Mining and Monitoring Sensor Networks

Johannes Gehrke
Department of Computer Science
Cornell University
johannes@cs.cornell.edu

Abstract

The widespread distribution and availability of small-scale sensors, actuators, and embedded processors is transforming the physical world into a computing platform. One such example is a sensor network consisting of a large number of sensor nodes that combine physical sensing capabilities such as temperature, light, or seismic sensors with networking and computation capabilities. Applications range from environmental control, warehouse inventory, health care to military environments. Sensor networks will produce a flood of spatio-temporal data of unprecedented scale. In this paper, we outline some of the associated data mining research challenges and some initial results.

1 Sensor Networks

Sensor nodes are one example of an emerging computing environment. Moore's law tells us that we will soon see sensors that measure 1 cm^3 running Linux [uCl01, Sys01] or QNX [Hil01], and there is a plethora of research to scale down components to the 1 mm^3 range (about the size of a large piece of dust) [KKPG98, KKP99]. Instead of deploying preprogrammed sensor networks only for specific applications, future networks will have sensor nodes with different physical sensors for a wide variety of application scenarios and different user groups.

Sensor networks have the following physical resource constraints:

- **Communication.** The bandwidth of wireless links connecting sensor nodes is usually limited, on the order of a few hundred Kbps, and the the wireless networks connecting sensors provide only limited quality of service.
- **Power consumption.** Wireless sensors have limited supply of energy, and thus energy conservation is a major system design consideration. Current sensor nodes already have three different sleep modes with several orders of magnitude different power usages [HSW$^+$00], and future nodes will have sophisticated power management features.
- **Computation.** Sensor nodes have limited computing power and memory sizes that restrict the types of data processing algorithms that can run on the sensors and the sizes of intermediate results that can be stored on the sensor nodes.

We advocate a database approach to sensor networks. Declarative queries are especially suited for sensor network interaction: Users and applications programs issue high-level queries without knowing where and how the data is generated in the sensor network and how the data is processed to compute the query answer. Sophisticated catalog management, query optimization, query processing, and analysis techniques will isolate the user from the physical details of contacting the relevant sensors, in-network processing, composition of high-level knowledge, and sending the results back to the user. Due to the large volumes of data produced in sensor networks and the above described physical constraints of sensor networks, data transmission of all data back to a central node for offline storage, querying, and data analysis is infeasible for sensor networks of non-trivial size [EGHK99, PK00]. Thus information extraction from sensor networks has to perform non-trivial in-network data processing, data reduction, and aggregation.

Given the view of the sensor network as a huge distributed database system where each sensor node corresponds to a database site that holds part of the data, we would like to adapt existing techniques from distributed and heterogeneous database systems for the sensor network environment. But there are two major differences between sensor networks and traditional distributed and heterogeneous database systems.

First, sensor networks have physical characteristics that are very different from regular desktop computers or dedicated equipment in data centers. Individual sensors might fail at any time, the networking layer only provides very weak quality of service (such as a best-effort delivery of network packets [IGE00]), and the sensor nodes have strict resource limitations such as limited memory, computational power, and energy. Query processing has to be aware of these physical constraints. As an analogy, consider access to secondary storage in traditional database systems. Database systems bypass the operating system buffer to have direct control over secondary storage resource [Sto81]. In a sensor network database system the query processing layer will be tightly integrated with the networking layer in order to manage the resource "network" intelligently.

Second, sensors produce data continuously in data streams, and sensor nodes have only limited memory and computational resources. We need to develop new data analysis techniques for the online processing of data streams that do not assume that all records are materialized at some level of the storage hierarchy.

While developing techniques that address the two problems above, we must not forget that scalability of our techniques with the size of the network, the data volume, and the query workload is an intrinsic consideration to any design decision. In the next section, we describe initial results on techniques for data stream aggregation directly at individual sensor nodes with limited amount of memory.

