Research Statement

Murat Demirbas

My research area is distributed systems. The fundamental challenge in distributed systems is to coordinate the behavior of nodes which execute concurrently with limited information about each other. For safety reasons concurrency needs to be tamed to prevent undesired executions, on the other hand, for performance reasons, concurrency needs to be boosted to achieve efficiency.

My research focus is to **develop principled, efficient, and resilient protocols for coordination in distributed systems**. I have investigated distributed coordination in the context of different application domains, including wireless sensor/actor networks, smartphone-based crowdsourced sensing, and most recently in the context of large scale coordination in cloud computing.

My approach to distributed coordination is rooted in principled **invariant-based reasoning** about the correctness and fault-tolerance of systems, eschewing the case-by-case view considered by operational reasoning. I design protocols leveraging strong algorithmic bases/abstractions and employ TLA+ modeling [1] to verify the correctness and fault-tolerance of these distributed protocols. I then tailor and optimize these protocols for various environments by employing systems techniques, considering the tradeoffs in the design space. The resulting protocols are elegant, coherent, performant, and correct against corner cases, failed assumptions, concurrency, and failures.

To keep my work relevant and practical, I source my research questions from industry relevant needs. My recent work on hybrid logical clocks has been adopted by several big distributed database systems, including MongoDB, Couchbase, and CockroachDB. To keep in contact with the industry, I spent a year at Microsoft Azure Cosmos DB on sabbatical, and learned about the challenges of dealing with big data processing systems at extreme scale. I practice open research and disseminate my results as well as my thought process to the practitioners in the field. My blog [2] has seen more than 2 million pageviews, and has a substantial following both from industry and academia.

Next, I briefly summarize my current and past research projects. Each project description is organized into problem, approach, and results sections.

1 Current research: Paxos Unpacked

Problem: Due to their excellent fault-tolerance and consistency benefits, Paxos protocols are employed at the core of many distributed systems infrastructures. While there has been a lot of work on Paxos, there is still a vast unexplored algorithm design space. A striking evidence of this arrived in 2017 with the flexible quorum [3] breakthrough which came unanticipated almost 30 years after the Paxos protocol was first proposed. This further opened up the design space for customizing Paxos to different environments and workloads, which is yet to be realized. We argue that when properly tailored and optimized, Paxos family of protocols can deliver efficiency, performance, and scalability at par with weakly consistent protocols, while providing a stable and strong foundation to build services and applications on top.

Approach: Scaling Paxos and making it practical and customizable involves many challenges and require an assortment of distributed algorithmic tools and techniques. For scaling Paxos horizontally, we investigate novel in-protocol sharding and reconfiguration strategies enabled by the flexible quorums result. For scaling Paxos vertically, we alleviate the bottleneck at the leaders by decoupling control from communication flow and by employing in-protocol aggregation optimizations. To provide consensus as a lightweight serverless primitive, we develop efficient single-instance consensus as a systems primitive and showcase its use in applications. For improving the efficiency of strongly-consistent read operations, we investigate novel quorum read techniques. As another fundamental contribution, we show how to relax/detach the decades-old accidental coupling of linearizability with Paxos/consensus protocols, by providing sharded, flexible, and finer granularity variants of Paxos.

Expected contributions: The techniques developed for in-protocol sharding across leaders, alleviating the communication bottlenecks via in-protocol aggregation, and performing linearizable reads from quorums will have broader applicability beyond Paxos protocols to other strong consistency protocols. The systems primitives developed for realizing serverless consensus will also benefit other coordination protocols. In order to accelerate technology transfer to academia and industry, the techniques developed in this project will be showcased in the context of a globally distributed database prototype, called FleetDB `https://github.com/acharapko/fleetdb`. FleetDB has the potential to lead the way for integrating novel tailored Paxos protocols in replication operations of distributed databases. We are collaborating with prominent distributed database teams in the industry to make this happen. Finally, with the convergence of Byzantine fault-tolerant (BFT) consensus protocols to Paxos like leader-to-all and all-to-all communication patterns (thanks to the use of threshold certificates), we expect that our contributions in this project will be applicable for distributed ledger and blockchain systems.

