INTERNATIONAL JOURNAL OF PURE AND APPLIED RESEARCH IN ENGINEERING AND TECHNOLOGY

A PATH FOR HORIZING YOUR INNOVATIVE WORK

SPECIAL ISSUE FOR INTERNATIONAL LEVEL CONFERENCE "ADVANCES IN SCIENCE, TECHNOLOGY & MANAGEMENT" (IC-ASTM)

PERFORMANCE ANALYSIS OF CACHE INVALIDATION ALGORITHMS IN MOBILE ENVIRONMENT

MR. AJAY P. CHENDKE[1], DR. S. S. SHEREKAR[2]

1. Department of Computer Science, D.C.P.E., H.V.P.M., Amravati, India.
2. Department of Computer Science, Amravati University, Amravati, India

Accepted Date: 05/09/2017; Published Date: 10/10/2017

Abstract: Wireless computing becomes most popular with the development of mobile computers. Cache technique in wireless computing is crucial because it facilitates the data access at clients for reducing servers' loading, hence improve the performance. However, conventional cache technique requires coherence between servers and clients because of frequent disconnection. Caching of frequently accessed data items will be an important technique that will reduce contention on the narrow bandwidth, wireless channel. In this paper, we address the problem of cache invalidation in mobile environments. Bit-Sequence, Multidimensional BS, Multilevel BS a new cache invalidation algorithm and old cache invalidation algorithm like TS and SIG in which a periodically-broadcast invalidation report is organized as a set of binary bit sequences with an associated set of timestamps. These algorithms performs consistently well under conditions of variable update rates/patterns and client disconnection times. Furthermore, the size of the invalidation report in this algorithm is relatively small and is independent of the number of data items to be invalidated. The purpose of this paper is to understand how these algorithms behave to performance of the queries according to parameters with the request of the users. This paper provides many unique characteristic, challenges and performance and behavior of query processing algorithms in mobile environment.

Keywords: Query processing, mobile computing, query optimization, algorithm, mobile environment.

Corresponding Author: MR. AJAY P. CHENDKE

Co Author: - DR. S. S. SHEREKAR

Access Online On:

www.ijpret.com

How to Cite This Article:

Ajay P. Chendke, IJPRET, 2017; Volume 6 (2): 265-274

PAPER-QR CODE

INTRODUCTION

The main objective is to investigate the performance improvement of mobile query processing, focusing on the server and client sides. In server side query processing, a single-cell and multi-cell queries, whereby a cell is a service area for a single stationary host to communicate with a static network. A quick response in answer to a mobile query is important, because mobile users invariably move to another location while awaiting the query result. To handle such a dynamic situation and give better solutions single-cell and multi-cell queries. The solutions for processing single-cell queries are divided into static and dynamic query scopes, and angle of movement. The static and dynamic query scopes are extended to process multi-cell queries [1]. Furthermore, another solution is in order to deal with a situation where the areas of several base stations are either disjoint or overlapping. Finally, to handle disconnections which occur during query result transmission from a base station to the mobile client. Indexing mechanisms are important to speed up query processing, especially for handling multi-cell queries. There aretwo indexing mechanisms called Local Index and Global Index mechanisms. The local index stores indexes of any requested objects with limited slots, whereas the global index builds the Index while a base station is starting up. For both mechanisms, to deal with the existence and non-existence of replicated objects at the requested cell. A frequent disconnection is a common problem occurring in a mobile environment. Providing a cache in a mobile device is an important consideration [2].

A cache is useful if the repeat of many queries can be retrieved from the cache. Due to the limitation of storage space in the mobile device, three cache replacement policies, called: Path-based, Density-based and Probability Density Area Inverse Distance (PDAID) mechanisms, which are based on distance, weight and cost factors for each method, respectively. The server side query processing show an improvement of the total retrieved objects while the query processing time and the amount of data transfer are reduced [3].

Furthermore, the server is able to decide whether the next query result needs to be produced when the mobile users missed the current query result. The indexing mechanism has reduced the execution time compared with the conventional approach in processing multi-cell queries. The approaches for the client side have also improved the cache-hit rate while reducingthe amount of data transfer [4].

