ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

An ISO 3297: 2007 Certified Organization
Vol. 3, Special Issue 3, April 2014

International Conference on Signal Processing, Embedded System and Communication Technologies and their applications for Sustainable and Renewable Energy (ICSECSRE '14)

Organized by

Department of ECE, Aarupadai Veedu Institute of Technology, Vinayaka Missions University,

Paiyanoor-603 104, Tamil Nadu, India

Proactive Route Optimization Protocol for Mobile Sink in Wireless Sensor Network

Madasamy P, P Subramanian

PG Student, Department of ECE, Aarupadai Veedu Institute of Technology, Vinayaka Missions University, Chennai, India

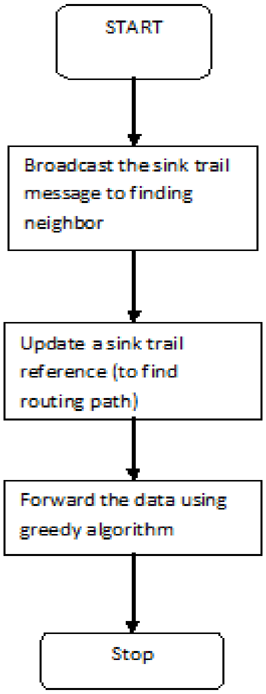
Associate Professor, Department of ECE, Aarupadai Veedu Institute of Technology, Vinayaka Missions University, Chennai, India

Abstract: Wireless sensor networks (WSNs) have enabled a wide spectrum of applications through networked low-cost low-power sensor nodes. Mobile sinks, such as animals or vehicles equipped with radio devices, are sent into a field and communicate directly with sensor nodes, resulting in shorter data transmission paths and reduced energy consumption. Data gathering using mobile sinks introduces new challenges to sensor network applications. To better benefit from the sink's mobility, many research efforts have been focused on studying or scheduling movement patterns of a mobile sink to visit some special places in a deployed area, in order to minimize data gathering time. In such approaches a mobile sink moves to pre-determined sojourn points and query each sensor node individually. The work proposed does not fix predefined trajectories for the mobile sink but allows flexible movement over different terrains. Simulations in the Network simulator of the Proactive Data Reporting Protocol for Wireless Sensor Networks are performed to analyses the efficiency and benefits of using Sink Trail in the upcoming wireless sensor applications.

I.INTRODUCTION

Wireless sensor networks (WSNs) have enabled a wide spectrum of applications through networked low-cost low-power sensor nodes, e.g., habitat monitoring, precision agriculture, and forest fire detection. In these applications, the sensor network will operate under few human interventions either because of the hostile environment or high management complexityfor manual maintenance. Since sensor nodes have limited battery life, energy saving is of paramount importance in the design of sensor network protocols. Recent research on data collection reveals that, rather than reporting data through long, multihop, and errorprone routes to a static sink using tree or cluster network structure, allowing and leveraging sink mobility is more promising for energy efficient data gathering. Mobile sinks, such as animals or vehicles equipped with radio devices, are sent into a field and communicate directly with sensor nodes, resulting in shorter data transmission paths and reduced energy consumption. However, data gathering using mobile sinks introduces new challenges to sensor network applications. To better benefit from the sink's mobility, many research efforts have been focused on studying or scheduling movement patterns of a mobile sink to visit some special places in a deployed area, in order to minimize data gathering time. In such approaches a mobile sink moves to pre-determined sojourn points and query each sensor node individually. Although several mobile elements scheduling (MES) protocols have been proposed to achieve efficient data collection via controlled sink mobility, determining an optimal moving trajectory for a mobile sink is itself an NP-hard problem, and may not be able to adapt to constrained access areas and changing field situations. Take the precision agriculture application. where mobile sinks collecting data mainly follow trails or field boundaries in order not to damage crops, and change trajectories dynamically according to farmland situations. Typically, without scheduling the trajectory for a mobile sink in advance; a data gathering protocol using mobile sinks suggests that a mobile sink announce its location information frequently throughout the network. Many sink-oriented data dissemination protocols use such approach, e.g., Directed Diffusion, Declarative Routing Protocol (DRP), and GRAB, whereas different aggregation methods may be adopted. This class of methods is referred to as SODD in literature. This approach is much more flexible in terms of sinks' movement, but incurs significant control message overheads. An example data reporting path of SODD is presented using black solid route In addition to large amount of energy consumption on flooding control messages, change of routing paths due to the sinks' movement, and energy cost on detouring large data packets (originally targeted at the previous sink location, now changed to the current sink location) severely impair protocol performances. This problem can be alleviated by transmitting data via the shortest route to the mobile sink's future locations, as observed in the red dashed route. Therefore if sensors can predict the mobile sink's movement, the energy consumption would be greatly reduced and data packets handoff would be smoother. In this paper, we propose SinkTrail, a proactive data reporting protocol that is self-adaptive to various application scenarios, and its improved version, SinkTrail-S, with further control message suppression. In SinkTrail, mobile sinks move continuously in the field in relatively low speed, and gather data on the fly. Control messages are broadcasted at certain points in much lower frequency than ordinarily required in existing data gathering protocols. These sojourn positions are viewed as "*footprints*" of a mobile sink. Considering each footprint as a virtual landmark, a sensor node can conveniently identify its hop count distances to these landmarks. These hop count distances combined represent the sensor node's coordinate in the logical coordinate space constructed by the mobile sink. Similarly, the coordinate of the mobile sink is its hop count distances from the current location to previous virtual landmarks. Having the destination coordinate and its own coordinate, each sensor node greedily selects next hop with the shortest logical distance to the mobile sink. As a result, SinkTrail solves the problem of movement prediction for data gathering with mobile sinks. Our contributions in this paper are manifold. It propose a unique logical coordinate representation for tracking mobile sinks without assistance of GPS devices or predefined landmarks, which is widely applicable to various network settings and scenarios. We design a novel low-complexity dynamic routing protocol for data gathering with one or multiple mobile sink(s), which effectively reduces average route length and cuts down total energy consumption. We propose and evaluate an enhanced SinkTrail protocol, called SinkTrail-S..We conduct extensive comparison studies and simulations with popular existing solutions.

