센서 네트워크 환경에서 최적화된 분산 R-tree를 이용한 에너지 인식 질의 처리 방법

수라즈 반데† · 어 상 훈†† · 김 호 석††† · 배 해 영††††

요 약

본 논문에서는 센서 네트워크 환경에서 최적화된 분산 R-tree를 사용하여 공간 범위 질의 처리시 센서들의 에너지 소모를 최소화하는 방법을 제안한다. 제안된 기법은 센서 네트워크를 이용하는 공간 범위 질의 처리시 센서들의 공간상의 위치에 대한 색인을 이용하는 새로운 방법이다. 최근들어 센서 네트워크 환경에서의 공간 범위 질의는 특정 지역에 대한 센서 노드들의 집계 값을 계산하는 방법으로 더욱 중요시 되어지고 있다. 기존 연구들은 공간 범위 질의 처리의 중요성을 많이 언급을 하였지만 현재까지 이에 대한 효율적인 방법에 대해서는 제안하지 못하고 있는 실정이다. 제안된 기법에서 센서 네트워크 상의 각각의 센서 노드들은 자신과 자신의 자식 노드들의 위치를 포함하는 MBR을 갖는다. 공간 범위 질의는 제안하는 분산 R-tree를 기반으로 센서들의 공간상의 위치와 질의 범위가 서로 겹치는 지역에 대하여 평가된다. 이러한 접근 방법은 공간 범위 질의에 대한 평가를 수행함에 있어 참여하지 않는 불필요한 노드들과의 통신을 방지하여 센서 노드들의 에너지 소모를 최소화한다.

키워드 : 센서 네트워크, 질의 처리, 공간 색인

Power-Aware Query Processing Using Optimized Distributed R-tree in Sensor Networks

Suraj Pandey† · Sang Hun Eo†† · Ho-Seok Kim††† · Hae-Young Bae††††

ABSTRACT

In this paper, a power-aware query processing using optimized distributed R-tree in a sensor network is proposed. The proposed technique is a new approach for processing range queries that uses spatial indexing. Range queries are most often encountered under sensor networks for computing aggregation values. The previous work just addressed the importance but didn't provide any efficient technique for processing range queries. A query processing scheme is thus designed for efficiently processing them. ·Each node in the sensor network has the MBR of the region where its children nodes and the node itself are located. The range query is evaluated over the region which intersects the geographic location of sensors. It ensures the maximum power savings by avoiding the communication of nodes not participating over the evaluation of the query.

Key Words : *Sensor Networks, Query Processing, Spatial Index*

1. INTRODUCTION

A sensor network consists of many spatially distributed sensors, which are used to monitor or detect phenomena at different locations, such as temperature changes or pollutant level. Sensor nodes, such as the Berkeley MICA Mote[6] which already support temperature sensors, a magnetometer, an accelerometer, a microphone, and also several actuators, are getting smaller, cheaper, and able to perform more complex operations, including having mini embedded operating systems. There are thousands of different ways that motes might be used, and as people get familiar with the concept, they come up with even more sophisticated scenario. It is a completely new paradigm for distributed sensing and it is opening up a fascinating new way to look at computers.

While these advances are improving the capabilities of sensor nodes, there are still many crucial problems with deploying sensor networks. Limited storage, limited net-

---

※ 본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT연구 센터 육성 · 지원사업의 연구 결과로 수행되었음.

† 준 회 원 : 인하대학교 대학원 컴퓨터·정보공학과 석사과정
†† 준 회 원 : 인하대학교 대학원 컴퓨터·정보공학과 통합과정
††† 준 회 원 : 인하대학교 대학원 컴퓨터·정보공학과 박사과정
†††† 종신회원 : 인하대학교 컴퓨터공학부 교수

논문접수 : 2005년 9월 2일, 심사완료 : 2006년 1월 5일

work bandwidth, poor inter-node communication, limited computational ability, and limited power still persist.