Nowadays more people have started using mobile phone to access information they need from anywhere at any time. In advanced mobile technology, Location Service lets users quickly pinpoint their location. However, it has other factors to achieve more efficient service. There are three prominent factors which are time, place and purpose and between theirrelationship. The basic rules are to analyze essential data and algorithms to query processing. Query processing can be recommended to push the appropriate message to users. Advanced query processing for mobile environment and its optimization is an area which comes under the field of mobile computing [5]. The fast development of wireless communication technology and battery powered portable devices had led to emergence of mobile computing as new tool data communication and processing. The mobile environments are composed of wireless technologies in which user asked for query to be processed on query processing.The mobile environment is collection of mobile heterogeneous hosts, which are enabled to communicate using wireless links. These wireless links may change according to the natures of mobile networks, moreover, nodes in the ad-hoc network have to communicate without any centralized or help [6]. The usability for the user will also change using query processing mechanism Thus, this mechanisms that allows the sharing of functionality among differentdevices in same environment will change the way of user interaction for searching fast query time in mobile computing environment. The increasing success of relational database technology in data processing is suitable, in part, to the availability of nonprocedural languages, which can significantly improve application development and end-user productivity. By hiding the low-level details about the physical organization of the data, relational database languages allow the expression of complex queries in a concise and simple fashion [7].

The algorithms are effective for a large number of cached data items with low update rates. It also illustrates that the algorithms can be used with other complementary techniques to address the problem of cache invalidation for data items with varied update and access rates. The effectiveness of the algorithms can, however, is improved by applying complementary invalidation techniques to deal with dynamically changed update rates and patterns.

Advances in wireless and mobile computing environments allow a mobile user to access a wide range of applications[8].

However, sensor nodes in sensor networks are considerably constrained because with their energy and memory resources they have a very limited ability to process any information compared to conventional computer systems. Thus query processing over the nodes should be constrained because of their limitations. Due to the problems, the join operations in sensor networks are typically processed in a distributed manner[9].

- **background of cache invalidation algorithms:**

In the Bit-Sequences (BS) algorithm, the server broadcasts a set of bit sequences. Each sequence consists of a series of binary bits and is associated with a timestamp. Each bit represents a data item in a database [10]. A bit "1" is in a sequence means that the item represented by the bit has been updated since the time specified by the timestamp of the sequence. A bit "0" means that that item has not been updated since that time. The basic BS algorithm can be applied in the applications where both servers and clients do not need to frequently synchronize the mapping of bits in the sequence (or vector) to the names of data items in the database. In the bit-sequence naming, each bit in a bit-sequence (or bit- vector) represents one data item in the database. Three techniques were applied in the BS method. First, where there are mostly cached and referenced data items are predictable. Second, instead of including one update timestamp for each data item, an update aggregation technique to group a set of data items and to associate the set with only one timestamp in the report [11].

The client that disconnects after the timestamp can use the bit-sequence to identify the updated items. Third, the hierarchical structure of bit-sequences technique to link a set of bit sequences such that the structure can be used by clients with different disconnection times. In this case, the bit mapping information does not have to be explicitly included. This mapping scheme is called a static bit mapping scheme.The optimization is achieved at the cost of about two binary bits per each item. The basic BS algorithm can also be used in applications where clients and servers need to constantly synchronize the mapping of bits. For example, data items represented by the bits can be those that are updated recently (e.g., during the last minute) and can be different from one report to another. In this case, the bit mapping information has to be included in each report so thatthe clients know which data items are indicated by the bit sequences. This bit mapping scheme is called a dynamic bit mapping scheme the basic BS algorithm can be applied to optimize broadcast-based cache invalidation algorithms in which the dynamic bit mapping is required [12].

Cache invalidation techniques are used to scale not only to a large number of mobile clients, but also to a large number of data items that are cached in the mobile clients. Caching of these frequently accessed, but less frequently updated items is an effective way to relieve the bandwidth constraints in wirelessenvironments. Wireless Web access is a good example of an application that would greatly benefit from using scalable cache invalidation for rarely updated and cached Web pages. In accessing the Web, users may frequently refresh web pages that contain cached, but rarely updated, data items. The scalable cache invalidation, along with other complementary techniques for the hot spots will be very desirable to support the refresh operations in the wireless web browsing. The BS method is effective for frequently and long disconnected mobile clients [13].