FLOWCHART


START
Broadcast the sink trail message to finding neighbor
Update a sink trail reference (to find routing path)
Forward the data using greedy algorithm
Stop

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

An ISO 3297: 2007 Certified Organization Vol. 3, Special Issue 3, April 2014

International Conference on Signal Processing, Embedded System and Communication Technologies and their applications for Sustainable and Renewable Energy (ICSECSRE '14)

Organized by

Department of ECE, Aarupadai Veedu Institute of Technology, Vinayaka Missions University,

Paiyanoor-603 104, Tamil Nadu, India

II.MODULES

Network Topology Generation:

In communication networks, a topology is a usually schematic description of the arrangement of a network, including its nodes and connecting lines. There are two ways of defining network geometry: the physical topology and the logical (or signal) topology. The physical topology of a network is the actual geometric layout of workstations. Logical (or signal) topology refers to the nature of the paths the signals follow from node to node. The number nodes is going to participate in the simulation is decided. Here we conduct experiments to a group of wireless nodes in a network that operates on a suitable protocol. We hence use only a logical topology as it is wireless environment.

Finding neighbor

During the data gathering process, the mobile sink moves around in N with relatively low speed, and keeps listening for data report packets. It stops at some places for a very short time, broadcasts a message to the whole network, and moves on to another place. We call these places "Trail Points", and these messages "Trail Messages". Let τ be the average transmission range. Apparently two adjacent trail points should be separated by a distance longer than τ , otherwise, the hop count information won't be significantly different. To facilitate the tracking of a mobile sink, we assume that the distances between any two consecutive trail points are same (or similar), denoted as $K\tau, K \geq 1$. However, distribution of these trail points doesn't necessarily follow any pattern. A trail message from a mobile sink contains a sequence number (msg.seqN) and a hop count (msg.hopC) to the sink. The time interval between a mobile sink stops at one trail point and arrives at the next trail point is called one "move". There are multiple moves during a data gathering round.

Destination identification

SinkTrail facilitates the flexible and convenient construction of a logical coordinate space. Instead of scheduling a mobile sink's movement, it allows a mobile sink to spontaneously stop at convenient locations according to current field situations or desired moving paths. These sojourn places of a mobile sink, named trail points in SinkTrail, are footprints left by a mobile sink, and they provide valuable information for tracing the current location of a mobile sink. Considering these footprints as virtual landmarks, hop count information reflects the moving trajectory of a mobile sink. A logical dv- dimensional coordinate space is then established.