Energy is the most critical factor in designing sensor networks. One way to help improve the problem of limited power is through the in-network query processing rather than query processing at the base station. Analytical result proves that communicating 1 bit over the wireless network is equivalent to performing approximately 1000 CPU instructions[9]. Therefore, any solution devised for sensor networks has to minimize the amount of communication overhead imposed on the network.

For power efficient query processing, in-network aggregation has been widely adopted. Spatio-temporal aggregation, data centric techniques like directed diffusion [17] etc. have been proposed. In addition, the importance of knowing results over a fixed region is drawing more attention. Range queries provide the aggregated values over the region of interest. However to facilitate such requirements, a good design is needed that must work under the constrained environment.

In this paper, we propose the use of a distributed version of conventional R-tree[2], with specifications to work under the constraints of individual sensor node, in facilitating the spatial querying over the region of interest in the deployed sensor network. The concept of node traversal and selection adopts the similar algorithms as laid out for the R-tree structure. As R-tree is the primary choice when handling spatial attributes efficiently, almost all index structures are motivated from it. We follow the TiNA[4] approach, which uses the temporal coherency tolerances to reduce energy consumption and at times increase the quality of data. Our approach is to minimize the power utilization in communication by using the capability of each node to store and manipulate the data.

The remainder of this paper is organized as follows. In Section 2, we briefly review related work. In Section 3, we lay out the assumption and the system model. In Section 4, we propose the structure and power-aware query processing using the distributed R-tree. Section 5 presents the performance evaluation based on the simulated environment. Finally, we conclude in Section 6 providing insights into future works.

2. RELATED WORK

In this section, we discuss related work from both the database and sensor networking communities. Although many literatures have laid out the importance to the need of spatial indexing schemes, there is no other work that we are aware of that proposes a generic, range query-based scheme for extracting data from sensor networks.

With respect to aggregation, the semantics used here are largely a part of the SQL standard[5]. The design is also compliant with the existing OGC spatial extension. In the previous work[14, 15, 17], grouping computes aggregates over partitions of sensor readings. The basic technique for grouping is to push down a set of predicates that specify the group membership, ask sensors to choose the group they belong to, and as answers flow back, update the aggregate values in the appropriate groups. In this paper the sensor nodes are grouped according to their regions.

The Cougar project at Cornell[10] discusses queries over sensor networks, as does the work on Fjords[7]. It only considers moving selection operators onto sensors and does not present a specific, power-sensitive algorithms related to spatial grouping for use in sensor networks. Madden et.al., in[8] proposed TAG, an aggregation service as a part of TinyDB1)[14] which is a query processing system for a network of Berkeley motes. It presents the in-network processing of the aggregation queries on the data generated in the sensor network. The necessity of computing spatial aggregates in large clusters of nodes has been addressed but no solutions have emerged. The work on directed diffusion[17] discusses techniques for moving specific pieces of information from one place in a network to another, and proposes aggregation-like operations that nodes may perform as data flows through them. A scheme for imposing names onto related groups of sensors in a network was also proposed[16], in much the same way our scheme groups sensor nodes into regions according to their geographic location. The TinyOS group at UC Berkeley has published a number of papers describing the design of motes, the design of TinyOS[20], and the implementation of the networking protocols used to conduct ad-hoc sensor networks. Ratnasamy et.all described a novel Geographic Hash Table (GHT)[19] system which hashes keys into geographical coordinates. In GHT the data is stored at a node with location determined by a geographical hash function of its name. The advantage of this system is that it allows to lookup the location of data by its name. However, the problem associated is that the location defined by a GHT function can be quite far from the data source. Also, spatially related data might become scattered across the network. If we use hashing based on grid cells, we may look at a good number of points that

---

1) http://telegraph.cs.berkeley.edu/tinydb/

are not answers to the query. Even more, for range queries we may be required to examine many buckets.

However, these work just point out the necessity and do not directly address spatial data collection or aggregation issues.

3. ASSUIMPTIONS AND SYSTEM MODEL

Although there have been many advances in sensor network applications and technology, sensors still suffer from the major problems of limited bandwidth and have energy constraints. We describe our model for sensor networks and sensor data, and outline our architectural assumptions. Sensor networks have the following physical resource constraints:

Communication: The wireless network connecting the sensor nodes is usually limited, with only a very limited quality of service, with high variance in latency, and high packet loss rates.