2 Synchrony-aware primitives for building highly auditable, scalable, available distributed systems (funded by NSF XPS 2015-2019)

Problem: Auditability is a key property for developing highly scalable and highly available distributed systems; auditability enables identifying performance bottlenecks, dependencies among events, and latent concurrency bugs. In turn, for the auditability of a system, time is a key concept. However, there is a gap between the theory and the practice of distributed systems in terms of the use of time. The theory of distributed systems shunned the notion of time and considered asynchronous systems, whose event ordering is captured by logical clocks. The practical distributed systems employed NTP synchronized clocks to capture time but did so in ad hoc undisciplined ways.

Approach: Our work focused on providing auditability by combining two key concepts: time and causality. In particular, we prescribed hybrid logical clocks (HLC) [4,5] which offer the functionality of logical clocks while keeping them *close* to physical clocks. HLC combines the theoretical underpinnings of causality and the practicality of physical clocks by identifying how logical clocks can be improved and tuned based on the availability of NTP synchronization. The principle guiding HLC design is "uncertainty resilience". HLC is designed to be always wait-free/nonblocking and correct (albeit with reduced efficiency) even when time synchronization has degraded or is not available.

Contributions: HLC clocks enabled highly auditable systems [6] since they can efficiently provide global consistent-state snapshots without needing to wait out clock synchronization uncertainties and without requiring prior coordination. Our project found applications to cloud computing, distributed NewSQL databases, and globally distributed web services. Our HLC clocks have recently been adopted by CockroachDB [7], an opensource clone of Google Spanner, MongoDB [8], and Couchbase [9].

We also leveraged the auditability support provided by HLCs and designed and developed a monitoring service, Retroscope [10], that detects distributed system state corruptions. Retroscope's cut monitoring approach is complementary to the request tracing solutions, and brings a number of advantages. First, by exposing the nonlocal state, Retroscope enables users to examine nonlocal properties and invariants of distributed applications. Furthermore, by sifting through many past nonlocal states, users can perform root-cause analysis and diagnose race conditions, nonlocal state inconsistencies, and nonlocal invariant violations. To achieve a scalable implementation of Retroscope, we leveraged Apache Ignite for stream processing, computation, and storage. We arranged the log ingestion way to minimize data movement and improve data locality and achieve maximal parallelism when searching. The resultant publications from this project include [10–15]. All research products, including publications, software, and source code are made available to the public openly at `https://cse.buffalo.edu/~demirbas/Auditable/`.

3 Scalable coordination for wide-area distributed systems (funded by NSF CSR 2015-2018)

Problem: Scalable coordination of concurrently executing tasks is a challenging open problem for large-scale distributed systems. Especially with increasing demand for large-scale web-services for e-commerce, social networking, and Internet of Things, the coordination of tasks over wide-area (i.e., across clusters, across datacenters, and across Internet) has recently gained greater importance. Traditional distributed coordination techniques fail to scale for wide-area networks to support these new generation applications. Centralized coordination fails to scale with respect to the increased distances in the wide-area, whereas decentralized coordination fails to scale with respect to the number of nodes involved.

Approach: To achieve scalable coordination of distributed tasks over wide-area, we proposed a novel hybrid design that avoids the shortcomings of both fully-centralized and fully-decentralized solutions to coordination. Our approach was to shard Paxos variants with multileaders and build migration policies for moving data according to access patterns among the leaders.

Contributions: The project developed a system for hierarchical coordination, called WanKeeper [16]. WanKeeper provided wide-area optimized ZooKeeper deployment for distributed metadata management. By adjusting to the locality of accesses, the top-level WanKeeper cluster gives tokens/autonomy to the relevant leaf-level WanKeeper clusters. The performance of WanKeeper was showcased in the context of a wide-area filesystem. To provide a protocol level, flexible, and general solution, the project also developed WPaxos [17]. By using the flexible quorum result, WPaxos showed how to apply in-Paxos sharding to multileaders over zones. The project also performed an analytical and empirical dissection of Paxos performance and identified bottlenecks in Paxos protocols. The resultant publications from this project include [16,18–22]. More information can be found at `https://cse.buffalo.edu/~demirbas/WanCoord/`.