2 One Example Technique: Computing Correlated Aggregates

Correlated aggregates provide a natural mechanism for the flexible composition of standard aggregates that are useful for sensor network applications. For example, $\texttt{COUNT}\{x : x > 0.5 * \texttt{MAX}(x)\}$ operates on a multiset of x values, and computes the number of x values that are within 50% of the maximum x value in the multiset. Similarly, $\texttt{MAX}\{y : x < \texttt{AVG}(x)\}$ operates on a multiset of (x, y) tuples, and computes the maximum y value obtained from tuples where the x value is less than the average x value in the set; this value may not be the maximum y value in the entire multiset. Prior work considered only the exact computation of correlated aggregates over finite data sets [CR96, ACJK01, Cha99]. For data streams of large volume, such exact computation is not feasible, and providing a quick approximate answer will have to suffice.

Data Stream Models. Let us first introduce a data stream model for the computation correlated aggregates with limited amount of main memory. Consider a relational schema R with attributes $X_1, \ldots, X_k$ where attribute X_i has $\texttt{dom}(X_i)$. We call $\texttt{att}(R) \stackrel{\text{def}}{=} \texttt{dom}(R) \stackrel{\text{def}}{=} \texttt{dom}(X_1) \times \cdots \times \texttt{dom}(X_k)$ the *attribute space of* R. Let R be a relational schema with attribute space $\texttt{att}(R)$. We call a function $O : \mathbf{N} \rightarrow \texttt{att}(R)$ an *ordering of* R. A *sequence* is a tuple $S(R, O)$ where R is a relational schema and O is an ordering of R. Given a sequence $S(R_S, O_S)$, we also refer to the natural numbers as *positions*, and we use $S[i]$ for $O_S(i)$, the record at the ith position.

Our model of computation is similar to the model introduced by Henzinger et al. [HRR98]. It contains a single input sequence S_{in} and a single output sequence S_{out}, and the model has as single parameter m, the amount of space available. Computation in our model proceeds in steps, and each computation step consists of three substeps. Consider the ith computation step. In the first substep, we read $S_{in}[i]$ from the input sequence S_{in} into a memory location; in the second substep, we perform an unlimited amount of computation in memory; and in the third substep we write into the ith position of the output sequence $S_{out}[i]$. We call an algorithm for our model of computation a *stream algorithm*. Thus, our algorithms map input streams into output streams, which could be used for further processing.

A *stream aggregate* has three components: A *scalar aggregate function* $AGG : 2^{\mathbf{R}} \to \mathbf{R}$, a *scope function* $scope : \mathbf{N} \to 2^{\mathbf{N}}$, and a *selection predicate* P. Given an input sequence S_{in}, a stream aggregate operator $Agg(AGG, scope, P)$ returns the sequence S_{out} such that

$$S_{out}[i] \quad = \quad AGG\{S_{in}[j].X \mid j \in scope(i) \wedge P(S_{in}[j], S_{in}[scope(i)])\}$$

where $S_{in}[scope(i)] \stackrel{\text{def}}{=} \{S_{in}[j] \mid j \in scope(i)\}$ and X is an attribute of R.

Three particular types of scope functions are especially interesting. We call the scope function $fScope : \mathbf{N} \to 2^{\mathbf{N}}$ such that $fScope(i) = \{1, \ldots, i\}$ for $i \in \mathbf{N}$ a *full window scope*, and we call the scope function $swScope_w$ such that $swScope_w(i) = \{\max(1, i-w+1), \max(1, i-w+2), \ldots, i\}$ a *sliding window scope of size* w. A full window scope is just a special case of a *landmark window scope*. A landmark window scope takes as input a *landmark set* $S = \{s_1, s_2, \ldots\}$. Given such a set S and a position i, $lmScope(S, i) = \{s_j, s_j + 1, \ldots, i-1, i\}$, where s_j is the largest position in S such that $s_j \leq i$.