Power consumption: Sensor nodes have limited supply of energy, most commonly from a battery source, and thus energy conservation needs to be the main system design considerations.

Computation: Sensor nodes have limited computing power and memory sizes that restrict the types of data processing algorithms that can be used and intermediate results that can be stored on the sensor nodes.

We consider static sensor nodes placed in the network that is distributed over a large area. All sensors are aware of their geographical position. Each sensor could be equipped with GPS device or use location estimation techniques. For cost effective solution, we can assume that the locations of the sensors are known a priori during the initializing the network. The sensors store their position information as directed by the base station.

In this paper, we present the distributed tree and the query processing scheme using the Cougar approach[10] in solving the synchronization problem: a parent sensor node will keep a list of all its children, which is called the *waiting list,* and will not report its reading until it hears from all the sensor nodes on its waiting list.

4. DISTRIBUTED R-TREE

In this section, we propose the distributed R-tree used for querying with spatial attributes. All the schemes reviewed earlier are based on grouping of the sensor nodes either by event/attribute, which are data centric. The techniques demand communication that is redundant. If range queries are to be forwarded, almost all of the sensor nodes are populated with messages. Our query processing scheme with the distributed tree version overcomes these inherited deficiencies of the previous work.

4.1 THE TREE STRUCTURE

A distributed R-tree is simply an index tree designed to allow each node to efficiently determine if any of the nodes below it will need to participate in a given query over some queried range.

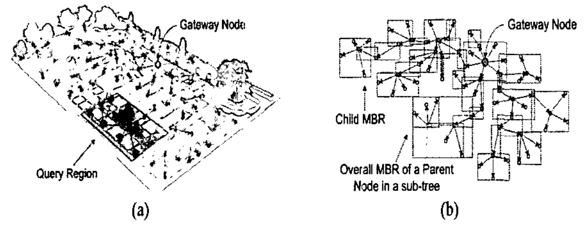
The nodes of the tree are linked together following the routing protocol used by the underlying sensor network which determines the parent-child relationship and their extents. We would recommend using the tributary-delta approach[3]. However, to accommodate the spatial query in the network we need additional parameters to be stored by individual nodes. Each node must store the calculated[2] MBR (Minimum Bounding Rectangle) of its children along with the aggregate values as have already been existing in each node under the in-network query processing paradigm and noted by several literatures[14, 4, 13].

The parent node of each region in the tree has a structure in the form <*child-pointers, child-MBRs, overall-MBR, location-info*>. The *child-pointers* helps traverse the node structure. As we are following the Cougar, the *waiting-list* carries the same semantics as these pointers. In addition, we have added the MBR in each node which confines the children into a box over which a query can be made. The confinement algorithm is responsible to analyze and distribute the sensor nodes into the appropriate MBR. This classification is largely based on their proximity to their respective parent and the contribution factor to the dead space of the resulting MBR. Other promising factors can be explored and analyzed, which we consider for our future work, and basically orthogonal to the present discussion. However, it is this classification that brings about efficient routing and accuracy to the queried result.

(Figure 1) shows the simulated environment setting consisting of distributed sensor nodes on which we base our experiments. For the construction of distributed R-tree, in the descending stage, a bounded box which overlaps the children and the parent itself should be stored by each parent in that region. This bounded box(partial MBR) assists each of the nodes to classify the nodes according

---

[2] Each MBR is updated during the ascending of the tree so that the modified MBR is stored in each node.



(Figure 1) Node positions in one section of our sensor test bed. (a) Simulated Physical Environment showing region of interest. (b) The MBR under each parent node of a sub tree

to their geographical locations. Each descent correspondingly stores the MBR of the region where there exists parent-child relationship until the leaf node is reached. At the end of the descent, when all the nodes have been traversed, the parent node of each region is notified about their child node MBR. Hence, in the ascending stage the parent of each region gets updated the new MBR of their children which now should include the sub-tree under that node, and the proposed structure is formed among the sensor nodes.

