5D-4

標準 SQL 基本関数を用いたデータ要約による近似集計手法

藤野 友也　　平井 規郎　　東　辰輔

三菱電機株式会社 情報技術総合研究所

1　はじめに

データベースサーバ上の膨大なデータに対して、クライアント側でデータベース単体ではできない高度な非定型集計を行う場合、直接アプリケーションが生データを取得して処理を行うと転送量が膨大となり、処理時間が増大する。

本論文では、データの特性把握など、精度が重視されない用途に向けて、標準的な DBMS が共通して提供する基本的な SQL 機能を用いてサーバ上で集計した要約データをクライアントに送信、クライアント側で元データを推定し集計処理を行う方法を検討し、データ転送量、処理性能および分析可能データ量に着目して改善効果を評価する。

以後、標準 SQL の基本的機能として、SQL-92 にて定義された機能を利用する。

2　近似集計手法

集計対象項目のとりうる値を、いくつかの範囲に分割し、各範囲に属する値の最大値･最小値･平均値･個数を一次集計値として利用し、近似集計を行う手法について述べる。

SQL クエリによる値域の等分割方法の一例として、副問合せとキャスト(共に SQL-92 に準拠)を利用する方法がある。例えば、スキーマ名 SCHEMA、テーブル名 TABLE の項目 ITEM について、値が 10.0 ごとの範囲で個数を求める集計を行う場合は、次のような SQL 文を使用する。

```
SELECT COUNT(ITEM) FROM (SELECT ITEM, CAST((ITEM/10.0) AS INT) AS ITEM_INDEX FROM SCHEMA.TABLE) AS T GROUP BY ITEM_INDEX
```

上記の SQL 文は副問合せがあるため効率はやや悪いが、対象テーブルを変更することなく動的に範囲の幅を変更可能である。使用する DBMS に効率的に代替可能な独自機能がある場合は、そちらを利用すべきであるが、本稿では上記の方法を用いる。

An approximate aggregation by summary summarized with standard SQL primitive functions.
Tomoya FUJINO, Norio HIRAI, Shinsuke AZUMA,
Mitsubishi Electric Corporation,
Information Technology R&D Center

ここで、i 番目の分割範囲を$[d_i\ u_i]$とし、その範囲内の値を一次集計した際の個数を c_i, 平均値を m_i, 最小値を p_i, 最大値を q_i とする。

最終的な集計方法 (元データ推定方法) として、本稿では以下の 4 つを扱う。(図 1参照)

(1) 個数 c_i のみを使用
　各 i に対し、範囲の中央値 $(u_i+d_i)/2$ が c_i 個存在するとして最終集計を行う。

(2) 個数 c_i, 平均値 m_i を使用
　各 i に対し、値 m_i が c_i 個存在するとして最終集計を行う。

(3) 個数 c_i, 最小値 p_i, 最大値 q_i を使用
　各 i に対し、値が c_i 個、 範囲 $[p_i\ q_i]$ に等間隔で存在するとして最終集計を行う。

(4) 個数 c_i, 最小値 p_i, 最大値 q_i,平均値 m_i を使用
　各 i に対し、c_i 個の値が、最小値 p_i、最大値 q_i、平均値 m_i の三角分布に従い分布していると仮定して最終集計を行う。

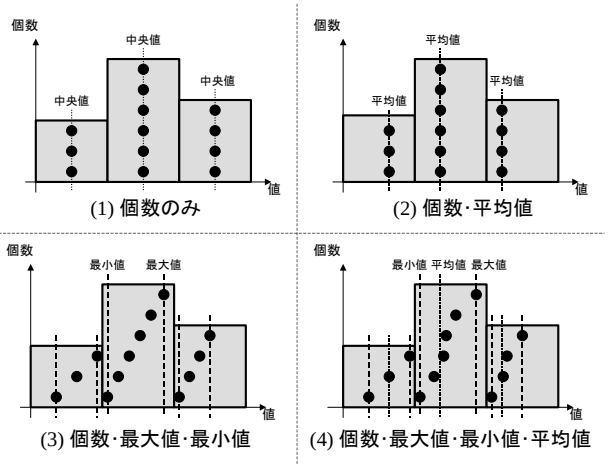

図 1 仮定している値の分布例

上記の (4) では、三角分布に限らず正規分布やベータ分布などを仮定することも可能である。

確率密度分布がよく知られたデータ分布を仮定すると、最終集計に必要な中間値の期待値を解析的に導出できるため、クライアント側の最終集計演算を効率的に行うことができる。