Consider the stream aggregate operator $Agg(AGG, scope, P)$. If the selection predicate P does not contain any aggregate function, then we call Agg a *level* 0 stream aggregate operator. Recursively, let Agg be a level i stream aggregate operator. Then $Agg'(AGG', scope', P')$ is a level $i+1$ stream aggregate operator if

$$S_{out}[i] \quad = \quad AGG'\{S_{in}[j].X_l \mid j \in scope'(i) \wedge P'(S_{in}[j], Agg(S_{in})[i])\}$$

Our notion of the level of a stream aggregate lets us relate stream aggregates to regular queries over a static relation. A level i stream aggregate can be evaluated over a static relation in at most $i+1$ scans. Note that our notation of a level i stream aggregate is purely syntactic. There are level i stream aggregates that have equivalent level 0 aggregates for any i. Establishing such equivalences is an interesting topic for future research.

If the amount of available space m is infinite, then we can compute S_{out} exactly for any stream aggregate through the following simple algorithm: At step i, we read $S_{in}[i]$ into memory location $M[i]$. We then compute the exact value of the stream aggregate $S_{out}[i]$ using the copy of the input stream being stored and store the output in $S_{out}[i]$.

Open Research Problems. In a recent paper, we described algorithms and associated performance experiments that show how to maintain correlated aggregates for the landmark and sliding window case [GKS01]. We believe that the algorithms in that paper are only a first step towards a more general investigation into computation of summary statistics and correlated aggregates over data streams. For example, all our algorithms are only empirically evaluated; no formal analysis of their properties has been developed yet.

A multitude of other open problems arises from this initial work. How do we compute one-dimensional and multi-dimensional histograms over data streams with limited amount of main memory, both in the landmark window and the sliding window model? Can we give formal a-priori performance guarantees for our algorithms? In recent work, we introduced a new quantitative measure called the deviation for data mining models whose structure exhibits the meet-semi-lattice property [GGR99]. Can we use the deviation as a starting point to quantify differences between summary data structures constructed off- and online? Another open question is how to combine such summary data structures in further processing. Data reduction at individual sensor nodes through the computation of aggregates or more sophisticated data mining models is only a first step. We need to be able to process these synopsis data structure themselves when we combine synopsis data from several sensors into higher-level knowledge. Conceptually, we can think about the data structures described in this section as new base relations; future research has to address how to design operators that work directly on these and similar summary data structures.

3 Related Work

There is a plethora of research that is relevant to the topics outlined above. Due to space restrictions, we only shortly survey related work on data stream algorithms, distributed query processing, and networking techniques for sensor networks.

Data Stream Computation. Data structures that hold such summary information are examples of synopsis data structures [BDF$^+$97, GM98, AGPR99a, AGPR99b, AGMS99, AGP99]. Construction of summary data structures over data streams has been of much interest recently [HRR98]. Algorithms and systems have been proposed for the computation of approximate frequency moments [AMS99], L^1 and L^p distance functions [FKSV99, FS00], property testing [FKSV00], and signatures [CFPR00]. There has also been recent work on mining data streams, such as the construction of decision trees over data streams [GGRL99, DH00], clustering data streams [GMMO00], incremental maintenance of association rules [CHNW96, CNT96, TBAR97], and incremental maintenance under block arrivals of records [GGR00]. Recent work by Agrawal et al. [AS95], Gibbons et al. [GMP97], Alsabti et al. [ARS97], Manku et al. [MRL98, MRL99], and Greenwald and Khanna [GK01] consider how to compute the approximate median and other quantiles in a single pass over a finite data set. Correlated aggregates are considered in [CR96, ACJK01, Cha99] with the focus on exact computation of correlated aggregates over finite data sets in multiple passes.