4 Smartsourced sensing and collaboration (funded by Google Research 2010, NIH/NIEHS 2010-2012, NSF CRI 2012-2015)

Problem: Although smartphones showed a lot of promise for solving the large-scale sensing problem and fulfilling the ubiquitous computing vision of pervasive collaboration, they fall short of their potentials. Consider DARPA's 2009 network grand challenge on accurately finding 10 red weather balloons deployed at arbitrary locations of the US. The winning team managed to solve the challenge in 9 hours, but the team had to prepare, campaign, and publicize aggressively for a month, and employed a multilevel incentive structure that distributed the $40K prize money among participants. To achieve full potential of smartphone based sensing and collaboration, a platform is needed to enable development of smartsourcing apps that solve similar collaboration and coordination problems without requiring month-long campaigns and $40K awards.

Approach: We believe that the reason current apps fail to solve problems like 10 red balloons is the lack of a platform to utilize smartphones for collaboration and coordination. In contrast, providing a platform for publish/subscribe and tasking of these devices would enable any smartphone to utilize the data published by other smartphones in a region and to task other smartphones in the region to acquire new data if needed. In order to utilize smartphones for solving collaboration and coordination problems, we proposed an open publish-subscribe middleware at the cloud backend, *Eywa*, which enables smartphones to benefit from data collected by other smartphones and Internet of Things (IoT).

Contributions: In our early work we employed Twitter as a primitive publish-subscribe middleware and developed a crowdsourced weather radar [23] and citywide sensing applications over Twitter [24–26]. To improve the success rate of location-based services, we built a system [27] that categorizes Twitter users based on their familiarity to location types in Foursquare and uses this information to forward more relevant queries to the users.

Our work on mobile user profiling [28–35] aimed to improve the efficiency of our smartsourced sensing system. As a limited scope demonstration of the Eywa vision, we employed the localization capabilities on the smartphones and leveraged on the computational power provided by cloud servers in order to estimate and forecast the wait times at coffee shops; our system, LineKing, became the first crowdsourced line wait-time estimation service [36].

Since aggregating responses from a crowd is a challenge for the Eywa platform, we investigated multiple-choice question answering (MCQA) [37–40]. To this end, we designed a gamified experiment. We developed an Android app to let the crowd answer questions with their smartphones as they watch the Who Wants To Be A Millionaire (WWTBAM) quiz show on a Turkish TV channel. Our WWTBAM app has been downloaded and installed more than 300,000 times and has enabled us to collect more than 3 GB of MCQA data (2000 live quiz-show questions and more than 200,000 answers) over a period of 9 months. We developed novel MCQA aggregation algorithms to improve the accuracy of answers to more than 90% even for the cases where majority voting resulted in 40% accuracy [37–39].

5 Efficient and resilient querying and tracking services for WSNs (funded by Office of Naval Research 2009-2012)

Problem: The goal of an in-network querying service in WSNs is to answer spatial queries, such as "What is the location of the nearest enemy tank to my coordinates?". To answer such queries, querying and tracking services require continuous maintenance of distributed data structures (trees, paths, and clusters) over a large number of nodes. In order to achieve scalability, only the relevant WSN nodes should be involved in the execution of the query. That is, these services should implement local operations over these global structures. Locality is also needed in handling of the faults. In the absence of a local healing mechanism, faults/inconsistencies in one part of the system may contaminate the entire system and hence may result in a high-cost, system-wide correction.

Approach: Our approach in achieving locality and scalability of WSN services is to exploit geometric ideas and techniques while devising distributed algorithms. In contrast to Internet, where the topology is logical and arbitrary graph models are used, WSNs are deployed in physical spaces and the use of geometric networks are warranted for modeling WSNs. We find that when the problem domain is constrained to geometric networks it is possible to devise simpler and more efficient algorithms than those designed for arbitrary graph topologies. Especially for reasoning about locality of solutions in WSN (where communication cost is the biggest constraint on design) geometric methods are a good fit. Furthermore, by exploiting the geometry of the network, we propose efficient fault-containment techniques to enable locality in handling of faults.

