Optimized Query Processing for Wireless Sensor Networks

Akbar Bemana

Abstract— in this paper we propose an approach for query optimization in Sensor networks that cause increasing of network life time. Proposed approach reduce network consumption energy via executing operators with optimized order and optimized method for executing aggregation functions that sensor nodes do not sent unimportant data to their parents. But other existing methods such as PA and PM are sent all data to base station and base station filters data.

Keywords: sensor networks; query optmization; operators; aggregation functionstuple; root node

1. INTRODUCTION

Recently sensor networks are used in various domains of computer applications. Infinite property and scalability of them give chance to getting around information in the highest resolution that impossible in previous technologies. They are used from military search to environmental control and home application and health care. Number of sensor nodes that are distributed in target region according to their application are 1000 or 1000000 [13]. Obvious that interfering and controlling of human being at all time is impossible. And self-organization ability is not isolation property of them. In these networks, nodes produce data from around visited events. Now from these networks are called strongly distributed database. Producing of data and sending it in sensor networks is similar to saving and query processing in conventional database. Although infrastructure property and data properties of sensor networks is different with a central database system complementally [1]. We use declarative query in interaction networking. Declarative query playas an interface role. User able access to data from this interface although directory management, query optimization, processing techniques, sensors physical details and sending result to user methods is hidden from user. Since sensor networks as database provide appropriate logical abstract for managing sensor data. SQL is an instance of declarative query language. Because sensor networks need support from longing executes and provide query [9, 11], we use a language similar to SQL with a few different. For accomplishing private requirements of this networks a few research projects such as Tiny DB [6] and Cougar [7], extends and implements a query based interface for sensor networks. In sensor networks nodes are always involved in processing resource and serve limited energy.

- *Akbar Bemana* Islamic Azad University Dashtestan branch, Borazjan, Iran

Since central programs that requires bigger processing and connecting are not fit for these networks. For example using a central application that all nodes send frequently their data to a base station is not possible. Because communication between nodes reduces power of networks most recent research have been focused on query processing strategies to reduce communications and propagated data in networks [12, 8]. Consecutively conservation of energy is most important parameter in sensor networks, because error in a node reducing battery power and this query in sensor networks. Section3 discuses database operators and effect ions of them in network energy consumption. In section4 we propose a method for ordering database operators. In section5, we will discuss aggregation queries. In section6, we propose a method for executing aggregation functions. Section7 is related work and section8 is conclusion.

```
SELECT      {attributes , aggregates}
FROM        {Sensor-data S}
WHERE       {condition-of-attributes}
GROUP BY    {attributes}
HAVING      {condition-of-aggregates}
DURATION    time interval
EVERY       time span e
```

Figuree 1. Query Template

2. Preliminaries

A. *DATA MODEL QUERY LANGUAGE*

In our plane user queries are injected to networks on simple language template similar to SQL. General form of use queries is shown in Figure 1 It clears that defined queries in sensor networks has a few different with usual SQL language. Nodes sample from around environment at epoch points. Query length epoch points to time between two sequential time sampling. In Figure 1 duration defines query life time and every of them defines sampling period [14]. For example in “select node id “query, user request node number, light, diagnostic temperature of each node in 1 second duration for 10 second network life time. This view of sensor networks as database has considerable grown recently. Each sensor node acts as autonomous data resource. Different nodes data same schema and together forms a relational distributed table. At every time each node makes a tuple of this table. A set of similar recorded tuples from a sensor group form a snapshot. In database literature, this table snapshot forms a connectivity that divided between group nodes vertically [1, 2].

A. Query optimization

System can select on order operator from many ordered operators executing. Procedure of select best possible plan is named query optimization. One general view query optimization is done with mentioning a set of possible planes and determines cost for each plane and selects a plane with the lowest cost. Large mass of this optimization often is done in base station before query propagation in network. Because it possible that base station dose not has complete information of sensor networks state. At times query optimization is necessary after query injection to network.

B. Query propagation and result gathering

After query optimization system inject query to networks. Query form simple flow propagation will arrive to all network nodes. Simultaneous with query propagation, nodes get information about their neighbors and each node in networks select a node that one step near to root as parent. From this way nodes collect in routing tree. Finally this tree used for collecting data at base station [4, 10]. After query distributed in network each node start processing it and will sends it's produced tuples that match with query to its parent. And in this way parents do this operation and will send data to their parents. This operation is follows until response arrives to root (end user).