The maintenance of aggregate queries is a special case of the the problem of incremental view maintenance; in particular, the maintenance of basic statistical aggregates in the presence of database updates was considered in [OW00]. The synopsis data structures of Matias et al [GM99] consider the approximate maintenance of more fancy aggregates in the presence of updates. In online aggregation, Hellerstein et al. study the convergence of basic aggregates over finite data sets [HHW97] and they describe access methods that retrieve records in random order in order to use statistical estimators based on independence assumptions. Related is also work on the processing on continuous queries [TGNO92], such as in the OpenCQ System [LPT$^+$98], the NiagaraCQ Project [CDTW00], and the XFilter System [AF00].

Distributed Query Processing. There are several great surveys on distributed query processing, such as work by Yu and Chang [YC84], Ceri and Pelagatti [CP84], Özsu and Valduriez [ÖV91], Yu and Meng [YM98], and a recent survey by Kossmann [Kos00].

Sensor Networks. Research in dynamic wireless communication networks has a long history, dating back to DARPA's PRNET (Packet Radio Network) [JT87] and SURAN (Survivable Adaptive Networks) projects [SW87]. Most active in the area of sensor networks has been the networking community, and a plethora of papers has been published on routing protocols for ad-hoc mobile wireless networks [PB94, JM96, BMJ+98, Per99, PC99, ACPR, DPR00, JLH+99]. The networking community has recently started to investigate the issue of power-aware networking infrastructures [CT00, PK00, HKB99, SWR98, RRH00]. PicoNet proposes an integrated design of radios, small, battery powered nodes, and MAC-layer and application protocols that minimize power consumption [BCE+97]. IEEE 802.11 supports ad hoc network configuration and provides control over power management [otICS99].

4 Conclusions

We believe that sensor networks are an exciting new research area with challenging data management and data mining problems. In the Cougar Sensor Database Project and the Amazon Stream Processing Project (`www.cs.cornell.edu/database`), we have started to address some of the problems outlined above.

Acknowledgements. The algorithms on computing correlated aggregates over data streams are joint work with Flip Korn and Divesh Srivastava. Past and current contributors to the Cougar Project are Philippe Bonnet, Zhiyuan Chen, Tony Faradjian, Wai Fu Fung, Tobias Mayr, Daniel Mosse, Praveen Sheshadri, David Sun, and Yong Yao. Past and current contributors to the Amazon Stream Processing Project are Rohit Ananthakrishna, Abhinandan Das, Flip Korn, and Divesh Srivastava.

References

[ABC+00] Amr El Abbadi, Michael L. Brodie, Sharma Chakravarthy, Umeshwar Dayal, Nabil Kamel, Gunter Schlageter, and Kyu-Young Whang, editors. *VLDB 2000, Proceedings of 26th International Conference on Very Large Data Bases, September 10-14, 2000, Cairo, Egypt.* Morgan Kaufmann, 2000.

[ACJK01] Michael Akinde, Damianos Chatziantoniou, Theodore Johnson, and Samuel Kim. The MD-join: An operator for complex OLAP. In *Proceedings of the IEEE International Conference on Data Engineering*, 2001.

[ACM98] ACM SIGMOBILE. *Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM-98).* ACM Press, 1998.

[ACM99] ACM SIGMOBILE. *Proceedings of the Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom-99.* ACM Press, 1999.

[ACPR] J. Agre, L. Clare, G. Pottie, and N. Romanov. Development platform for self-organizing wireless sensor networks. In *Proceedings of the Aerosense.*

[AF00] Mehmet Altinel and Michael J. Franklin. Efficient filtering of xml documents for selective dissemination of information. In Abbadi et al. [ABC+00], pages 53–64.

[AGMS99] Noga Alon, Phillip B. Gibbons, Yossi Matias, and Mario Szegedy. Tracking join and self-join sizes in limited storage. In *Proceedings of the Eighteenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, May 31 - June 2, 1999, Philadelphia, Pennsylvania* [DBL99], pages 10–20.

[AGP99] Swarup Acharya, Phillip B. Gibbons, and Viswanath Poosala. Aqua: A fast decision support systems using approximate query answers. In Atkinson et al. [AOV+99], pages 754–757.