4.2 ENEGY EFFIECIENT & POWER-AWARE QUERY PROCESSING

One critical operation of our scheme, called *energy efficient forwarding*, is to isolate the regions containing the sensor nodes that can contribute to the range query. This operation behaves much like the flooding scheme if the queried range demands high computation and expensive searches to determine the region of interest and isolate the sensor nodes. Also, if the classification is not suitable, some extra node traversal occurs inadvertently. We assume such cases to be rare in widely spread sensor networks. Our prime objective is to maintain the minimum count of nodes taking part in the query. As we explained in section 4.1, the construction of the distributed tree determines the ease of forwarding the query to pin point the sensor nodes.

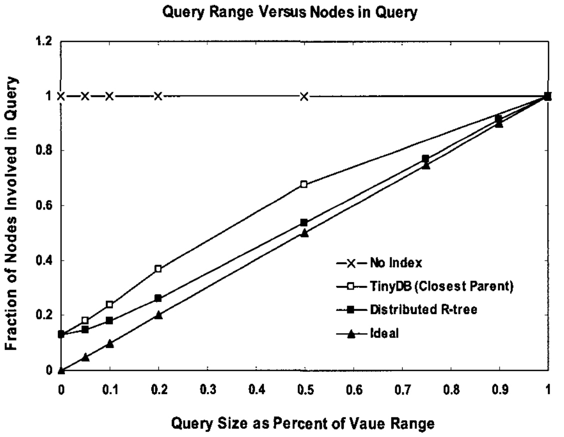
We deal with two approaches for answering any range query. First, considering that the user requires strictly accurate data over the queried region, we leave no room for error and hence forward the query to only the relevant sensors in the region of interest. Second, we propose a more relaxed approach where the user is searching for only approximations, as is most generally the case over sensor networks considering the topological redundancy. The former requires more communication as compared to the latter solution. Again, it is orthogonal to us to point out the types of aggregation queries that would yield more probable results adopting either of these approaches. We leave this to be analyzed in our future work.

A *range query* returns all the relevant data collected/relayed that are associated with regions within a given query window W (e.g., a rectangle in a two-dimensional space). To process a range query in accordance to our design, at first the root node receives the query. The disseminating of this request to the nodes on the sub-tree now is based on the calculation of the child node/s whose *overall-MBR* overlaps W. Each parent under that overlapping region receives this query and based on the overlapping regions of its children, the corresponding network (sub-tree) is flooded. It is here that the *child-MBR* is used to decide the particular regions which need precise selection in-order to limit unnecessary node traversal. These *child-MBRs* are comparatively small regions that cover only the perimeter of the children including their parent. So the selection operation needs minimum traversal to include the nodes in the list needed for range query. The optional parameter *location-info* should help to get accurate result for overlapping, independent regions in the sensor network environment. Its inclusion is partly based on the type of sensor network and its extensiveness. In addition to the geographic information it may include additional values e.g., time *t*, location attributes etc., that should act as a filter, which again is largely dependent on the computational power of each sensor node. The query dissemination operation descends the tree from the root in a manner similar to the R-tree. Each node searches for its child nodes that need to participate in the range query.

5. PERFORMANCE EVALUATION

To study the performance of the proposed scheme in sensor networks, we created a simulation environment using AVRORA[12]. Following typical sensor network simulation practices, the simulated network was configured as a grid of sensors. Each node could transmit data to sensors that were at most one hop away from it. In a grid this means it could only transmit to at most 8 other nodes. We used a contention-based MAC protocol(PAMAS) which avoids collision[11]. In this protocol, a sender node will perform a carrier sensing before initiating a transmission. If a node fails to get the medium, it goes to sleep and wakes up when the channel is free.

In the experiments we evaluated the performance of our proposed scheme, the distributed R-tree, against the *best-case* approach and *closest parent* as used by TinyDB[14].