3. DATABASE OPERATION AND AFFECTION OF THEM IN SENSOR NETWORKS

Select, project and join operators combine in complex query. Select phrase determines attributes and aggregation of sensor records [14].

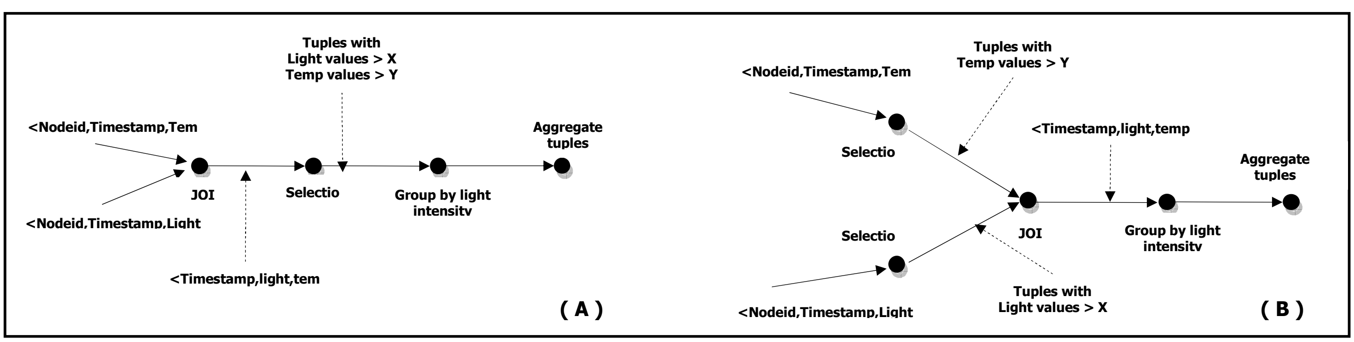
FIGURE 2. COMPLEX QUERY EXAMPLE

With this phrase can prevent from selecting of unimportant attributes and prevent assumption of them in network processing and communication. From phrase determine a virtual table that database operators should acts on them. In addition this phrase can use for join. When phrase filter sensor records through condition (s). Group by phrase grouping records into various groups according to several some attributes. Having phrase deletes some of unimportant groups according to defined conditions in it. Some of this operator cause that refined data arrive to end user. Affection several some of them for reducing number of communication and network processing is clearly. We discus affection of where, having and join in this section.

Where affection: for example we want compute temperature average. We can add "where Floor=3" condition to our query that prevent from computation of temperature average in other floors. Before each sensor node in routing tree sends its sensed data to its parent it controls its reading data with this condition in where. Only sensors that are in floor 3 sends their data to root and other nodes sends a notice bit (often zero) to their parents so parent node do not wait for receiving data from child. It is clear that sending a notice bit instead sending many bytes of real data has less cost [2].

Having affection: this phrase only used for necessary aggregate grouping in query and filter unnecessary nodes from final results. So this will use in base station after aggregate grouping for prevent forming unnecessary group centrally. For more study about having see [2].

Join operator: this is a very strong operator with full using. Because false using it consumes time, space and much energy, should be used with carefully [14]. A simple executing of it can produce all possible tuples. Noxiousness of this way for using in sensor networks is clear. There are many planes for how this operator is used. For example, result of R∞S and S∞R is similar but with different cost. Although result of (R∞S) ∞T and R∞ (S∞T) is similar but consumed energy in nodes, communication and response time are different on two forms of these operations. For more study about join see [4].



Releated WORK

The main idea of recent papers on sensor networks is focused on decreasing of node consumption energy and increasing of network life time. Researchers have done this idea in different ways. For example in [4] proposed a set of algorithms for executing effective join query that able to execute in bounded RAM. In [3] has been defined a cost model for queries that keep track of all received periodic sampling, communication and query routing cost. In [12] proposed a peer to peer query processing model via peer to peer tree structure that query can appear in each node (against other models that query appear only in base station). In [2] proposed LMGANC algorithm that permit grouping aggregation for decreasing energy consumption. In [5] proposed methods for collecting selectivity-aware data in sensor network that selectivity power is nominated as parameter for query optimization. In [8, 9] proposed optimization for multiple queries that in [9] this is done by greedy iterative join and a new received query is rewritten in set of combination queries.

