

CTCBMQ: A Novel Corner Transformation-Based Algorithm for Continuous Border Monitoring Query Processing In Data Streams

Chongzheng Huang^{1,2}, Hong Chen^{1,2}

¹School of Information, Renmin University of China, Beijing, China

²Key Laboratory of Data Engineering and Knowledge Engineering, MOE, Beijing, China
{huangchongzhengruc, chong}@163.com

ABSTRACT

This paper describes my Ph.D. work on data stream research. In this paper we propose a corner transformation-based algorithm CTCBMQ (Corner Transformation-Based Algorithm for Continuous Border Monitoring Query) for Continuous Border Monitoring Query processing (CBMQ). CTCBMQ transforms the CBMQ processing problems into the spatial join processing problems and retrieves query results using corner transformation. In this paper, I also give the directions for our further work.

Keywords

Continuous border monitoring, data stream.

1. INTRODUCTION

In order to monitor data streams, a large number of range queries may be created and evaluated continually against each data item. In many cases, users are interested in knowing which ranges a stream just jumped into or out of, rather than those ranges which the stream is lying in. It is useful enough to report to users the beginning and end of satisfying range conditions. We characterize such type queries as *Continuous Border Monitoring Continuous Queries (CBMQs)*. CBMQs are quite different from traditional continuous range monitoring queries (CRMQs). CRMQs report the streams within the configured ranges, on the contrary, CBMQs focus on the streams which just jumped into or out of the ranges.

Many efficient approaches have been proposed for CRMQ [1][4][5] etc. However, in spite of the usefulness of CBMQs, few previous works have addressed the problem and developed a special mechanism for CBMQ. Usually, people evaluate CBMQs in a mechanism based on CRMQ, namely DiffRMQ[2]. DiffRMQ derives CBMQ results via two steps. Firstly, it retrieves two consecutive query range sets at previous and current moments through CRMQ index. Then, it performs a difference operation on

them. We can easily see that DiffRMQ is not suitable for CBMQs. In [2], a index structure named RS-list is proposed to CBMQ processing. However, RS-list suffers from dramatic performance degradation as the number of queries increases.

In this paper, we propose a novel corner transformation-based algorithm for continuous border monitoring query processing, namely *CTCBMQ*. CTCBMQ transforms the CBMQ processing problems into the spatial join processing problems and retrieves query results using corner transformation.

2. CBMQ SPATIAL MODEL

Suppose that the data stream S consists of attributes ($attr_1, attr_2, \dots, attr_i$) and the domain of $attr_i$ is Dom_i ($0 < i < n$). Then, the data stream S corresponds to a n -dimensional domain space $Dom_1 \times Dom_2 \times \dots \times Dom_i$ and a data item $d(v_1, v_2, \dots, v_n)$ is a point $p(v_1, v_2, \dots, v_n)$ in the domain space. Suppose d_{i-1} and d_i are two consecutive data items of S , and their corresponding points in domain space are p_{i-1} and p_i respectively. We define the line connecting p_{i-1} to p_i as the *stream value-changing line*, denoted as $svcl(i)$. If d_{i-1} and d_i are the last two data items, we can say that the $svcl(i)$ is the *current svcl*, denoted as $csvcl$. Suppose that a CBMQ with a monitoring range is $\delta(attr_1, attr_2, \dots, attr_i)$. In the domain spatial space, the CBMQ can be represented as a region $R \{(v_1, v_2, \dots, v_n) \mid \delta(v_1, v_2, \dots, v_n), v_i \in Dom_i\}$. The stream whose $csvcl$ intersects with the region R is crossing the border of the CBMQ monitoring range.

3. CTCBMQ ALGORITHM

3.1 Basic CTCBMQ

CTCBMQ transforms stream value-changing lines and monitoring regions in original space (o-space) into points in transformed space (t-space). The left end of object in o-space is mapped to the horizontal (lx) axis and the right end to the vertical (rx) in the t-space.

Fig.1 gives a stream value-changing line L in o-space and its corresponding point p in t-space. The upper triangle, which includes all the transformed objects, is partitioned into six areas labeled A to F according to the spatial relationships with the line L . These areas have the following characteristics: (1) All monitoring range enclosing L in the o-space are mapped into area A since their lx-values are less than that of lx and the rx-values greater than that of rx. (2) All monitoring ranges enclosed by L are mapped into area D since their lx-values are greater than lx and rx-value less

* Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Proceedings of SIGMOD2007 Ph.D. Workshop on Innovative Database Research 2007(IDAR2007), June 10, 2007, Beijing, China.

* This research is supported by the National Science Foundation under grant number 60673138 and by Program for New Century Excellent Talents in University.

than rx . (3) All monitoring ranges only intersecting with the left end of L are mapped into area B since their lx -values are less than lx and the rx -value between lx and rx . (4) All monitoring ranges only intersecting with the right end of L are mapped into area E since their rx -values are between lx and rx and the rx -value greater than rx . (5) All monitoring ranges residing within the left of L are mapped into area C since their rx -values are less than lx . (6) All monitoring ranges residing within the right of L are mapped into area F since their lx -values are greater than rx .