(Figure 2) Number of nodes participating in range queries of different sizes(20 × 20 grid, 400 nodes)

We used the random distribution to select the query range. (Figure 2) shows the number of nodes that participate in queries over variably sized range queries. It is drawn over the average values obtained after the simulation. The number of nodes that are involved in the query is reduced in comparison to the closest parent approach of TinyDB in its SRT[14]. It is due to the fact that the unnecessary messages forwarded to irrelevant nodes are drastically reduced. In the graph analyzing the performance against TinyDB, we have deliberately excluded the in-network aggregation so as to make a fair comparison. Nevertheless, our approach is close to the ideal number of nodes. In order to emphasize the effectiveness of storing the partially aggregated value for in-network aggregation and thus to reduce the power utilization, we used the TiNA scheme. So it is obvious that this scheme should perform better than the stand-alone distributed R-tree with just TinyDB in terms of consumption of power.

As we can visualize the difference in node selection in TinyDB with our design from the graph, we can readily conclude that our approach is up to 20% more energy efficient.

6. CONCLUSION AND FUTURE WORK

In this paper, we contribute a new technique to group the sensors in a region for spatial range queries. We proposed the distributed R-tee for power-aware query processing in sensor networks.

Our design can reduce the number of nodes that disseminate queries by nearly an order of magnitude. Isolating the overlapping regions of sensor nodes with the range query, non-relevant nodes can be avoided in the communication. Only the sensor nodes leading to the path of the requested region are communicated, and hence substantial reduction in power is achieved due to reduced number of sub-trees involved. In addition, the aggregate values for the region of interest is collected, following the in-network aggregation paradigm which has an advantage over the centralized index structure in that it does not require complete topology and sensor value information to be collected at the root of the network. Since data transmission is the biggest energy-consuming activity in sensor nodes, using the distributed tree results in significant energy savings.

In conclusion, our query processing design using the distributed version of R-tree provides a scalable solution to facilitate range queries adopting similar protocols and query processing used so far, making it highly portable. Currently, we are expanding our scheme to consider moving objects trying to achieve moreover the same throughput as in static networks. Our work on clustering the nodes based on several factors contributing to the performance of the network is still underway. For moving sensors, the obvious cost of tree maintenance can be handled by introducing constraints over the trajectory of the sensor nodes and over the tree's structure. We can argue that like in R+-tree, super nodes or redundant architecture should limit the overhead of communication to all the parent nodes. The approximation over the moving object's trajectory is just another idea that poses a separate research direction on itself. Adoption of distributed redundant architecture for efficient processing of concurrent queries and for supporting join operations, are challenges which are under scrutiny as the capabilities of sensor nodes reaches higher levels.

REFERENCES

[1] S.Bapat, V.Kulathumani, and A.Arora, Analysing the Yield of ExScal, a Large-Scale Wireless Sensor Network Experiment. OSU-Technical Report OSU-CISRC-6/05- TR46. (To appear, 13th International Conference on Network Protocols(ICNP)2005).

[2] A. Guttman, R-Trees : A Dynamic Index Struc-ture for Spatial Searching. In Proceedings of ACMSIGMOD 1984, Annual Meeting, Boston, USA, pages 47-57. ACM Press, June, 1984.

[3] A.Manjhi, S.Nath, P.B.Gibbons, Tributaries and Deltas: Efficient and Robust Aggregation in Sensor Network Streams. In SIGMOD 2005 June 14-16, 2005, Baltimore,

Maryland, USA.

[4] M. A. Sharaf, J. Beaver, A. Labrinidis, P.Chrysanthis, TiNA: A Scheme for Temporal Coherency-Aware in-Network Aggregation. In Proceedings of 2003 International Workshop in Mobile Data Engineering.

[5] ANSI. SQL Standard, 1992. X3. 135-1992.

[6] J.Hill and D.Culler, Mica: A wireless platform for deeply embedded networks. IEEE Micro., 22(6) : 12-24, Nov/Dec., 2002.

[7] S.Madden and M.Franklin, Fjording the Stream : An Architecture for Queries over Streaming Sensor Data. In ICDE(2002).

[8] S.R.Madden, M.J.Franklin, J.M. Hellerstein, and W.Hong, TAG : a Tiny AGgregation Service for Ad-Hoc Sensor Networks, OSDI, Dec., 2002.

