Tuning Query Performance in Mobile Sensor Databases*

Vladimir Zadorozhny
University of Pittsburgh
Pittsburgh, PA
Phone: 1 412 624 9411
vladimir@sis.pitt.edu

Divyasheel Sharma
University of Pittsburgh
Pittsburgh, PA
Phone: 1 412 624 7378
dsharma@sis.pitt.edu

Prashant Krishnmurthy
University of Pittsburgh
Pittsburgh, PA
Phone: 1 412-624-5144
prashant@tele.pitt.edu

Alexandros Labrinidis
University of Pittsburgh
Pittsburgh, PA
Phone: 1 412 624-8843
labrinid@cs.pitt.edu

ABSTRACT

In this paper we propose a query-driven approach for tuning the time/energy trade-off in sensor networks with mobile sensors. The tuning factors include re-positioning of mobile sensors and changing their transmission ranges. We propose an algebraic query optimization framework that explores these factors while utilizing collision-free concurrent data transmissions with different degrees of data filtering and aggregation.

Categories and Subject Descriptors

C.2.1 [**Computer – Communication Networks**]: Network Architecture and Design – *Wireless communications;* H.2.4 [**Database Management**]: Systems – *Distributed databases*, *Query processing*

General Terms

Algorithms, Design, Performance, Experimentation

Keywords

Mobile Sensors, Sensor Networks, Query Optimization, Data Transmission Algebra, Collision Domains

1. INTRODUCTION

We adopt a broad definition of a sensor database to be a wireless network composed of a large number of sensor nodes most of which are power-constrained [YHE02]. These sensor nodes can be attached to PDAs or other mobile devices such as mobile robots. In this way, teams of humans and/or mobile robots in conjunction with stationary sensor nodes can be deployed to acquire and process data for surveillance and tracking, environmental monitoring for highly sensitive areas, or execute search and rescue operations.

A *sensor query* is characterized by large data streams among participating nodes with possible in-node data filtering/aggregation, and can be described as a tree-like data delivery pattern (*query routing tree*). Minimizing sensor query response time becomes crucial in mission-critical sensor networks. At the same time, minimizing energy consumption per query is equally crucial for battery-powered devices. In general, the time/energy trade-offs involve energy and time gains/losses associated with specific layouts of the nodes.

Several techniques have been proposed to alleviate the problem of limited power at the network level such as energy-efficient routing, clustering and transmission scheduling [HIS01, YYA02, HCB00, CFS03]. Sensor database research has also looked into sensor query processing strategies to minimize the query response time and reduce energy consumption by sampling [MFHH03], prediction [GI01], approximation [CLKB04], and in-network query processing (or aggregation) [BGS01, MFHH02, SBLC03]. With the same goal in mind, our research makes an effort to fuse techniques and methods currently used in the two different areas of databases and networking. In this paper we propose a query-driven approach for properly tuning the time/energy trade-off in sensor networks. Our approach is based on two key observations that have a considerable impact on the time/energy trade-off:

- Positioning (relocation) of mobile sensors.
- Changing the transmission ranges of sensors

Specifically, both these factors impact on the following characteristics of a query tree:

● **Collision-free concurrency:** Packet collision is a major source of energy and time waste. If two or more nodes in the same *Collision Domain (CD)* transmit packets at the same time, a collision occurs, and the corrupted packets must be discarded. Both node relocation and changing the transmission range could result in changing the number of potential collision-free concurrent transmissions.

● **Filtering factor:** Both node relocation and change of the transmission range can result in changing the number of hops and the intermediate transmission nodes involved in query execution. This, however, brings both benefits and penalties. If the filtering factor of the intermediate node is low (i.e., it just retransmits the data) then by introducing it, we expect to have some time and energy loss due to the extra hop. On the other hand, the intermediate node does reduce the transmission range, which results in the saving energy. If the intermediate node does a lot of filtering, the benefit includes spending less energy by transmitting less data.