One way to achieve scalability is to increase the granularity of each bit in the BS method. That is, instead of using one bit in the BS structure for each data item, can increase the granularity of bits and use each bit to represent a block of data items. Obviously, the coarse granularity technique may affect the accuracy (or effectiveness) of cache invalidation algorithms when a client has to validate data items that are represented by the same bit of BS structure.

Two scalable BS-based algorithms for cache invalidation: the Multidimensional BS (MD-BS) algorithm and the Multilevel BS (ML-BS) algorithm. The MD-BS algorithm achieves scalability by organizing data items into a multidimensional data structure and using each BS structure to represent the blocks of data items in each dimension.

The ML-BS algorithm provides a scalable solution that applies hybrid and variable granularity of bits and BS structuresto adapt to dynamically changing update rates and patterns.

The ML-BS algorithm adapts well to thechanges of "hot spots" and skewed access and update patterns. The results verify our observation that it is feasible to provide complementary and adaptive techniques to address cache invalidation problems for the data items with varied update rates and patterns [14].

III.Related Work:

Mobile computing environment consists of two distinct sets of entities: mobile hosts and fixed hosts. Some of the fixed hosts, called Mobile Support Stations (MSS) are augmented with a wireless interface to communicate with mobile hosts, which are located within a radio coverage area called a cell [15].

A mobile host can move within a cell or between two cells while retaining its network connections. There is a set of database servers each covering a cell. The database, which consists of N named data items, is fully replicated at each data server. Each server can only service users which are currently located in its cell. A large number of mobile hosts reside in each cell; all issue queries which take the form of simple requests to read the most recent copy of a data item. Itassumes that the database is updated only by the servers.

Each server will periodically broadcast invalidation reports. To answer a query, the client on a mobile host will wait for the next invalidation report and use the report to conclude whether its cache is valid or not. If there is a valid cached copy that can be used to answer the query, the client will return the result immediately. Invalid caches must be refreshed via a query to the server [16].