Definition 1. Spatial Join Window for Point (SJWP) Let $TS(R)$ and $TS(S)$ be the transformed spaces of stream value-changing lines and the monitoring regions of CBMQs respectively. Point p is a point in $TS(R)$, corresponding to a stream value-changing line L in o -space. The $SJWP(p)$ is defined as the minimal region in $TS(S)$ where all the monitoring ranges intersecting with L are mapped into.

Using this area partition, we know that $SJWP(p)$ is the union of area B and E. That is, $SJWP(p)$ is the shaded region in Fig. 1(b).

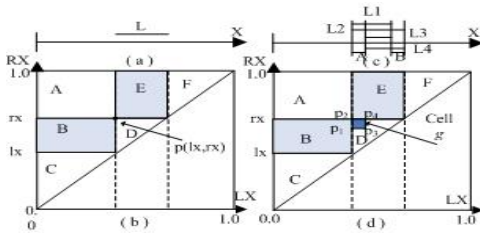


Fig 1. Corner transformation

3.2 Advanced CTCBMQ

In many practical situations, data streams exhibit locality. The values of data streams often change slowly, rather than change abruptly. To exploit locality of data streams, we partition t -space into many grid-like cells of the same size. Let us assume that L_{i-1} and L_i is two *svcls* and their transformed points in t -space are p_{i-1}, p_i respectively. If the cell which p_i lies in is the same one where p_{i-1} stay, we call this a *hit*, else a *miss*. When a hit happens, we can reuse the search result of p_{i-1} , otherwise, we will reevaluate the queries for p_i .

Definition 3: Spatial Join Window for Cell (SJWC) Let $TS(R)$ and $TS(S)$ are the transformed spaces of *svcls* and the monitoring regions of CBMQs respectively. The spatial join window $SJWC(g)$ for a cell g in $TS(R)$ is defined as the minimal region in $TS(S)$ where all the monitoring ranges intersecting with the *svcls* in g can reside.

In Fig. 1(d), cell g is the dark shaded region above the diagonal. Four corner points of g can be regarded as the transformed points of four stream value-changing lines $L_1, L_2, L_3,$ and L_4 in Fig.1(c). we can see that L_2 , representing the upper left point p_1 , is the largest object in cell g . If the lx -value of a stream value-changing line is in A and the rx -value in B, its transformed point is located in cell g . In Fig. 1(d), $SJWC(g)$ can be obtained as follows. To intersect with a *svcl* in cell g , the monitor range of the CBMQ must intersects with the L_2 since it is the largest object enclosing all the other objects in cell g . Thus, $SJWC(g)$ is $([0, lx] * [lx, rx]) \cup ([lx, rx] * [rx, 1])$. That is, $SJWC(g)$ is the light shaded region in Fig. 1(d).

3.3 Query Ranges Indexing

The query ranges tend to overlap with one another, especially when a large number of CBMQs are registered into system. We avoid this problem by transforming ranges with extent to points without extent. Hence, when a query is registered, its range is mapped into a point in t -space using corner transformation, before it is inserted into the multi-dimensional index. To handle the skewed distribution efficiently, we choose MLGF [3] to index monitoring ranges in t -space.

4. PERFORMANCE EVALUATION

In our experiments, we use the real streams obtained from Shanghai Stock Market as input data and synthetically generate CBMQs with different monitoring ranges. We compare the search performance and storage cost of advanced CTCBMQ with the existing algorithms: DiffRMQ based on CEI [1] and RS-list [2].

Search performance comparison: The search time of DiffRMQ is significantly higher than RS-list and CTCBMQ. The monitoring ranges tend to overlap each other, making the AET (average elapsed time of data items) of RS-list soars as the number of queries increases. The slight increase in the AET of advanced CTCBMQ comes from the increase in the final result size as the number of registered queries increases.

Storage Cost: As the number of queries increases, RS-list uses much more storage than DiffRMQ and CTCBMQ. To DiffRMQ and CTCBMQ, there exists linear relationship between the query number and the storage cost as both of them store a query only once.

5. CONCLUSION AND FUTURE WORK

This paper describes my Ph.D. work on data stream research. In this paper, we propose CTCBMQ, which evaluates a large number of CBMQs efficiently based on corner transformation spatial join. CTCBMQ can also utilize the locality of data stream. In future study, we plan to implement a prototype system special to CBMQs and extent corner transformation to support join operator in data streams.

6. REFERENCES

- [1] K.L.Wu and P.S.Yu.: Interval Query Indexing for Efficient Stream Processing. CIKM 2005.
- [2] Jinwon Lee, Youngke Lee.,etc.; BMQ-Index: Shared and Incremental Processing of Border Monitoring Queries over Data Streams. MDM 2006.
- [3] Song, J.-M., Whang, K.-Y.: Spatial Join Processing Using Corner Transformation. IEEE Trans. On Knowledge and Data Engineering, Vol.11, No.4, July 1999.
- [4] Jinwon Lee, Youngke Lee.,etc.; LARI: Locality-Aware Range Query for High Performance Data Stream Processing. Technical Report August 2004.
- [5] E. Hanson and T. Johnson.: Selection Predicate Indexing for Active Databases using Interval Skip Lists. Information Systems, 21(3):269-298, 1996.