In this paper we introduce a novel query routing tree optimization technique that minimizes query response time and energy consumption. Our approach is founded on algebraic analysis of alternative query routing trees. In related research [ZCR04, ZCL04] we proposed a Data Transmission Algebra (DTA) that uniformly captures the structure of data transmissions, their

* This work was partially supported by a 2002 startup fund from the University of Pittsburgh, School of Arts & Sciences.

MDM 2005, May 9-13, 2005 , Ayia Napa Cyprus.

constraints, and their requirements. The DTA framework enables both qualitative analysis and quantitative cost-based optimization of sensor queries. In this paper we extend the basic DTA with a translation operation that formalizes the relocation cost of mobile nodes and the costs of changing transmission ranges of the sensors. The extended DTA optimizes the time/energy trade-off utilizing the concept of *Pareto optimality* and utility-based selection of the optimal query tree.

2. SYSTEM MODEL

We assume that a query optimizer executes at a base station along with other utilities such as data mining for cost-effective and model-driven data acquisition [DGMHH04]. For a given query or data acquisition model, our query optimizer selects the query routing tree with optimal response time and energy consumption.

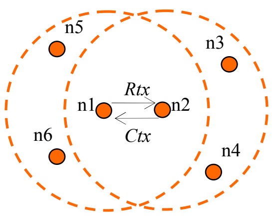

Figure 1: Collision domain of two communicating sensors

Mobile sensors are moved into target positions according to the selected routing query tree. A query optimizer generates alternative query routing trees and mobile sensor deployment plans taking into consideration the current topology of stationary sensor nodes, the applications' coverage requirements, and the collision domains of the sensor nodes.

Figure 1 elaborates the concept of collision domains in a typical wireless network (e.g. IEEE802.11) and illustrates how collisions are handled in such networks. In order for a sensor node *n1* to communicate with sensor node *n2*, *n1* needs to first send a request for transmission packet (*Rtx*) to *n2*, so that all other nodes in its transmission range (*n5* and *n6* in Figure 1) become aware of the communication and remain silent until *n1* ends the transmission. Sensor *n2* replies to *n1* with a confirmation packet (*Ctx*), so that the nodes in its transmission range (*n3* and *n4* in Figure 1) also become aware of the communication and avoid any transmission until the end of the current transmission. In this case, nodes *n3*, *n4*, *n5*, and *n6* belong to the same CD. In general, any two communicating nodes *ni* and *nj* specify a collision domain CD(*ni,nj*) defined as the union of the transmission/reception ranges of *ni* and *nj*.

Mobile sensor nodes can move in order to improve time and energy query performance. The mobile sensors should position themselves and adjust their transmission power so as to minimize overlap of CDs in the query tree. In some cases, however, this general strategy may result in time and/or energy loss. The optimizer is responsible for the choice of the best query strategy.

3. ALGEBRAIC QUERY OPTIMIZATION

In related research [ZCR04, ZCL04] we introduced an algebraic query optimization technique for static sensor networks. We developed a *Data Transmission Algebra* (DTA) for a query optimizer to generate query routing trees to maximize collision-free concurrent data transmissions. Here we extend the DTA to handle mobility of sensor nodes.

3.1 Basic DTA

The DTA consists of a set of operations that take transmissions between wireless sensor nodes as input and produce a schedule of transmissions as their result. We call an *elementary transmission* (denoted *ni~nj*) a one-hop transmission from sensor node *ni* to node *nj*. Each transmission *ni~nj* is associated with a collision domain CD(*ni*, *nj*) as defined above. A transmission schedule is either an elementary transmission or a composition of elementary trans-missions using one of the operations of the DTA. The basic DTA includes three operations that combine two transmission schedules A and B:

- ***o*(A,B)**. This is a strict order operation, that is, A must be executed before B.
- ***a*(A,B)**. This is an overlap operation, that is, A and B can be executed concurrently.
- ***c*(A,B)**. This is a non-strict order operation, that is, either A executes before B, or vice versa.