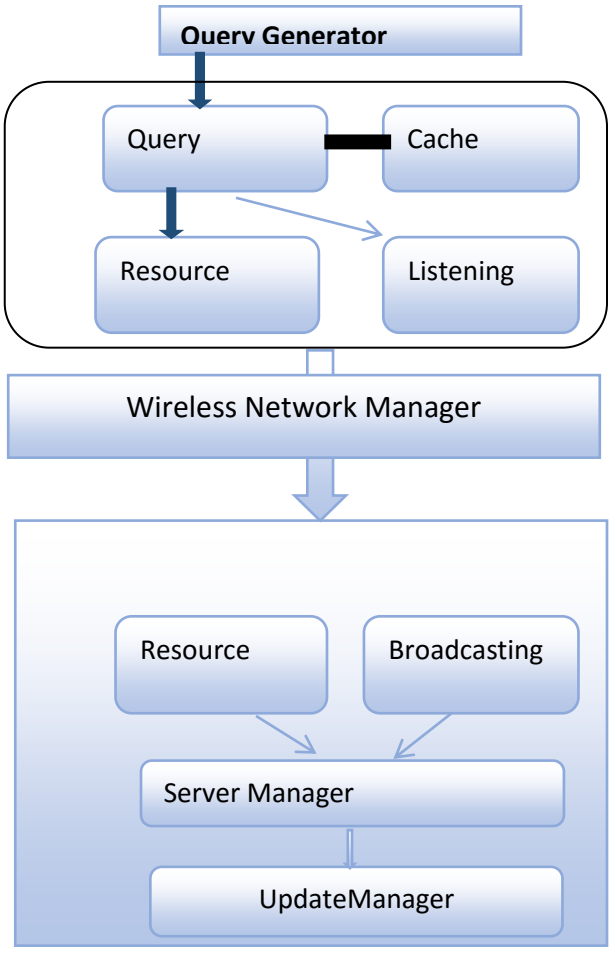

Fig. MOBILE CLIENT MODEL

In this figure, it is assumed that the system consists of a central data server and a fixed number ofclients. As shown in Figure each client comprises five modules. These are a Resource Manager, whichmodels the client CPU time devoted to query processing; a Query Source, which generates query requests;a Client Manager, which executes the query requests, processes disconnection and reconnection operations,and monitors messages received from the server; a Listening Manager, which is in charge of listening forinvalidation reports over the wireless channels; and a Cache Manager, which manages the client cache pool.The central server model has four modules. A Resource Manager models the server CPU time devotedto query and update processing; there is a Server Manager component that coordinates the query requests from clients and update requests from Update Source; the Update Source generates these update requests;and the Broadcasting Manager produces and broadcasts the invalidation reports.Client queries are submitted from mobile hosts. Each query requests only one data item and is submittedindividually to the client. Upon the completion ofone query, the client pauses for a random think time or a random disconnection time and then begins a newquery. For each query, the Client Manager must first call the Listening Manager, which will wait for thenext invalidation report.

The caches will be validated according to the received report. If a valid cache canbe used to answer the query, the client returns the result immediately; otherwise, the query will be forwardedto the server and the cache will be refreshed. A client can enter a disconnection mode only after the current query has finished; if a query is processed at the server, then the client will wait for the return before itcan go into a sleep mode [17].

The parameters that specify the database and physical resources of the system are listed as follows with their meaning:

1.Item Size: Size of a data item in bytes; 2.DatabaseSize: Size of database in data items; 3.NumClients: Number of mobile client hosts; 4.ClientQryTime: Service time for a query in a mobile client; 5.ClientBufSize: Per-client buffer size (% of database); 6.ServerQryTime: Service time for a query in the data server; 7.ServerUpdTime: Service time for an update in the data server; 8.NetworkBandwidth: Wireless network bandwidth; 9.ControlMsgSize: Size of a control message on network, in bytes;

10. Broadcast Period: Broadcast period of invalidation reports, in seconds.

The database is as a collection of Database Size data items of Item Size bytes each. There is one central data server that manages the database. Resource Manager Module is one of the central dataservers,CPU time for processing a query is specified by the parameter ServerQryTime.CPU time is used for processing an update is specified by the parameter ServerUpdTime. All query and update requests are processed with same priority on a first-come first-served. This has been ignored for simplicity [18].

The buffer pool in the central server is large enough to hold the entire database.The number of mobile client hosts is specified by the NumClients parameter. In the case of Resource Manager Module of each mobile client host, CPU time for processing a query is specified by the parameter ClientQryTime. Again, no I/O time is modeled in the Resource module. The size of client buffer pools (ClientBujSize) is specified as a percentage of the database size.

All other messages are of equal priority and will be served on a first-come first-served basis.1.ThinkTime:Think time (in seconds) between queries in a connect mode;2. DisconnectTime: Disconnect time (in seconds) between queries; 3.HotQueryBounds:Data item bounds of hot query range; 4.ColdQueryBounds: Data item bounds of cold query range; 5.HotQueryProb: Probability of querying a data item in the hot query range; 6.UpdateArrTime:Interarrival Time (in seconds) between Updates;7.HotUpdateBounds: Data item bounds of hot update range; 8.ColdUpdateBounds: Data item bounds of cold update range;9.HotUpdateProb: Probability of updating a data item in the hot update range.

Each mobile client host generates a single stream of queries. The arrival of a new query is separated from the completion of the previous query by either an exponentially distributed think time with a mean of ThinkTime or an exponentially distributed disconnection time with a mean of Disconnect Time.

Each client will enter into a disconnection mode once for every three consecutive queries. In other words, each group of three queries will be separated by a disconnection time, and these three queries are separated from each other by a think time. At the server, a single stream of updates is generated. These updates are separated by an exponentially distributed update inter arrival time [19].

IV. Taxonomy of Cache Invalidation Algorithms:

In mobile environment query processing algorithms plays its important role, it affects the complete performance of the system. The main parameters to be considered for performance measures are initial query, exploit query, assign to any possible query and query matching in mobile environment. Query processing in a mobile environment is to form a high level query on different algorithms [20].