Contributions: For in-network querying, we proposed a *distributed quad-tree (DQT) structure* [41] that exploits node localization information to achieve local construction without requiring any communication. DQT achieved a querying cost of $2\sqrt{2} * d$, and, due to its minimalist infrastructure and stateless nature, it showed graceful resilience to the face of failures. In *Glance* [42], we improved on our earlier results. Our main insight was to use the basestation node in an opportunistic manner for answering of some in-network queries, and show that in-network advertisements can safely ignore a majority of directions/regions and focus their advertisement to a small cone to be able to satisfy a given distance-sensitivity requirement. As a result, Glance ensures that a query invoked within d distance of an event intercepts the event's advertisement within $d * s$ distance, where s is a "stretch-factor" tunable by the user. For tracking in WSNs, in *Stalk* [43], we employed hierarchical partitioning to maintain a locally-healing tracking structure over a small number of nodes and with accuracy proportional to the distance from the evader. In *Trail* [44], we presented a tracking protocol that achieves the same linear costs for find and update *without* requiring a hierarchical partitioning of the network. We also applied our geometric ideas to devise energy-efficient low-latency data collection mechanisms for WSNs using network-controlled mobile basestations [45–47].

6 An in-network collaboration and coordination framework for wireless sensor/actor networks (funded by NSF-CAREER 2008-2013)

Problem: As WSNs get increasingly more integrated with actuation capabilities, consistency and timeliness guarantees become significant issues. However, in the presence of unreliable communication channels—as is the case in WSNs— even the most basic consensus problem of getting nodes agree on a binary decision is unsolvable (due to the well-known Coordinated Attack impossibility result). Therefore, effectively managing concurrent execution ranks as one of the biggest challenges for future wireless sensor-actor networks (WSANs).

Approach: Our key insight in this project is to observe that singlehop wireless broadcast has many useful features for facilitating collaboration and coordination. Firstly, broadcasting is atomic (i.e., for all the recipients of a broadcast, the reception occurs simultaneously), which is useful for synchronizing the nodes in singlehop for building a structured operation. Secondly, broadcast messages share the same medium enabling collision detection and snooping, which are useful for implementing collaboration and coordination in a decentralized manner. Leveraging these properties, we developed a framework that provides simple programming abstractions to cope with the consistent coordination challenges of WSANs while retaining efficiency of execution. Our framework consists of two components: (1) a singlehop communication primitive for fast robust feedback collection, and (2) a transactional abstraction for robust computing in WSANs.

Contributions: In [48] we have shown, for the first time, that it is possible to solve consensus efficiently in WSNs using a novel receiver-side collision detection (RCD) technique. The idea here was to provide a dedicated round for communicating negative feedback, and hence conveying information even when the negative feedback messages collide. While the transmitter cannot detect collisions in WSNs, there is no barrier against the RCD. In our algorithm, a collision detected in the veto round indicates the existence of at least one veto message and that the consensus should be deferred for a later round. We have also given a classification of RCD with respect to its completeness (ability to detect collisions) and accuracy (ability to avoid false positives) and identified the lower-bounds for solving consensus for each class. We developed reliable implementations and quantitative evaluations of RCD on TinyOS and mote platforms in [49]. Using RCD, we built a primitive "pollcast" for quick and robust singlehop feedback collection from a singlehop neighborhood. We have analyzed the theoretical lowerbounds and upperbounds associated with querying with pollcast in [50]. In [51], we developed and showcased applications of RCD for singlehop collaboration protocols in WSNs, including a quick and robust threshold querying primitive *tcast*. We also applied these principles to the reliable broadcast problem in [52,53], which is another relevant and significant problem for single-hop communication. **Our receiver-side collision detection technique has been adopted by several WSN MAC layers since then.**

As complementary to the singlehop communication primitive for fast robust feedback collection, we designed and developed the TRANSACT framework [54, 55] which provides an efficient and lightweight implementation of a transaction primitive in a distributed manner by exploiting the properties of broadcast communication inherent in WSANs. A major contribution of TRANSACT is to simplify the reasoning and verification of a distributed WSANs program. Building blocks for process control and coordination programs (such as, leader election, mutual exclusion, cluster construction, neighborhood discovery, recovery actions, and consensus) are easy to denote using TRANSACT.