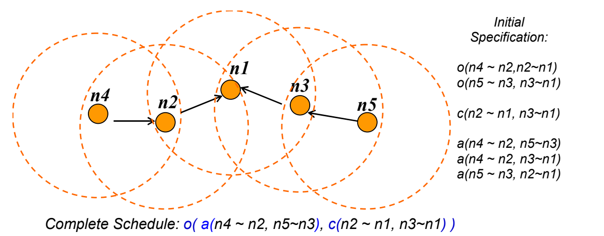

Figure 2: Example of DTA specifications

For example, consider the query tree in Figure 2, which was generated for some query Q. Figure 2 shows the *initial DTA specification* reflecting basic constraints of the query tree. For instance, operation *o(n4~n2, n2~n1)* specifies that transmission *n2~n1* occurs after *n4~n2* is completed because of the query tree topology. Operation *c(n2~n1, n3~n1)* specifies that there is an order between transmissions *n2~n1* and *n3~n1* since they share the same destination. However this order is not strict. Operation *a(n4~n2, n5~n3)* specifies that *n4~n2* can be executed concurrently with *n5~n3*, since neither *n3* nor *n5* belongs to CD(*n4,n2*), and neither *n4* nor *n2* are in CD(*n5,n3*)

The DTA introduces a set of transformation rules [ZCR04, ZCL04] that can be used to generate complex transmission schedules. Figure 2 shows an example of a complete schedule that includes all elementary transmissions of the query tree.

3.2 Mobility-enhanced DTA

In this paper we will focus on mobile sensors that can facilitate data delivery acting as intermediate nodes rather than data acquisition nodes. Such mobile facilitators can introduce extra hops in order to reduce transmission ranges of the static sensors. In addition, the facilitators can also act as *filters*, decreasing the amount of data transmitted from the static sensors to the root node. Figure 3 shows a tree topology with four fixed sensors *s1, s2, s3, s4* and three different positions of a mobile facilitator *Ms1*. The facilitators consume extra energy and introduce some extra processing delay. However, by reducing the transmission range and data stream sizes, they are also capable of reducing the overall query time and energy consumption.

Given a query, the coverage requirements, and the initial position of both stationary and mobile sensors, the query optimizer shifts through possible mobile sensor positions in order to generate the candidate trees with acceptable response time and energy consumption. In order to support such mobility-based query optimization we extended our DTA with a translation operation ***tr*** that transforms initial schedules associated with given tree topology. The ***tr*** operation takes two input parameters: target initial specification *tis* and a set of mobile facilitators *ms* whose repositioning should transform the initial specification of the current query tree into *tis*. The output of ***tr*** is a complete schedule *Sch* generated from the *tis*: $\boldsymbol{tr}_{tis,ms}$***(****Sch****)****.*

It follows from the definition of the ***tr*** operation that it is not deterministic, since we may have more than one schedule satisfying constraints of one initial specification. For example, consider the query tree topology in Figure 3a with initial specification *is1* and one facilitator *m*.

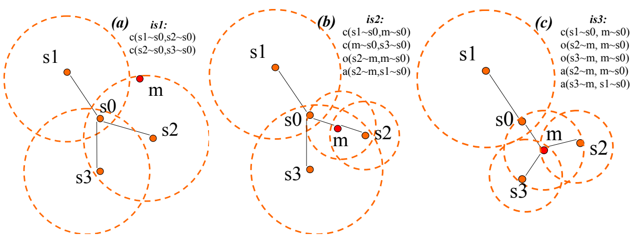

Figure 3: Impact of mobility on DTA specification

Figures 3b and 3c illustrate two different re-positioning of *m* reflected in the updated initial specifications *is2* and *is3*. Then the following specifications are valid DTA expressions:

- $\boldsymbol{tr}_{is2,m}$ ***(****o(a(s2~m, c(s1~s0,s3~s0)), m~s0)****)***
- $\boldsymbol{tr}_{is2,m}$ ***(****o(a(s2~m,s1~s0), c(m~s0,s3~s0))****)***
- $\boldsymbol{tr}_{is3,m}$ ***(****o(c(a(s3~m,s1~s0),s2~m),m~s0))****)***
- $\boldsymbol{tr}_{is3,m}$ ***(****o(c(s3~m,s2~m),c(m~s0,s1~s0))****)*.**

In order to identify which out of the many transmission schedules would be the best one, we propose using a cost-based tree generation framework, which we describe in the next section.