There are two competent and scalable BS-based algorithms for cache invalidation: Multidimensional BS (MD-BS) algorithm and the Multilevel BS (ML-BS) algorithm. The MD-BS algorithm achieves scalability by organizing data items into a multidimensional data structure and using each BS structure to represent the blocks of data items in each dimension.

No-Checking Caching Scheme: In Broadcasting Timestamps (TS) and Signatures (SIG) are effective only for clients which have not been disconnected for a period that exceeds an algorithm specified parameter or if the number of updated items during the period is not greater than an algorithm-specified parameter. However, it is possible in these methods that some of the cached objects are still valid after a long disconnection period or a large number of updates at the server. Thus, these methods don't utilize the bandwidth efficiently. Bit-sequences (BS) algorithm improves the finite limitation. The

BS algorithm can approach the "optimal" effectiveness for all data items indicated in the report regardless of the duration of disconnection of the clients. BS also is applied to optimize other broadcast-based cache invalidation algorithms in which the dynamic bit mapping has to be included explicitly. The optimization reduces the size of the report by about one half while maintaining the same level of effectiveness for cache invalidation. However, compared to TS and BS algorithm actually wastes downlink broadcast bandwidth and causes clients to spend more time in active mode.

Checking Caching Scheme : In order to retain valid data items in the cache, we must identify which part of the cache is still valid. There are several approaches to this problem with different trade-offs. Such as Simple-checking caching scheme, Simple-grouping caching scheme and Grouping with cold update-set report. The third solution is that the mobile host sends back to the server the ids of all the cached data items and their corresponding timestamps. Then the server identifies which data items are valid and returns a validity report to the client. However, this requires a lot of uplink bandwidth and is not power efficient.

Adaptive Invalidation Report Scheme: This scheme contains two parts, adaptive invalidation report with fixed window and adaptive invalidation report with adjusting window. The former method guarantees that BS is broadcast as the next invalidation report the uplink bandwidth required by this method is much smaller than that of checking caching schemes. The latter method integrates TS, varying window size and BS. The adaptive methods make use of workload information from both clients and server so that the system workload has less impact on the system performance while maintaining low uplink and downlink bandwidth requirement. Furthermore, they achieve good balance between throughout and uplink bandwidth required.

Selective Cache Invalidation Scheme: There are there methods, group-based cache invalidation, hybrid cache invalidation and selective cache invalidation. The group-based cache invalidation scheme broadcasts a group invalidation report while the hybrid cache invalidation scheme and selective cache invalidation scheme broadcast a pair of invalidation reports. All the schemes allow clients to selectively tune to the portion of the invalidation reports.

V. PERFORMANCE ANALYSIS OF CACHE INVALIDATION ALGORITHMS:

Algorithms	Techniques	Performance	Result	Pros	Cons
Bit-Sequence: **The BS method is effective for frequently and long disconnected mobile clients.**	1. Bit Sequence Naming. 2.Update Aggregation 3. Hierarchical structure of bit sequence.	It measures how to compare pause effects of different workload factors aretime, query update rate.	The results of a set of show the regards to the query execution time.	1. It Provides a better complexity. 2. It Suitable during long distance.	1. Lack of concern on outer location factors that are time, place and purpose.

Multidimensional BS: **This algorithm is achieves scalability by organizing data items into multidimensional data structure.**	Each BS structure to represent the blocks of data items in each dimension.	The performance measure of query processing will be conducted in terms of query execution time.	Database is irrelevant planned in multiple dimensional array data structure.	It can equally increase the granularity of each bit in the invalidation report instead of growing the number of bits.	This algorithm in which the sizeof invalidation report does not increase linearly proportional to the size of database.
Multilevel BS: **The ML-BS algorithm is provides a scalable solution that applies hybrid and variable granularity of bits.**	The adaptive technique is used to address cache invalidation.	The ML-BS algorithm adapts well to the changes of hot spots and skewed access and update patterns.	The results verified the observation that it is feasible to provide complementary.	BS structures to adapt to dynamically changing update rates and patterns.	The adaptive techniques to address cache invalidation problems for the data items with varied update rates and patterns.
Broadcasting TS: It includes only the information regarding the data items.	The timestamp ordering technique enjoys number of benefits from flexible adjustment of serialization order.	TS is effective only for clients which have not been is connected for a period.	It includes the names of items and the timestamps at which they were updated.	It updated within the preceding seconds. It exploiting the semantics of read-only transactions, which comprise most of the existing applications.	The TS algorithm is not more effective than BS.
Signatures (SIG): It contains a set of combined signatures of data items.	The structure and size of these signatures are designed to diagnose up to differing items.	To demonstrate therelation between the actual number of differing items and the number of items to be invalidated.	It computes combined signatures for two databases with different data items.	The use the SIG algorithm to generatethe data items to be invalidated.	The most items cached by the clients will be invalidated many are in fact valid.