Forwarding data

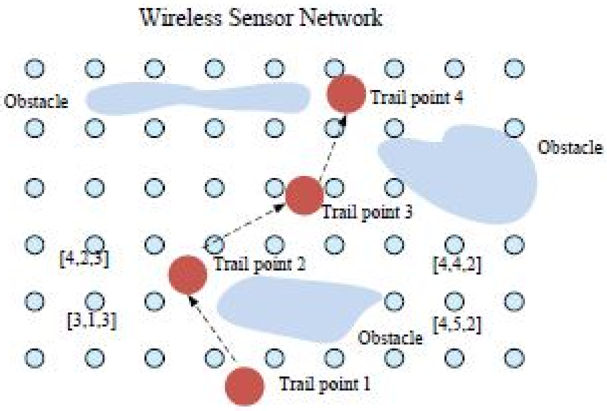
The data reporting procedure consists mainly two phases. The first phase is called logical coordinate space construction. During this phase, sensor nodes update their trail references corresponding to the mobile sink's trail messages. After dv hop counts have been collected, a sensor node enters the greedy forwarding phase, where it decide how to report data packets to the mobile sink.

Performance Evaluation:

During simulation time the events are traced by using the trace files. The performance of the network is evaluated by executing the trace files. The events are recorded into trace files while executing record procedure. In this procedure, we trace the events like packet received, Packets lost, Last packet received time etc. These trace values are write into the trace files. This procedure is recursively called for every 0.05 ms. so, trace values recorded for every 0.05 ms.

III.SINKTRAIL PROTOCOL DESIGN

3.1 Problem formulation



Data gathering with one mobile sink: large solid dotsindicate the mobile sink's trail points, and sensor nodes maintaintrail references as logical coordinates. Shaded areas stand forobstacles.

We consider a large scale, uniformly distributed sensor network N deployed in an outdoor area. Nodes in the network communicate with each other via radio links. We assume the whole sensor network is connected, which is achieved by deploying sensors densely. We also assume sensor nodes are awake when data gathering process starts (by synchronized schedule or a short "wake up" message). In order to gather data from N, we periodically send out a number of mobile sinks into the field. These mobile sinks, such as robots or vehicles with laptops installed,
have radios and processors to communication with sensor nodes and processing sensed data. Since energy supply of mobile sinks can be replaced or recharged easily, they are assumed to have unlimited power. A data gathering process starts from the time mobile sinks enter the field and terminates when: either (1) enough data are collected (measured by a user defined threshold); or (2) there are no more data reports in a certain period. The SinkTrail protocol is proposed for sensor nodes to proactively report their data back to one of the mobile sinks. To illustrate our data gathering algorithm clearly, we first consider the scenario where there is only one mobile sink in N.

3.2 SinkTrail protocol with one mobile sink

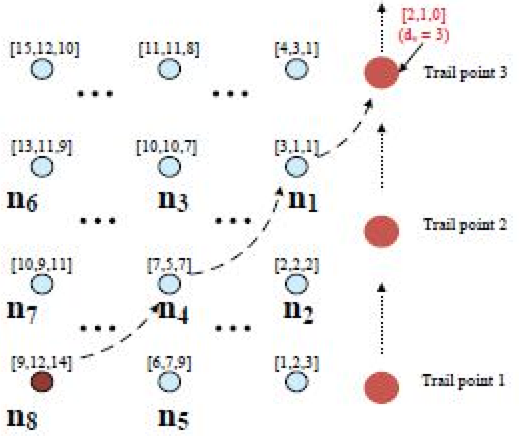
During the data gathering process, the mobile sink moves around in N with relatively low speed, and keeps listening for data report packets. It stops at someplaces for a very short time, broadcasts a message to the whole network, and moves on to another place. We call these places *"Trail Points"*, and these messages *"Trail Messages"*. Let τ be the average transmission range. Apparently twoadjacent trail points should be separated by a distance longer than τ , otherwise, the hop count information won't be significantly different. To facilitate the tracking of a mobile sink, we assume that the distances between any two consecutive trail points are same (or similar), denoted as $K\tau, K \geq 1$. However, distribution of these trail points doesn't necessarily follow any pattern. A trail message from a mobile sink contains a sequence number (*msg.seqN*) and a hop count (*msg.hopC*) to the sink. The time interval between a mobile sink stops at one trail point and arrives at the next trail point is called one *"move"*. There are multiple moves during a data gathering round. The tasks of a mobile sink is summarized in Algorithm

In the SinkTrail algorithm, we use vectors called *"TrailReferences"* to represent logical coordinates in a network.
The trail reference maintained by each node is used asa location indicator for packet forwarding. All trail referencesare of the same size. Notations used throughout the protocol.The data reporting procedure consists mainly twophases. The first phase is called *logical coordinate spaceconstruction*. During this phase, sensor nodes updatetheir trail references corresponding to the mobile sink's trail messages. After *dv* hop counts have been collected, a sensor node enters the *greedy forwarding* phase, where it decide how to report data packets to the mobile sink.