References

[1] L. Lamport. The temporal logic of actions. *ACM Transactions on Programming Languages and Systems*, 16(3):872–923, May 1994.

[2] The metadata blog. `http://muratbuffalo.blogspot.com`, 2019.

[3] Heidi Howard, Dahlia Malkhi, and Alexander Spiegelman. Flexible Paxos: Quorum intersection revisited. *arXiv preprint arXiv:1608.06696*, 2016.

[4] S. Kulkarni, M. Demirbas, D. Madappa, B. Avva, and M. Leone. Logical physical clocks. In *Principles of Distributed Systems*, pages 17–32. Springer, 2014.

[5] M. Demirbas and S. Kulkarni. Beyond truetime: Using augmentedtime for improving google spanner. *LADIS '13: 7th Workshop on Large-Scale Distributed Systems and Middleware*, 2013.

[6] M. Demirbas and S. Kulkarni. Highly auditable distributed systems. *USENIX HotCloud*, 2015.

[7] Cockroachdb: A scalable, transactional, geo-replicated data store. http://cockroachdb.org/.

[8] Kristina Chodorow and Michael Dirolf. *MongoDB: The Definitive Guide*. O'Reilly Media, Inc., 1st edition, 2010.

[9] Couchbase distributed nosql database. https://www.couchbase.com, 2019.

[10] Aleksey Charapko, Ailidani Ailijiang, Murat Demirbas, and Sandeep Kulkarni. Retroscope: Retrospective monitoring of distributed systems. *IEEE Transactions on Parallel and Distributed Systems*, 2019.

[11] Duong Nguyen, Aleksey Charapko, Sandeep Kulkarni, and Murat Demirbas. Using weaker consistency models with monitoring and recovery for improving performance of key-value stores. *Journal of the Brazilian Computer Society*, 25(1), 2019.

[12] Mohammad Roohitavaf, Murat Demirbas, and Sandeep Kulkarni. Causalspartan: Causal consistency for distributed data stores using hybrid logical clocks. In *2017 IEEE 36th Symposium on Reliable Distributed Systems (SRDS)*, pages 184–193. IEEE, 2017.

[13] S. Kulkarni, M. Demirbas, D. Madappa, B. Avva, and M. Leone. Logical physical clocks. In *Principles of Distributed Systems*, pages 17–32. Springer, 2014.

[14] Sorrachai Yingchareonthawornchai, Vidhya Tekken Valapil, Sandeep S. Kulkarni, Eric Torng, and Murat Demirbas. Efficient algorithms for predicate detection using hybrid logical clocks. In *International Conference on Distributed Computing and Networking (ICDCN)*, 2017.

[15] Aleksey Charapko, Ailidani Ailijiang, Murat Demirbas, and Sandeep Kulkarni. Retrospective lightweight distributed snapshots using loosely synchronized clocks. In *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*, pages 2061–2066. IEEE, 2017.

[16] Ailidani Ailijiang, Aleksey Charapko, Murat Demirbas, Bekir O Turkkan, and Tevfik Kosar. Efficient distributed coordination at wan-scale. In *IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*, pages 1575–1585. IEEE, 2017.

[17] L. Lamport. Paxos made simple. *ACM SIGACT News*, 32(4):18–25, 2001.

[18] A. Ailijiang, A. Charapko, and M. Demirbas. Dissecting the performance of strongly-consistent replication protocols. In *ACM Sigmod*, pages 1696–1710, 2019.

[19] A. Charapko, A. Ailijiang, and M. Demirbas. Linearizable quorum reads in Paxos. In {*USENIX*} *HotStorage*, 2019.