VI.ANALYSIS AND DISCUSSION:

The performance analysis and discussion of various competent algorithms for query processing in mobile environment on various parameters that are given as follows:

The performance and how to compare the effects of different workload parameters, such as disconnection time and query update rate, on the relative performance of the MD-BS and ML-BS algorithms. The performance metrics is the cache hit ratio of clients after disconnection. The cache hit ratio is computed by dividing the sum of the queries that are answered using client caches by the sum of the total queries. The HotQueryProb parameter specifies the probability that a query will address a data item in the client's "hot" database region. Within each region, data items are chosen through a uniform distribution. For the data server, the HotUpdateBounds and ColdUpdateBounds parameters are used to specify the "hot" and "cold" regions, respectively, for update requests. The HotUpdateProb parameter specifies the probability that an update will address a data item in the update's "hot" database region.

Broadcasting TS Algorithm: It includes only the information regarding to the data items. The timestamp ordering technique enjoys number of benefits from flexible adjustment of serialization order. TS is effective only for clients which have not been is connected for a period. It includes the names of items and the timestamps at which they were updated.

Signatures (SIG) Algorithm: It contains a set of combined signatures of data items. The structure and size of these signatures are designed to diagnose up to differing items. The relation is generated between the actual number of differing items and the number of items to be invalidated.

VII. Conclusion:

The algorithms are performs consistently well under conditions of variable update patterns and client disconnection times. Furthermore, the size of the invalidation report is in these algorithms are relatively small and is independent of the number of data items to be invalidated. Cache invalidation techniques are used to scale not only to a large number of mobile clients, but also to a large number of data items that are cached in the mobile clients. Caching of these frequently accessed, but less frequently updated items is an effective way to relieve the bandwidth constraints in wireless environments. Bit sequence algorithm measures how to compare pause effects of different workload factors are time, query update rate.

Multi-dimensional BS algorithm can measure performance of query processing will be conducted in terms of query execution time. Multilevel-BS algorithm is changing workload parameters, such as client disconnection times and update patterns. Broadcasting TS algorithm is effective only for clients which have not been connected for a period. Signatures algorithm computes combined signatures for two databases with different data items. The proposed performance analysis of three new bit sequence cache invalidation algorithms and two old cache invalidation algorithms and different techniques and their respective results with their benefits and drawbacks also.

Finally figure out that the cache invalidation algorithms till date are significant but considering it from the future prospective there is need for making advancement in cache invalidation algorithms in mobile environment.

References

1. Soumi Chattopadhyay, Ansuman Banerjee,"QSCAS : QoS Aware Web Service Composition Algorithms With Stochastic Parameters", IEEE International Conference on Web Services,pp.1-8,2016.
2. Zhou Z.,Zhao D.,Hancke G.,Shu L. and Sun Y.,"Cache-Aware Query Optimization in Multi application Sharing Wireless Sensor Networks", IEEE transactions on systems, man, and cybernetics: systems 1, pp. 1-17, July-2016.
3. Chatzimilioudis G., Costa C., Yazti D.Z., Lee W.C. and Pitoura E., "Distributed In-Memory Processing of All k Nearest Neighbor Queries", IEEE transactions on knowledge and data engineering, Vol. 28, No. 4, pp.925-938, April 2016.
4. Kolcun R., D.E. Boyle and McCann J.A., "Efficient Distributed Query Processing", IEEE transactions on automation science and engineering, Vol. 13, No. 3, pp.1230-1246, July 2016.
5. Li R., Liu A.X., Wang A.L. and Bruhadeshwar B.,"Fast and Scalable Range Query Processing With Strong Privacy Protection for Cloud Computing", IEEE/ACM transactions on networking, Vol. 24, No. 4, pp.2305-2318, August 2016.
6. Zhang W.N., Ming Z.Y., Zhang Y., Liu T., and Chua T. S., "Capturing the Semantics of Key Phrases Using Multiple Languages for Question Retrieval", IEEE transactions on knowledge and data engineering, Vol. 28, No. 4, pp.888-900, April 2016.
7. Fegaras L. "Incremental Query Processing on Big Data Streams", IEEE transactions on knowledge and data engineering, Vol. 28, No. 11, pp.2998-3012, November 2016.