4. COST-BASED QUERY TREE GENERATION

4.1 Pareto-optimal Query Trees

Generating a query tree with acceptable query response time and overall energy consumption is a multi-objective optimization (MOP) problem [Miett99]. In general, MOP aims at minimizing values of several objective functions *f1,…fn* under a given set of constraints. In most cases it is unlikely that different objectives would be optimized by the same parameter choice. To choose between different vectors of the optimization objectives the optimizer utilizes the concept of *Pareto optimality* [Miett99]. Informally, an objective vector is said to be Pareto optimal if all other feasible vectors in the objective space have a higher value for at least one of the objective functions, or else have the same value for all objectives. Typically, there is more then one Pareto optimal vector (Pareto points) reflecting the trade-offs between different objectives. For example, if the following set includes feasible solutions for bi-objective MOP: {(5,1), (2,2), (2,3), (1,5)}, then the *Pareto optimal set* (also called *Pareto front*) is {(5,1), (2,2), (1,5)}. Among all Pareto optimal solutions the optimizer should chose one using an application-dependent *utility function*.

4.2 Time/Energy Utility Function

Consider two Pareto optimal objective vectors (T1,E1) and (T2,E2), where T1, T2 are response times and E1,E2 – consumed energy. In order to choose one of them the optimizer should trade time for energy. Informally, the optimizer should evaluate time and energy gains/losses and make a preference considering the relative importance of time and energy in the context of a specific query. The optimizer considers two factors: time factor TF and energy factor EF ranging from 0 to 1. The following algorithm describes the computation of our time/energy utility function:

```
UF ((T1,E1),(T2,E2),TF,EF)
  BEGIN
    DT=(T1-T2)/(T1+T2);  DE=(E1-E2)/(E1+E2);
    DT1 = abs(DT * TF); DE1 = abs(DE * EF);
    if (DT < 0 and DT1>DE1)
    then return  (T2,E2),
    else   if (DE < 0 and DE1>DT1),
           then return (T2,E2),
           else return (T1,E1).
  END
```

Consider the *Pareto set* from a previous subsection: {(5,1), (2,2), (1,5)}.Then ***UF(***(5,1),(2,2),0.8,0.2***)*** will return (5,1), while ***UF(***(5,1),(2,2),0.2,0.8***)*** results in (2,2). In general UF impose an order on the Pareto set for a given setting of TF and EF. For example, with TF=0.2 and EF=0.8 the order would be {(2,2),(1,5),(5,1)}.

4.3 Query Optimizer

It is well-known that the problem of generating Pareto sets is NP. In practice MOP should be combined with scalable optimization techniques. Our optimizer utilizes randomized algorithms [IK90] to generate Pareto fronts for large query trees. Randomized algorithms will search for a Pareto optimal solution by performing random walks in the solution space via a series of valid moves [ZCK04]. Our optimizer uses a cost model to estimate both response time and energy consumption of a DTA schedule. Details of the time cost model are in [ZCK05]. Below we elaborate on the energy cost model.

The energy consumed by a node in sensor network is dependent upon the amount of data sent, received, processed, and discarded. Consider an elementary transmission *ni~nj*. Node *ni* transmits *x* bytes to node *nj* and the node *j* receives the *x* bytes correctly. No packets are discarded since none of the nodes *ni* and *nj* is idle. The energy consumed in this elementary transmission (E) can be represented as follows:

$$\mathbf{E_{total} = E_{Tx}+E_{Rx}} \qquad (1)$$

where E_{Tx} represents the energy consumed for transmitting a packet of *x* bytes by *ni* and E_{Rx} is the energy spent for receiving the packet of *x* bytes by *nj*. To determine E_{Tx} and E_{Rx}, we use the energy model from [FN01] with some modifications. In [FN01], it is assumed that the energy spent for transmission or reception of *x* bytes is given by:

$$\mathbf{E = C + } \boldsymbol{mx} \qquad (2).$$

The constant C and the slope *m* are 454 μJ and 1.9 μJ/byte for transmission and 356 μJ and 0.5 μJ/byte for reception. While this is unspecified in [FN01], we assume that the C and *m* values for transmission in (2) satisfy the requirements for a receiving node to *just correctly receive the data* when the distance between the transmitting node and receiving node is 300m.