[20] A. Ailijiang, A. Charapko, M. Demirbas, and T. Kosar. WPaxos: Wide area network flexible consensus. *IEEE Transactions on Parallel and Distributed Systems*, 2019.

[21] Aleksey Charapko, Ailidani Ailijiang, and Murat Demirbas. Bridging paxos and blockchain consensus. In *IEEE International Conference on Blockchains*, pages 1545–1552. IEEE, 2018.

[22] Ailidani Ailijiang, Aleksey Charapko, and Murat Demirbas. Consensus in the cloud: Paxos systems demystified. In *25th International Conference on Computer Communication and Networks (ICCCN)*, pages 1–10. IEEE, 2016.

[23] M. Demirbas, M. A. Bayir, C. G. Akcora, Y. Yilmaz, and H. Ferhatosmanoglu. Crowd-sourced sensing and collaboration using twitter. *11th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2010.

[24] C. G. Akcora, M. A. Bayir, M. Demirbas, and H. Ferhatosmanoglu. Identifying breakpoints in public opinion. *Workshop on Social Media Analytics*, 2010.

[25] B. Sriram, D. Fuhry, E. Demir, H. Ferhatosmanoglu, and M. Demirbas. Short text classification in twitter to improve information filtering. *33rd ACM SIGIR Conference*, 2010.

[26] `http://ubicomp.cse.buffalo.edu/citypulse`.

[27] M. Demirbas M. F. Bulut, Y. S. Yilmaz. Crowdsourcing location-based queries. *IEEE Workshop on Pervasive Collaboration and Social Networking (PerCol)*, 2011.

[28] M. A. Bayir, M. Demirbas, and N. Eagle. Discovering spatiotemporal mobility profiles of cellphone users. *Pervasive and Mobile Computing Journal, Special Issue on Human Behaviour in Ubiquitous Environments*, 6(4):435–454, 2010.

[29] M. A. Bayir, M. Demirbas, and A. Cosar. A web based personalized mobility service for smartphone applications. *The Computer Journal*, 54(5):800–814, 2011.

[30] M. A. Bayir and M. Demirbas. Pro: A profile based routing algorithm for pocket switched networks. *IEEE Global Communications Conference (Globecom)*, 2010.

[31] M. Demirbas, C. Rudra, A. Rudra, and M. A. Bayir. imap: Indirect measurement of air pollution with cellphones. *PerCOM*, pages 537–542, 2009.

[32] M Glasgow, L Mu, P Nayak, C Crabtree-Ide, M Demirbas, E Yoo, A Szpiro, A Rudra, J Merriman, J Wactawski-Wende, et al. Smartphone technology for improving air pollution exposure estimates. In *American Journal of Epidemiology*, volume 175, pages S9–S9, 2012.

[33] M. L. Glasgow, C. B Rudra, E-H Yoo, M. Demirbas, J. Merriman, P. Nayak, C. Crabtree-Ide, A. A. Szpiro, A. Rudra, J. Wactawski-Wende, et al. Using smartphones to collect time–activity data for long-term personal-level air pollution exposure assessment. *Journal of Exposure Science and Environmental Epidemiology*, 2014.

[34] A. Nandugudi, A. Maiti, F. Bulut, S. Batra, Y Ki, G. Challen, M. Demirbas, S. Ko, T. Kosar, and C. Qiao. Participant behavior in phonelab. *Third International Conference on the Analysis of Mobile Phone Datasets (NetMob)*, 2013.

[35] M. F. Bulut and M. Demirbas. A holistic approach for energy efficient proximity alert on android. In *Global Communications Conference (GLOBECOM), 2013 IEEE*, pages 2816–2821. IEEE, 2013.

[36] M.F. Bulut, Y.S. Yilmaz, M. Demirbas, H. Ferhatosmanoglu, and N. Ferhatosmanoglu. Lineking: Crowdsourced line wait-time estimation using smartphones. In *MobiCASE: Fourth International Conference on Mobile Computing, Applications and Services*, 2012.