8. Godfrey P., Gryz J., and Lasek P., "Interactive Visualization of Large Data Sets", IEEE transactions on knowledge and data engineering, Vol. 28, No. 8, pp.2142-2157, August 2016.
9. Caruccio L., Deufemia V. and Polese G., "Relaxed Functional Dependencies: A Survey of Approaches", IEEE transactions on knowledge and data engineering, Vol. 28, No. 1, pp.147-165, January 2016.
10. Shen C.Y., Yang D.N., Huang L.H., Lee W.C. and Chen M.S., "Socio-Spatial Group Queries for Impromptu Activity Planning", IEEE transactions on knowledge and data engineering, Vol. 28, No. 1, pp.196-210, January 2016.
11. Chen M., Huang S., Fu X., Liu X. and He J. "Statistical Model Checking-Based Evaluation and Optimization for Cloud Workflow Resource Allocation", IEEE transactions on cloud computing, pp.1-14, 2016.
12. Miao X., Gao Y., Zheng B., Chen G. and Cui H., "Top-k Dominating Queries on Incomplete Data", IEEE transactions on knowledge and data engineering, Vol. 28, No. 1, pp.252-266, January 2016.
13. Wan L., Tang K., Li M., Zhong Y. and Qin A.K. "Collaborative Active and Semi supervised Learning for Hyper spectral Remote Sensing Image Classification", IEEE transactions on geoscience and remote sensing, Vol. 53, No. 5, pp.2384-2396, May 2015.
14. Mi young Jang,Sune-Je Jang,Jae-Woo Chang,"A New K-NN Query Processing Algorithm Enhancing Privacy Protection In Location Based Services", IEEE First International Conference on Mobile Services,pp.1-8,2012.
15. Ing Ray Chen, Anh Phan Speer, Mohamed Eltoweissy,"Adaptive Fault-Tolerant Qos Control Algorithms For Maximizing System Lifetime Of Query-Based Wireless Sensor Networks", IEEE transactions on dependable and secure computing, Vol. 8, No. 2, pp.161-176, March-April 2011.
16. Jiaheng Lu,Tok Wang Ling,Zhifeng Bao,Chen Wang,"Extended XML Tree Pattern Matching: Theories And Algorithms",IEEE transactions on knowledge and data engineering, Vol. 23, No. 3,pp.402-416, March 2011.
17. Fan Yuanyuan Mi Xifeng "Distributed Database System Query Optimization Algorithm Research", IEEE,pp.1-4,2010.
18. Ke Yi,Feifei Li,George Kollios,Divesh Srivastava,"Efficient Processing Of Top-K Queries In Uncertain Databases With X-Relations", IEEE transactions on knowledge and data engineering, Vol. 20, No. 12,pp.1669-1682, December 2008.
19. Sudipto Guha,Kyuseok Shim,"A Note on Linear Time Algorithms for Maximum Error Histograms",IEEE transactions on knowledge and data engineering, Vol. 19, No. 7, pp.993-997, July 2007.
20. Ahmed Elmagarmid, Jin Jing, Abdelsalam (Sumi) Helal, and Choonhwa Lee, "Scalable Cache Invalidation Algorithms For Mobile Data
21. Access", IEEE transactions on knowledge and data engineering, Vol. 15, No. 6, pp. 1498-1511, November/December 2003.