The power loss with distance is typically characterized by a path-loss equation [Pah02] of the form:

$$\mathbf{P_r(\mathit{d})\ (dBm) = P_t\ (dBm) - L_0 - 10\alpha\ log_{10}(\mathit{d})} \qquad (3)$$

Here, P_t is the transmit power, P_r is the received power at distance *d* from the transmitter, α is the path-loss exponent (α = 2 in free space and typically a value of α = 4 is assumed for other areas), and L_0 is a constant that depends on the frequency and other gains and losses. The received power from (3) at a distance of 300m given P_t = 34.17 dBm and α = 4 will be P_r = -64.915 – L_0 which we assume is the smallest value of the received power necessary for correctly receiving a packet.

Suppose the distance between the transmitter and receiver is *d* = 30m. The received power at the sensor node *for just correctly receiving data* must be –64.915 – L_0 which is also equal to $P_t - L_0 - 40\log_{10}(30)$. From this, we see that the transmit power needs to be only equal to –5.83 dBm = 0.26 W. The corresponding energy consumed for transmission will be 0.26 W × 0.727s = 0.19 J. In general, the transmit power required at a distance *d* from the transmitter is:

$$\mathbf{P_{t,\,new}\ (dBm) = P_{t,300} + 40\ log_{10}\ (\mathit{d}/300)} \qquad (4)$$

In our model, the energy consumed by the receiver does not change with distance as it has to perform the same amount of processing of the data.

Our optimizer uses equations 1, 2, 3 and 4 to calculate the energy cost (ECost) for elementary transmissions. The energy cost for the strict order, overlapping schedules and choice operation over strict order are sums of ECost for each elementary transmission in there respective schedules. Energy cost of the translation operator ***tr****tis,ms****(Sch)*** consists of the energy to move the facilitator sensor to required position plus energy cost of the schedule *Sch***.**

5. EXPERIMENTS AND ANALYSIS

For our experiments we have considered several star sensor query tree topologies including both fixed sensors and mobile facilitators. The optimizer evaluated best query plan for different positioning of the facilitator. In this paper we report experimental results for a two-hop scenario with one facilitator.

The distance between leaf static sensor and the root was selected to be 300 meters. The amount of transmitted data was uniformly distributed in the range 1MB-4MB per leaf sensor. For simplicity we assumed that the mobile facilitator could only be positioned in pre-defined "placeholders". In addition, we considered facilitator filtering factors ranging from 0.2 (facilitator retransmits only 20% of the input data stream) to 1.0 (facilitator retransmits all input data stream).

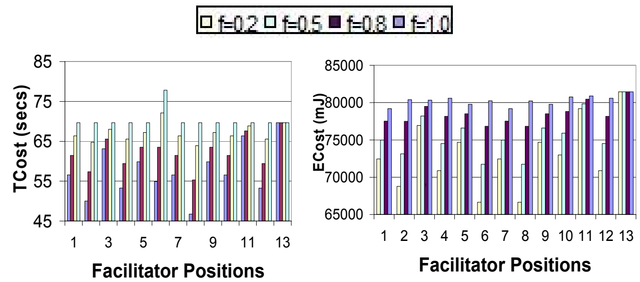

Figure 4: Energy and Time vs Facilitator Position

Figure 4 reports time and energy distributions for suggested query plans chosen by optimizer for 12 facilitator positions and four facilitator filtering factors. Position 13 represents a scenario where no facilitator has been used. In general we observe stable improvement in both time and energy cost with decrease of the filtering factor. This is consistent with the fact that the amount of the transmitted data is the most critical factor in sensor query optimization. However, different facilitator positions are characterized by considerable variance in time and energy cost for all filtering factors. Almost all scenarios that utilize facilitator outperform the no-facilitator scenario in both time and energy consumption.