[37] B. Aydin, Y. Yilmaz, Y. Li, Q. Li, J. Gao, B. Aydin, and M. Demirbas. Crowdsourcing for multiple-choice question answering. *Twenty-Sixth Annual Conference on Innovative Applications of Artificial Intelligence (IAAI-14)*, 2014.

[38] Y. S. Yilmaz, B. Aydin, and M. Demirbas. Targeted question answering on smartphones utilizing app based user classification. *International Conference on Collaboration Technologies and Systems (CTS)*, 2014.

[39] Qi Li, Yaliang Li, Jing Gao, Lu Su, Bo Zhao, Murat Demirbas, Wei Fan, and Jiawei Han. A confidence-aware approach for truth discovery on long-tail data. *Proceedings of the VLDB Endowment*, 8(4), 2014.

[40] Y. S. Yilmaz, B. I. Aydin, and M. Demirbas. Google cloud messaging (gcm): An evaluation. In *Global Communications Conference (GLOBECOM), 2014 IEEE*, pages 2807–2812. IEEE, 2014.

[41] M. Demirbas, X. Lu, and P. Singla. An in-network querying framework for wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 2009.

[42] M. Demirbas, A. Arora, and V. Kulathumani. A lightweight querying service for wireless sensor networks. *Theoretical Computer Science Journal, Elsevier*, 2009. To appear.

[43] M. Demirbas, A. Arora, T. Nolte, and N. Lynch. A hierarchy-based fault-local stabilizing algorithm for tracking in sensor networks. *8th International Conference on Principles of Distributed Systems (OPODIS)*, pages 299–315, 2004.

[44] V. Kulathumani, M. Demirbas, A. Arora, and M. Sridharan. Trail: A distance sensitive network service for distributed object tracking. *ACM Transactions on Sensor Networks*, 2009. To appear.

[45] M. Demirbas, O. Soysal, and A. S. Tosun. Data salmon: A greedy mobile basestation protocol for efficient data collection in wireless sensor networks. *IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS)*, pages 267–280, 2007.

[46] O. Soysal and M. Demirbas. Data spider: A resilient mobile basestation protocol for efficient data collection in wireless sensor networks. *The 6th IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS)*, 2010.

[47] S.K. Yoon, O. Soysal, M. Demirbas, and C. Qiao. Coordinated locomotion of mobile sensor networks. *Fifth Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, 2008.

[48] G. Chockler, M. Demirbas, S. Gilbert, N. A. Lynch, C. C. Newport, and T. Nolte. Consensus and collision detectors in radio networks. *Distributed Computing*, 21(1):55–84, 2008.

[49] M. Demirbas, O. Soysal, and M. Hussain. Singlehop collaborative feedback primitives for wireless sensor networks. *INFOCOM*, pages 2047–2055, 2008.

[50] J. Aspnes, E. Blais, M. Demirbas, R. O'Donnell, A. Rudra, and S. Uurtamo. k+ decision trees. *International Workshop on Algorithms for Sensor Systems, Wireless Ad Hoc Networks and Autonomous Mobile Entities (Algosensors)*, 2010.

[51] M. Demirbas, S. Tasci, H. Gunes, and A. Rudra. Singlehop collaborative feedback primitives for threshold querying in wireless sensor networks. *25th IEEE International Parallel & Distributed Processing Symposium (IPDPS)*, 2011.

[52] M. Demirbas and M. Hussain. A mac layer protocol for priority-based reliable broadcast in wireless ad hoc networks. *BroadNets*, 2006.

[53] M. Demirbas and S. Balachandran. Robcast: A singlehop reliable broadcast protocol for wireless sensor networks. *The Sixth International Workshop on Assurance in Distributed Systems and Networks (ADSN)*, 2007.

[54] M. Demirbas, O. Soysal, and M. Hussain. Transact: A transactional programming framework for wireless sensor/actor networks. *IEEE/ACM International Conference on Information Processing in Sensor Networks (IPSN)*, pages 295–306, 2008.

[55] O. Soysal, B. I. Aydin, and M. Demirbas. Optimistic concurrency control for multihop sensor networks. *IWCMC*, pages 89–94, 2011.