[AGPR99a] Swarup Acharya, Phillip B. Gibbons, Viswanath Poosala, and Sridhar Ramaswamy. Join synopses for approximate query answering. In Delis et al. [DFG99], pages 275–286.

[AGPR99b] Swarup Acharya, Phillip B. Gibbons, Viswanath Poosala, and Sridhar Ramaswamy. The aqua approximate query answering system. In Delis et al. [DFG99], pages 574–576.

[AMS99] Alon, Matias, and Szegedy. The space complexity of approximating the frequency moments. *JCSS: Journal of Computer and System Sciences*, 58, 1999.

[AOV+99] Malcolm P. Atkinson, Maria E. Orlowska, Patrick Valduriez, Stanley B. Zdonik, and Michael L. Brodie, editors. *VLDB'99, Proceedings of 25th International Conference on Very Large Data Bases, September 7-10, 1999, Edinburgh, Scotland, UK.* Morgan Kaufmann, 1999.

[ARS97] Khaled Alsabti, Sanjay Ranka, and Vineet Singh. A one-pass algorithm for accurately estimating quantiles for disk-resident data. In Matthias Jarke, Michael J. Carey, Klaus R. Dittrich, Frederick H. Lochovsky, Pericles Loucopoulos, and Manfred A. Jeusfeld, editors, *VLDB'97, Proceedings of 23rd International Conference on Very Large Data Bases, August 25-29, 1997, Athens, Greece*, pages 346–355. Morgan Kaufmann, 1997.

[AS95] Rakesh Agrawal and Arun Swami. A one-pass space-efficient algorithm for finding quantiles. In Surajit Chaudhuri, Anand Deshpande, and Ravi Krishnamurthy, editors, *Proceedings of the 7th International Conference on Management of Data (COMAD)*, December 1995.

[BCE+97] F. Bennett, D. Clarke, J. Evans, A. Hopper, A. Jones, and D. Leask. Piconet: Embedded Mobile Networking. *IEEE Personal Communications*, 4(5):8–15, October 1997.

[BDF+97] Daniel Barbará, William DuMouchel, Christos Faloutsos, Peter J. Haas, Joseph M. Hellerstein, Yannis E. Ioannidis, H. V. Jagadish, Theodore Johnson, Raymond T. Ng, Viswanath Poosala, Kenneth A. Ross, and Kenneth C. Sevcik. The new jersey data reduction report. *Data Engineering Bulletin*, 20(4):3–45, 1997.

[BMJ+98] Josh Broch, David A. Maltz, David B. Johnson, Yih-Chun Hu, and Jorjeta Jetcheva. A performance comparison of multi-hop wireless ad hoc network routing protocols. [ACM98], pages 85–97.

[CDTW00] Jianjun Chen, David J. DeWitt, Feng Tian, and Yuan Wang. Niagracq: A scalable continuous query system for internet databases. In Weidong Chen, Jeffrey F. Naughton, and Philip A. Bernstein, editors, *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, May 16-18, 2000, Dallas, Texas, USA*, volume 29, pages 379–390. ACM, 2000.

[CFPR00] Corinna Cortes, Kathleen Fisher, Daryl Pregibon, and Anne Rogers. Hancock: a language for extracting signatures from data streams. pages 9–17. ACM Press, 2000.

[Cha99] Damianos Chatziantoniou. Ad hoc OLAP: Expression and evaluation. In *Proceedings of the IEEE International Conference on Data Engineering*, 1999.

[CHNW96] David W. Cheung, Jiawei Han, Vincent T. Ng, and C. Y. Wong. Maintenance of discovered association rules in large databases: An incremental updating technique. In *Proc. Int. Conf. Data Engineering, ICDE*, February 1996.

[CNT96] David W. Cheung, Vincent T. Ng, and Benjamin W. Tam. Maintenance of discovered knowledge: A case in multi-level association rules. In Evangelos Simoudis, Jia Wei Han, and Usama Fayyad, editors, *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96)*, page 307. AAAI Press, 1996.

