial DBMS and also less known V-optimal histogram and the entropy-optimal one based on the dynamic programming idea and information theorems.

Adres

Dariusz Rafał AUGUSTYN: Politechnika Śląska, Instytut Informatyki, ul. Akademicka 16, 44-100 Gliwice, Polska, draugustyn@polsl.pl.