4. Our proposed method for executing operators

Queries always are not simple and short. So using complex query in sensor networks is naturally. But when queries are complex additional consumption energy risk is increased. We explain this problem with example. User want to know temperature array in different regions that have same light intensity and their light intensity is higher than X and their temperature is higher than Y. We assume temperature table tuples format is <Nodeid, Timestamp, Temperature>and light table tuples format is <Nodeid, Timestamp, Light>. One way for executing this query is shown in Figure 2.A First all possible tuples from joining two tables are created and rows that have Light>X and Temperature>Y are selected, after they are grouped according to their Light ranges. Finally average temperature of each group gives to user. We propose a new method for executing queries and its affection on decreasing node consumption energy. So we define three rules:

1- Do select before all operators.
2- Do project after select before other operators.
3- After doing 1 and 2 rules you could do join operator.

   We apply above rules in our example; executing flow is shown in Figure 2.B.

Therefore sensor data that has Light>X and Temperature>Y have associate in join and aggregation apply on them. Suppose a state that a group of nodes are sense temperature and a group is sense light. Too we suppose a light sensor produce <254, 101, X0> (X0<X) data and its neighbor temperature sensor produce <256, 101, Y0> (Y0<Y) data (we suppose that sensor with 200 through 300 IDs are in same region place). In the first sequence of this query executing: first has been doing join then produced <101, X0, Y0> composite data. After checked conditions and determined this is not data that user wanted. Doing join operation caused overhead that caused node energy consumption and even best algorithm for doing join cause this energy consumption and consume memory node because produce additional data. But second sequence will guaranty that join operator only done on user wanted data. Therefore it will prevent from internetwork communication for joining unusable data and this will result preserving node energy and increasing network lifetime.

5. Aggregation QUERIES

Aggregation is one of the most usable queries in sensor networks. Example of aggregation queries is: "Temperature average on each 10s", "Maximum pressure in network total", "number of region that have Temperature>30". We could mean aggregation as summaries data of one attribute (columns) to a small digit value. Different methods such as Direct Delivery, Packet Merging and Partial Aggregation are proposed for executing aggregation functions [14, 6].

Recently papers have shown that executing aggregation queries in central form on base station is more expensive, because sending all messages to base station is more expensive. Autonomous ability and full computing ability of nodes has possible compositing and filtering of message locality. Therefore internetwork executing of aggregation functions with decreasing additional communication is common approach for preservation network energy. This method gives executing permission to each node (except leaf nodes) local aggregation before sending result to their parents. For example a way for executing Average query in sensor network in a form that each node sends Sum and Count of received data from childes and it's data to its parent. Also parents affect their received data in Sum and Count and send it direct up to root. Then root divide total Sums per Count and compute Average [1, 2, 6]. This approach is shown in Figure 3 it is clear that this way is cheaper from the way that all nodes send their data via multi hope protocol to base station and base station compute average of all received data. MIN and MAX aggregation functions also could execute as this way in internetwork. In this state each non leaf node compare received data from its children with self produced data and according to user request sends MIN or MAX of them to its parent, until root from this method computes MIN or MAX. In next section we propose an approach for executing MIN and MAX functions that has better behavior in special sensor network application from above procedure.