[CP84] S. Ceri and G. Pelagatti. *Distributed Database Design: Principles and Systems.* MacGraw-Hill (New York NY), 1984.

[CR96] D. Chatziantoniou and K. A. Ross. Querying multiple features of groups in relational databases. In *Proceedings of the International Conference on Very Large Databases*, pages 295–306, 1996.

[CT00] Jae-Hwan Chang and Leandros Tassiulas. Energy conserving routing in wireless ad-hoc networks. In *Proceedings of the 2000 IEEE Computer and Communications Societies Conference on Computer Communications (INFOCOM-00)*, pages 22–31, Los Alamitos, March 26–30 2000. IEEE.

[DBL99] *Proceedings of the Eighteenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, May 31 - June 2, 1999, Philadelphia, Pennsylvania*. ACM Press, 1999.

[DFG99] Alex Delis, Christos Faloutsos, and Shahram Ghandeharizadeh, editors. *SIGMOD 1999, Proceedings ACM SIGMOD International Conference on Management of Data, June 1-3, 1999, Philadephia, Pennsylvania, USA*. ACM Press, 1999.

[DH00] Pedro Domingos and Geoff Hulten. Mining high-speed data streams. In *Proceedings of the Sixth International Conference on Knowledge Discovery and Data Mining*, pages 71–80, Boston, MA, August 2000. ACM.

[DPR00] Samir Das, Charles Perkins, and Elizabeth Royer. Performance comparison of two on-demand routing protocols for ad hoc networks. In *Proceedings of the 2000 IEEE Computer and Communications Societies Conference on Computer Communications (INFOCOM-00)*, pages 3–12, Los Alamitos, March 26–30 2000. IEEE.

[EGHK99] Deborah Estrin, Ramesh Govindan, John Heidemann, and Satish Kumar. Next century challenges: Scalable coordination in sensor networks. [ACM99], pages 263–270.

[FKSV99] Feigenbaum, Kannan, Strauss, and Viswanathan. An approximate L1-difference algorithm for massive data streams (extended abstract). In *FOCS: IEEE Symposium on Foundations of Computer Science (FOCS)*, 1999.

[FKSV00] J. Feigenbaum, S. Kannan, M. Strauss, and M. Viswanathan. Testing and spot-checking of data streams. In *Proceedings of the 11th ACM-SIAM Symposium on Discrete Algorithms*, 2000.

[FS00] Fong and Strauss. An approximate L^p-difference algorithm for massive data streams. In *STACS: Annual Symposium on Theoretical Aspects of Computer Science*, 2000.

[GGR99] Venkatesh Ganti, Johannes Gehrke, and Raghu Ramakrishnan. A framework for measuring changes in data characteristics. In *Proceedings of the Eighteenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, May 31 - June 2, 1999, Philadelphia, Pennsylvania* [DBL99], pages 126–137.

[GGR00] Venkatesh Ganti, Johannes Gehrke, and Raghu Ramakrishnan. Demon: Mining and monitoring evolving data. In *Proceedings of the 16th International Conference on Data Engineering*, pages 439–448. IEEE Computer Society, 2000.

[GGRL99] Johannes Gehrke, Venkatesh Ganti, Raghu Ramakrishnan, and Wei-Yin Loh. Boat-optimistic decision tree construction. In Delis et al. [DFG99], pages 169–180.

[GK01] Michael Greenwald and Sanjeev Khanna. Space-efficient online computation of quantile summaries. In *SIGMOD 2001, Proceedings ACM SIGMOD International Conference on Management of Data, 2001, Santa Barbara, California, USA*. ACM Press, 2001.

[GKS01] Johannes Gehrke, Flip Korn, and Divesh Srivastava. On computing correlated aggregates over continual data streams. In *SIGMOD 2001, Proceedings ACM SIGMOD International Conference on Management of Data, 2001, Santa Barbara, California, USA*. ACM Press, 2001.