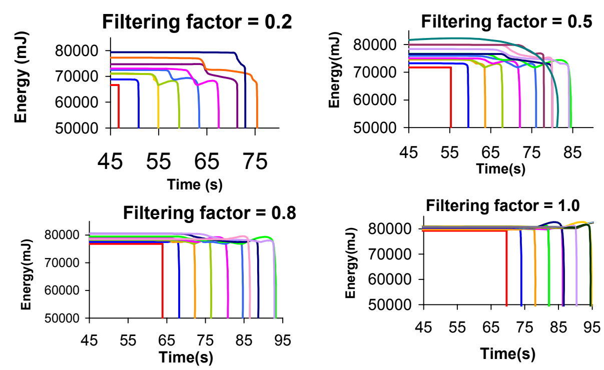

Figure 5: Pareto fronts vs facilitator filtering factor

This is also expected behavior since even with the filtering factor of 1.0 the facilitator still helps by reducing transmission ranges and introducing additional opportunities for collision-free concurrent data transmissions.

Figure 5 reports on the Pareto fronts explored by the optimizer for each of the facilitator filtering factors. A major observation here is stable increase of variance in both time and energy consumption with decrease of the facilitator filtering factor. For the filtering factor of 0.2 the energy varies between 66000 mJ and 80000mJ, while for the filtering factor of 1.0 the energy range is 78000-81000 mJ. The time ranges are 46-76 sec and 70-95sec correspondingly. This means that in general the optimizer can benefit from higher filtering factors (the lower filtering factor reduces more input data). However, there is a considerable risk for the optimizer to behave as badly as in the case of high filtering factor. This, in particular, motivates our current research in fine-tuning the optimization strategies. Part of it is a proper design of a utility function to choose among multiple Pareto optimal solutions.

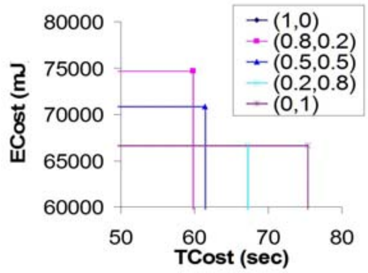

Figure 6: Effect of TF and EF on utility-based tree selection

Figure 6 shows optimization choices for one of generated Pareto fronts using the utility function described in Section 4. The choices are made for different time and energy factors. We observe a consistent optimizer behavior in making preferences with respect to the time and energy factors. We observed similar performance trends with other query tree topologies considered in our experiments.

6. CONCLUSION

This paper takes a novel view of sensor networks by considering them to be comprised by both stationary and mobile sensor nodes. In this mobile ad-hoc sensor network, we proposed an innovative approach to query performance tuning. Our approach utilizes an algebraic framework that formalizes basic query and network constraints. We developed a scalable randomized query optimizer that efficiently explores Pareto fronts in order to minimize both query response time and energy consumption and, through extensive experiments, illustrated the significant benefits it can provide over traditional approaches.

Acknowledgments: The authors would like to thank the anonymous reviewers, Panos K. Chrysanthis, and the other members for the Advanced Data Management Technologies Laboratory at the University of Pittsburgh for their invaluable feedback.

7. REFERENCES

[BGS01] P. Bonnet, J. Gehrke, and P. Seshadri. Towards Sensor Database Systems. *Proc. of MDM Conference*, 2001.

[BS00] P. Bonnet and P. Seshadri. Device Database Systems. *Proc. of ICDE Conference*, 2000.

[BSLC03] J. Beaver, M. A. Sharaf, A. Labrinidis, and P. K. Chrysanthis. Location-Aware Routing for Data Aggregation in Sensor Networks. *Proc. of the 2nd Hellenic Data Management Symposium*, 2003.

[CFS03] U. Cetintemel, A. Flinders, and Y. Sun. Power-Efficient Data Dissemination in Wireless Sensor Networks. *Proc. of ACM MobiDE Workshop*, 2003.