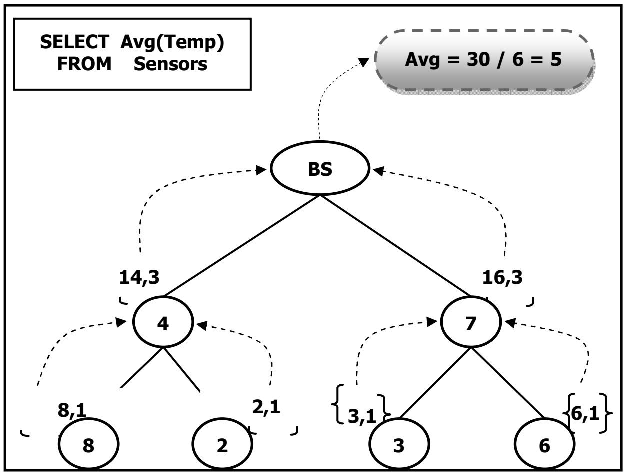


Figure 3. Simple aggregate query

6. Our proposed approach for executing MIN and MAX

There is some critical application that network formed for responding to predefined special conditions. For example suppose a sensor network is created for controlling and monitoring on wet rating in chemical material pool. When wet rating grows above a threshold, network atomically will execute necessary policy and dispatch backing group. Other instance is monitoring network of a radioactive pool to give necessary warning to user before leakages probability. And other example is place where sensors control pressure in a big boiler of stream generator. In all mentioned instances should be finding minimum or maximum from produced data node continually. We propose a procedure that according to it only effective data in minimum (or maximum) are flowed in network. In our procedure similar to MIN and MAX approach (that explained in pervious section), started at first (Figure 4.A). But after this time each node will store calculated MIN or MAX in previous epoch. On new epoch on each node after receiving data from child and node produce sensing data, this node compare above data with calculated local MIN or MAX in previous epoch, locality. If only one of the new data was less than past local MIN (or bigger than local past MAX), MIN (or MAX) selected and is sent to root. Else a notice bit (often zero) is sent to its parent and aware it from unaffecting of child (or children) data in network operation. In other method parent node wait a time slot and if do not received no news from child, means this as non effective of child data and produced sub tree from it. According to Figure 4 in first epoch sensors are produced their data that MAX process is shown in Figure 4.A on after epochs sensors are produced various data and are done MAX operation via our proposed approach with using previous epoch data (Figure 4.B). Root only received data that is better than existing MAX (or MIN). With this method some of produced data that no affection on defined target not sending in network and this cause more saving energy in nodes. In addition on these critical applications network life time is important factor that is improved with saving energy in nodes.

I. Simulation

We use SWANS [15] simulator and 811.b standards. We consider connectivity range of each node 50 meter and full duplex Chanel. We distribute sensors in uniform square region. We execute MAX query aggregation (that shown in Figure 4) and optimized order of operator (that shown in Figure 2.B) and other approach (Packet Merging, Partial Aggregation) in network. We increase size of network each time and compute node consumption energy average. Result are shown in Figure 5