[GM98] Phillip B. Gibbons and Yossi Matias. New sampling-based summary statistics for improving approximate query answers. In Haas and Tiwary [HT98], pages 331–342.

[GM99] Phillip B. Gibbons and Yossi Matias. Synopsis data structures for massive data sets. In *Proceedings of the Tenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 909–910, N.Y., January 17–19 1999. ACM-SIAM.

[GMMO00] S. Guha, N. Mishra, R. Motwani, and L. O'Callaghan. Clustering data streams. In *In Proceedings of the Annual Symposium on Foundations of Computer Science*. IEEE, November 2000.

[GMP97] P. Gibbons, Y. Mattias, and V. Poosala. Fast Incremental Maintenance of Approximate Histograms. *Proceedings of VLDB, Athens Greece*, pages 466–475, August 1997.

[HHW97] Joseph M. Hellerstein, Peter J. Haas, and Helen Wang. Online aggregation. In Joan Peckham, editor, *SIGMOD 1997, Proceedings ACM SIGMOD International Conference on Management of Data, May 13-15, 1997, Tucson, Arizona, USA*, pages 171–182. ACM Press, 1997.

[Hil01] Dan Hildebrand. An architectural overview of qnx. http://www.qnx.com/literature/whitepapers/archoverview.html, 2001.

[HKB99] Wendi Rabiner Heinzelman, Joanna Kulik, and Hari Balakrishnan. Adaptive protocols for information dissemination in wireless sensor networks. [ACM99], pages 174–185.

[HRR98] M. R. Henzinger, P .Raghavan, and S. Rajagopalan. Computing on data streams. *Technical Report 1998-011, Digital Eqipment Corporation, Systems Research Center*, May, 1998.

[HSW+00] Jason Hill, Robert Szewczyk, Alec Woo, David Culler, Seth Hollar, and Kristofer Pister. System architecture directions for networked sensors. *ACM SIGPLAN Notices*, 35(11):93–104, November 2000.

[HT98] Laura M. Haas and Ashutosh Tiwary, editors. *SIGMOD 1998, Proceedings ACM SIGMOD International Conference on Management of Data, June 2-4, 1998, Seattle, Washington, USA*. ACM Press, 1998.

[IGE00] Chalermek Intanagonwiwat, Ramesh Govindan, and Deborah Estrin. Directed diffusion: A scalable and robust communication paradigm for sensor networks. pages 56–67. ACM SIGMOBILE, ACM Press, 2000.

[JLH+99] Per Johansson, Tony Larsson, Nicklas Hedman, Bartosz Mielczarek, and Mikael Degermark. Scenario-based performance analysis of routing protocols for mobile ad-hoc networks. [ACM99], pages 195–206.

[JM96] David B Johnson and David A Maltz. Dynamic source routing in ad hoc wireless networks. In Imielinski and Korth, editors, *Mobile Computing*, volume 353 of *The Kluwer International Sereies in Engineering and Computer Science*. Kluwer Academic Publishers, 1996.

[JT87] John Jubin and Janet D. Tornow. The DARPA packet radio network protocol. *Proceedings of the IEEE*, 75(1):21–32, January 1987.

[KKP99] J. M. Kahn, R. H. Katz, and K. S. J. Pister. Next century challenges: Mobile networking for "smart dust". [ACM99], pages 271–278.

[KKPG98] J. Kymissis, C. Kendall, J. Paradiso, and N. Gershenfeld. Parasitic power hasvesting in shoes. In *In Proceedings of the Second IEEE International Conference on Wearable Computing (ISWC)*, pages 132–139. IEEE Computer Society Press, October 1998.

[Kos00] Donald Kossmann. The state of the art in distributed query processing. *Computing Surveys*, 32, 2000.