[CJBM01] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris. SPAN: An Energy-Efficient Coordination Algorithm for Topology Maintenance in Ad Hoc Wireless Networks. *Proc. of Mobile Computing and Networking Conference,* 2001

[CLKB04] J. Considine, F. Li, G. Kollios and J. Byers. Approximate Aggregation Techniques for Sensor Databases. *Proc. of IEEE ICDE Conf.*, 2004.

[DGMHH04] A. Despande, C. Guestrin, S. Madden, J Hellerstein, W. Hong. Model-driven Data Acquisition in Sensor Networks. *Proc. of VLDB Conf.*, 2004.

[FN01] Feeney, M. Nilsson. Investigating the energy consumption of a wireless network interface in an ad hoc networking environment. *Proc. of INFOCOM.* 2001.

[GI01] S. Goel and T. Imielinski. Prediction-based monitoring in sensor networks: Taking lessons from MPEG. Computer Comm. Review, 31(5), 2001.

[IG99] T. Imielinski and S. Goel. DataSpace - Querying and Monitoring Deeply Networked Collections in Physical Space. *Proc. of ACM MobiDE Workshop*, 1999.

[IK90] Y. E. Ioannidis, Y. Kang. Randomized algorithms for optimizing large join queries. ACM SIGMOD, 1990.

[HIS01] J. Heidemann, F. Silva, C. Intanagonwiwat, R. Govindan, D. Estrin and D. Ganesan. Building efficient wireless sensor networks with low-level naming. *Proc. of SOSP*, 2001.

[HCB00] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wirelessmicrosensor networks. *Proc. of HICSS Conf.*, 2000

[Miett99] K. Miettinen. Nonlinear Multiobjective Optimization Kluwer Academic Publisher, 1999.

[MFHH02] S. Madden, M.J. Franklin, J.M. Hellerstein, and W. Hong. Tag: A tiny aggregation service for ad hoc sensor networks. *Proc. of OSDI*, 2002.

[MFHH03] S. Madden, M.J. Franklin, J.M. Hellerstein, and W. Hong. Tag: The Design of an Acquisitional Query Processor for Sensor Networks. SIGMOD 2003.

[Pah95] K. Pahlavan and A. Levesque, Wireless Information Networks, John Wiley and Sons, 1995.

[Pah02] K. Pahlavan and P. Krishnamurthy, Principles of Wireless Networks: A Unified Approach, Prentice Hall, 2002.

[SBLC03] M. A. Sharaf, J. Beaver, A. Labrinidis, and P. K. Chrysanthis. TiNA: A Scheme for Temporal Coherency-Aware in-Network Aggregation. *Proc. of MobiDE Workshop*, 2003.

[YG02] Y.Yao and J.E. Gehrke. The Cougar Approach to In-network Query Processing in Sensor Networks. *SIGMOD Record*, 31(3), 2002.

[YHE02] W. Ye, J. Heidemann, and D. Estrin, "SMAC: An Energy-Efficient MAC Protocol for Wireless Sensor Networks. *Proc. of IEEE INFOCOM Conference*, 2002.

[YYA02] M. Younis, M. Youssef and K. Arisha. Energy-aware routing in cluster-based sensor networks. *Proc. MASCOTS*, 2002

[ZCK04] V. Zadorozhny, P. K. Chrysanthis, P.Krishnamurthy A Framework for Extending the Synergy between MAC Layer and Query Optimization in Sensor Networks. *Proc. of DMSN VLDB Workshop* 2004.

[ZCL04] V. Zadorozhny, P. K. Chrysanthis, A. Labrinidis. Algebraic Optimization of Data Delivery Patterns in Mobile Sensor Networks.*Proc. of MDDS DEXA Workshop* 2004.

[ZBVRU02] V. Zadorozhny, L. Bright, M.E. Vidal, L. Raschid, and T. Urhan. Efficient Evaluation of Queries in a Mediator for WebSources. *Proc. of ACM SIGMOD* 2002.

[ZC03] V. Zadorozhny, P. Chrysanthis. Location-Based Computing. In Telegeoinformatics: Location-Based Computing and Services, Taylor and Francis Books., 2003.