3　性能評価

性能評価は、サーバ PC とクライアント PC (共に CPU は Intel^{R} Xeon^{TM} 3.06GHz) を用意し、本稿で使用する機能を備える市販 DBMS を用いて、サーバ･クライアント間の通信速度を 100Mbps, 10Mbps, 1Mbps のそれぞれの場合について測定した。集計対象データとしては、ある設備の消費電力量推移の実データ (184,863,060 レコード) を使用し、最終集計値として分散を導出する場合の近似精度および処理時間を評価する。

3.1　近似精度

各値域分割数(10～20,000)に対する近似率推移を図 2 に、誤差率が 0.05% 以内となる値域分割数を表 1 に示す。

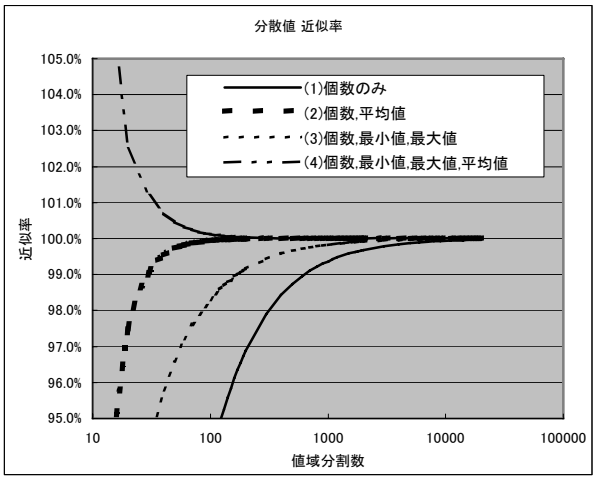

図 2 値域分割数に対する近似率推移

表 1 近似誤差が 0.05%以内となる値域分割数

近似集計手法	(1)	(2)	(3)	(4)
値域分割数	13000	120	4000	160

図 2 および表 1 から、第2章で述べた各手法のうち、(2)と(4)の収束が早いことがわかる。また、データ分布を仮定して高度な推定を行う(4)と比べても、平均値のみを利用した(2)の収束が早いという結果が得られた。

上記では、適切な集計手法 (2) を用いることで、120 の分割範囲それぞれの個数および平均値から、1 億 8 千万個の値の分散を十分な精度で近似算出できることを示した。

3.2　処理速度

通信速度 100Mbps の際の生データの集計処理時間を 1 とした時の、各近似集計手法 (値域分割数 20,000)、各通信速度に対する集計処理時間の比率を図 3 に示す。

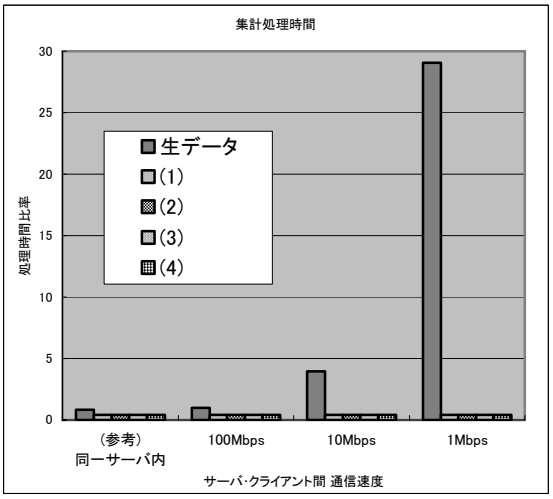

図 3 各通信速度に対する処理時間

生データを集計する場合、サーバがクライアントへ送信するデータ量が膨大なため通信速度の影響を受け、通信速度 100Mbps の場合の処理時間に比べ、通信速度 1Mbps の場合の処理時間は 29.1 倍である。

一方、近似集計処理の場合、処理時間は通信速度によらず、生データ集計 (通信速度 100Mbps) の処理時間の 0.41～0.43 倍である。これは、通信以外の、特にサーバ側の SQL 集計にかかる処理時間が支配的であるためである。

よって本手法により、通信環境に依存しない処理時間での近似集計が可能になる。

4　結論と今後の課題

本稿では、大半の DBMS が対応する SQL-92 のみに準拠した方法で、効率的に精度の高い近似集計値を導出可能であることを示した。これにより、SQL-99 にて定義されたユーザ定義関数機能 (ユーザ定義関数による集計値を HAVING 句や副問合せなど内部利用する集計を除く)を、SQL-92 のみ対応の DBMS にて近似的に実現することが可能となる。また、特に狭帯域通信環境下で大幅な効率化を達成可能である。

さらに、元データ量によらずクライアントで取り扱うデータ量を制御することが可能であるため、マシン性能 (演算速度･メモリ量) に応じた集計処理を行うことができる。

今後の課題としては、本手法の範囲分割数と精度の関係の解析的評価と、値域分割処理の SQL-92 の範囲内での効率化の検討が挙げられる。