[LPT+98] Ling Liu, Calton Pu, Wei Tang, David Buttler, John Biggs, Tong Zhou, Paul Benninghoff, Wei Han, and Fenghua Yu. Cq: A personalized update monitoring toolkit. In Haas and Tiwary [HT98], pages 547–549.

[MRL98] Gurmeet Singh Manku, Sridhar Rajagopalan, and Bruce G. Lindsay. Approximate medians and other quantiles in one pass and with limited memory. In Haas and Tiwary [HT98], pages 426–435.

[MRL99] Gurmeet Singh Manku, Sridhar Rajagopalan, and Bruce G. Lindsay. Random sampling techniques for space efficient online computation of order statistics of large datasets. In Delis et al. [DFG99], pages 251–262.

[otICS99] LAN MAN Standards Committee of the IEEE Computer Society. Wireless lan medium access control (mac) and physical layer (phy) specification. IEEE Std 802.11, 1999.

[ÖV91] M. Tamer Özsy and Patrick Valduriez. *Principles of Distributed Database Systems.* Prentice Hall, Englewood Cliffs, 1991.

[OW00] Chris Olston and Jennifer Widom. Offering a precision-performance tradeoff for aggregation queries over replicated data. In Abbadi et al. [ABC+00], pages 144–155.

[PB94] Charles Perkins and Pravin Bhagwat. Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. In *ACM SIGCOMM'94 Conference on Communications Architectures, Protocols and Applications*, pages 234–244, August 1994.

[PC99] V. Park and S. Corson. Temporally-ordered routing algorithm (tora) version 1 functional specication. Internet Draft, http://www.ietf.org/internet-drafts/draft-ietf-manet-tora-spec-02.txt, 1999.

[Per99] Charles E. Perkins. Ad hoc on demand distance vector (aodv) routing. Internet Draft, http://www.ietf.org/internet-drafts/draft-ietf-manet-aodv-04.txt, October 1999.

[PK00] G. J. Pottie and W. J. Kaiser. Embedding the Internet: wireless integrated network sensors. *Communications of the ACM*, 43(5):51–51, May 2000.

[RRH00] R. Ramanathan and R. Rosales-Hain. Topology control of multihop wireless networks using transmit power adjustment. In *In Proceedings of the IEEE Infocom*, pages pages 404–413, March 2000.

[Sto81] Michael Stonebraker. Operating system support for database management. *CACM*, 24(7):412–418, 1981.

[SW87] N. Schacham and J. Westcott. Future directions in packet radio architectures and protocols. *Proceedings of the IEEE*, 75(1):83–99, January 1987.

[SWR98] Suresh Singh, Mike Woo, and C. S. Raghavendra. Power-aware routing in mobile ad hoc networks. [ACM98], pages 181–190.

[Sys01] EMJ Embedded Systems. White dwarf linux. http://www.whitedwarflinux.org, 2001.

[TBAR97] Shiby Thomas, Sreenath Bodagala, Khaled Alsabti, and Sanjay Ranka. An efficient algorithm for the incremental updation of association rules in large databases. In David Heckerman, Heikki Mannila, Daryl Pregibon, and Ramasamy Uthurusamy, editors, *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining (KDD-97)*, page 263. AAAI Press, 1997.

[TGNO92] Douglas B. Terry, David Goldberg, David Nichols, and Brian M. Oki. Continuous queries over append-only databases. In Michael Stonebraker, editor, *Proceedings of the 1992 ACM SIGMOD International Conference on Management of Data, San Diego, California, June 2-5, 1992*, pages 321–330. ACM Press, 1992.

[uCl01] uClinux. The linux/microcontroller project. http://www.uclinux.org, 2001.

[YC84] C. T. Yu and C. C. Chang. Distributed query processing. *ACM Computing Surveys*, 16(4):399–433, December 1984.

[YM98] C. Yu and W. Meng. *Principles of Database Query Processing for Advanced Applications.* Morgan Kaufmann, San Francisco, 1998.