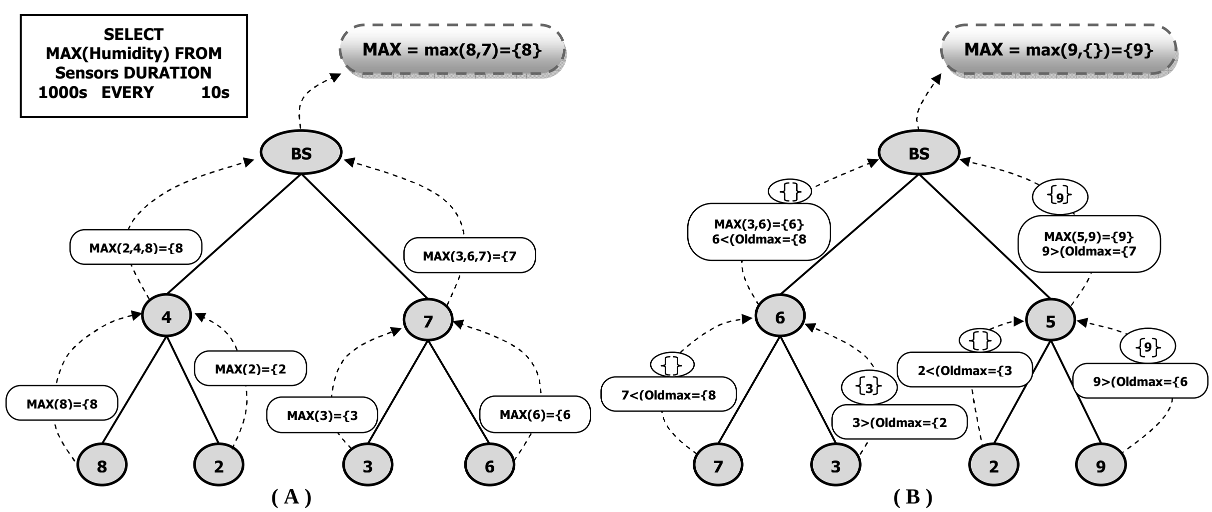


Figure 4. our algorithm on MIN query

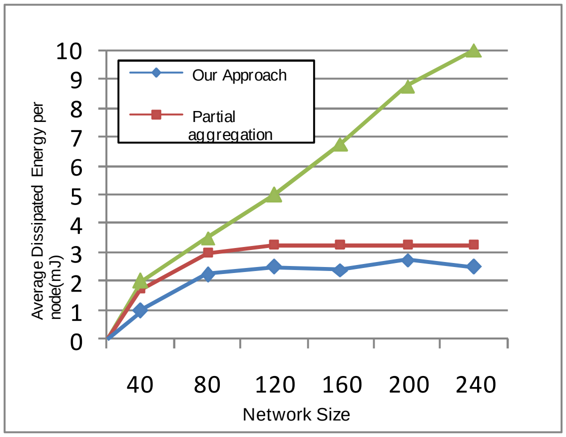


Figure 5. Simulation result of different methods

II. Conclusion

From Figure 5 appears that with increasing network size proposed approach minimize energy consumption of each node with minimizing set data in bus. Worst case happen when all nodes produce a data that greater than previous local maximum (or less than previous local minimum). Since node energy consumption in our approach will not greater than it in Partial Aggregation and worst case is similar to Partial Aggregation. Since because unimportant data are not sent, consumption energy for communicate data is decreased and network lifetime is increased.

REFERENCES

List and number all bibliographical references in 9-point Times, single-spaced, at the end of your paper. When referenced in the text, enclose the citation number in square brackets, for example [1]. Where appropriate, include the name(s) of editors of referenced books. The template will number citations consecutively within brackets [1]. The sentence punctuation follows the bracket [2]. Refer simply to the reference number, as in [3]—do not use "Ref. [3]" or "reference [3]" except at the beginning of a sentence: "Reference [3] was the first . . ."
Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do not put footnotes in the reference list. Use letters for table footnotes. Unless there are six authors or more give all authors' names; do not use "et al.". Papers that have not been published, even if they have been submitted for publication, should be cited as "unpublished" [4]. Papers that have been accepted for publication should be cited as "in press" [5]. Capitalize only the first word in a paper title, except for proper nouns and element symbols. For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation [6].

[1] R. Govindan, J. M. Hellerstein, W. Hong, S. Madden, M. Franklin, and Scott Shenker. "The sensor network as a database". Tech. Report 02-771, USC/CS, 2002.

[2] Z.Tu,W.Liang."Energy-Efficient Aggregate Query Evaluation in Sensor Networks". Australian National University,2006

[3] R.Rosemark ,W.C.Lee ,B.Urgaonkar, "Optimizing Energy-Efficient Query Processing in Wireless Sensor Networks".

[4] D.J.Abadi,S.Madden,W.Lindner," REED: Robust, Efficient Filtering and Event Detection in Sensor Networks". Proceedings of the 31st VLDB Conference, Trondheim, Norway, 2005

[5] Muhammad Umer," Selectivity-aware Query Processing in Sensor Networks", Sensys'07 Doctoral Colloquium, November 6, 2007, Sydney, Australia.

[6] S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong. "TinyDB: An acquisitional query processing system for sensor networks". *ACM TODS*, 30(1), November 2005.

[7] A. Demers, J. Gehrke, R. Rajaraman, N. Trigoni, and Y. Yao. "The cougar project: A work-in-progress report". *SIGMOD Record*, 32(4), 2003.

[8] N. Trigoni, Y. Yao, J. Gehrke, R. Rajaraman, and A. Demers." Multi-query optimization for sensor networks". In *Proc. of DCOSS*, 2005.

[9] S.Xiang ,H.B.Lim ,K.L.Tan," Impact of Multi query Optimization in Sensor Networks", Proceedings of the 3rd International Workshop on Data Management for Sensor Networks (DMSN'06), Seoul, South Korea, 2006.

[10] O.Gnawaliz, R.Govindanz, J.Heidemannx , "Implementing a Sensor Database System using a Generic Data Dissemination Mechanism",IEEE,2005.

[11] Yang Sun ," Query Processing for Sensor Network ", University ofWaterloo,2006.

[12] M.Demirbas, H.Ferhatosmanoglu," Peer-to-Peer Spatial QueriesinSensorNetworks".

1. [13] I.F.Akyildiz et al., "Wireless Sensor Networks: a survey", Computer Networks, Vol. 38, pp. 393-422, March 2002.

[14] . Yao and J. Gehrke." Query processing for sensor networks".In *Proc. ofCIDR*,2003.