[9] J.HILL, R. Szewczyk, A. Woo, S. Hollar, and D.C.K. Pister, System architecture directions for networked sensors. In proceedings for the 9th International Conference on Architecture Support for Programming Languages and Operating Systems, November, 2000.

[10] Y.Yao and J.Gehrke, The Cougar Approach to In-Network Query Processing in Sensor Networks, SIGMOD'02.

[11] S.Singh and C.Raghavendra, PAMAS : Power aware multi-access protocol with signaling for ad hoc networks. ACM Computer Comm. Review, 28(3).

[12] AVRORA, a research project of the UCLA Compilers Group, http://compilers.cs.ucla.edu/avrora/index.html.

[13] J.Beaver, M.A.Sharaf, A.Labrinidis, and P.K.Chrysanthis, Power-Aware In-Network Query Processing for Sensor Data, Proceedings of the 3rd ACM MobiDE Workshop, September, 2003.

[14] S.R.Madden, M.J.Franklin and J.M.Hellerstein, TinyDB: An Acquisitional Query Processing System for Sensor Networks. ACM Transasctions on Database Systems, Vol.30, No.1, pp.122-173, March, 2005.

[15] S.Madden, R.Szewczyk, M.J.Franklin and D.Culler, Supporting Aggregate Queries Over Ad-Hoc Wireless Sensor Networks. Et al-2002 ACM DBLP.

[16] J.Heidemann, F.Silva, C.Intanagonwiwat, R.Govindan, D. Estrin, and D.Ganesan, Building efficient wireless sensor networks with low-level naming. In SOSP, October, 2001.

[17] C. Intanagonwiwat, R. Govindan and D. Estrin, Directed Diffusion : A Scalable and Robust Communication Paradigm for Sensor Networks, ACM MobiCom'00.

[18] J.Hill et al., System architecture directions for network sensors. In ASPLOS 2000.

[19] Ratnasamy, S., Karp, B., Yin, L., Yu, F., Estrin, D., Govindan, R., and Shenker, S., GHT: A Geographic Hash Table for Data-Centric Storage. First ACM International Workshop on Wireless Sensor Networks and Applications (WSNA 2002), Atlanta, GA, September, 2002.

수라즈 반데

e-mail : suraj@dblab.inha.ac.kr
2003년 Institute of Engineering, Tribhuvan University(B.E.)
2005년~현재 인하대학교 대학원 컴퓨터·정보공학과(석사과정)
관심분야 : Stream DSMS, Sensor Network, 유비쿼터스 컴퓨팅, GRID

어 상 훈

e-mail : eosanghun@dblab.inha.ac.kr
2003년 인하대학교 컴퓨터공학부(공학사)
2003년~현재 인하대학교 대학원 컴퓨터·정보공학과(통합과정)
관심분야 : 다중레벨 데이터베이스, LBS, 유비쿼터스 컴퓨팅, RFID 미들웨어

김 호 석

e-mail : bluesnow@dblab.inha.ac.kr
2001년 인하대학교 전자계산공학과(공학사)
2003년 인하대학교 대학원 전자계산공학과(공학석사)
2003년~현재 인하대학교 대학원 컴퓨터·정보공학과(박사과정)
관심분야 : 공간데이터베이스, GIS, LBS

배 해 영

e-mail : hybae@inha.ac.kr
1974년 인하대학교 응용물리학과(공학사)
1978년 연세대학교 대학원 전자계산학과(공학석사)
1989년 숭실대학교 대학원 전자계산학과(공학박사)
1985년 Univ. of Houston 객원교수
1992년~1994년 인하대학교 전자계산소 소장
1982년~현재 인하대학교 컴퓨터공학부 교수
1999년~현재 지능형GIS연구센터 센터장
2000년~현재 중국 중경우전대학교 대학원 명예교수
2004년~현재 인하대학교 정보통신대학원 원장
관심분야 : 분산 데이터베이스, 공간 데이터베이스, 지리정보 시스템, 멀티미디어 데이터베이스 등