3.2.1 *Logical coordinate space construction*

At beginning, all sensor nodes' trail references are initialized to [−1,−1, . . . ,−1] of size *dv*. A special variable *λ* that is used to track the latest message sequence number is also set to −1. After the mobile sink *S* enters the field, it randomly select a place as its first trail point *π*1, and broadcasts a trail message to all the sensor nodes in N. The trail message, <*msg.seqN,msg.hopC*>, is set to <1, 0 >, indicating that this is the first trail message from trail point one, and the hop count to *S* is zero.

The nodes nearest to *S* will be the first ones to hear this message. By comparing with *λ*, if this is a new message, then *λ* will be updated by the new sequence number. And node *ni*'s trail reference *vi* is updated as follows. First, every element in *vi*is shifted to left by one position. Then, the hop count in the received trail message is increased by one, and replaces the right-most element. After its trail reference, this trail message is rebroadcasted with the same sequence number and an incremented hop count. The same procedure repeats at all the other nodes in N. Within one move of *S*, all nodes in the network have updated their trail references according to their hop count distances to *S*'s trail point *π*1. If a node receives a trail message with a sequence number equals to *λ*, but has a smaller hop count than it has already recorded, then the last hop count field in its trail reference is updated, and this trail message is rebroadcasted with the same sequence number and an incremented hop count. Trail messages that has sequence number less than *λ* will be discarded to eliminate flooding messages in the network. The steps described in Algorithm 2 summarizes the operations to update a trail reference. During the data gathering procedure, a node's trail reference needs to be updated every time a new trail message is received.



Example execution snapshot of SinkTrail: large soliddots indicate trail points and its moving path.

After each node in the network received *dv* distinct trail messages, the logical coordinate space is established. A snapshot of a part of the network N. Trail references, such as [3, 1, 1] or [2, 2, 2], are considered logical coordinates of the sensor nodes in a network.

3.2.2 *Destination identification*

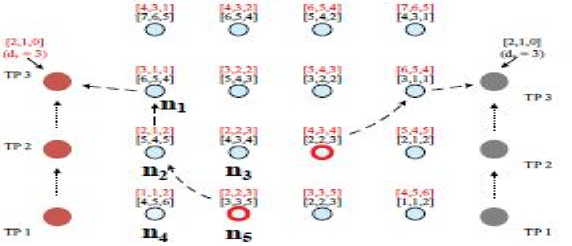
SinkTrail facilitates the flexible and convenient construction of a logical coordinate space. Instead of scheduling a mobile sink's movement, it allows a mobile sink to spontaneously stop at convenient locations according to current field situations or desired moving paths. These sojourn places of a mobile sink, named trail points in SinkTrail, are footprints left by a mobile sink, and they provide valuable information for tracing the current location of a mobile sink. Considering these footprints as virtual landmarks, hop count information reflects the moving trajectory of a mobile sink. A logical *dv*dimensional coordinate space is then established. One advantage of SinkTrail is that the logical coordinate of a mobile sink keeps invariant at each trail point, given the continuous update of trail references. This is because the mobile sink's hop count distance to its previous *dv*−1 footprints are always *K*(*dv*−1),*K*(*dv*−2), . . . , *K*, and 0 to its current location. Therefore the logical coordinate $[K(dv-1), K(dv-2), \ldots, 0]$ represents the current logical location of the mobile sink. We call this coordinate "*Destination Reference*". This destination reference does not necessarily require a mobile sink to have linear moving trajectory. Although arbitrary movement of a mobile sink may deteriorate the accuracy of destination reference, it can still serve as a guideline for data reporting. Here we set $K = 1$ and $dv = 3$ to ease our presentation. A large value of K means even less broadcast frequency. The impacts of mobile sinks' moving pattern and broadcast frequency are investigated. In Fig, assume S is at the trail point 3 now, then its destination reference should be [2, 1, 0]. When S moves to the trail point 4, the coordinate space is updated based on trail points 2, 3, and 4, and destination reference of the mobile sink is still [2, 1, 0].

3.2.3 *Greedy forwarding*

Once a node has updated the 3 elements in its trail reference (we use $dv = 3$ for easy understanding and clear presentation), it starts a timer that is inverse proportional to the right-most element in its trail reference. For example, node $n5$'s trail reference is [6, 7, 9], then the duration of its timer is set to $T5 = Tinit - \mu \times 9$. Here, $Tinit$and μ are predefined constants. The choice of timer function, $Tinit$, and μ may vary. However, we assume the timer durations are significantly longer than the propagation time of a trail message, so that timers on all nodes are viewed as starting at the same time. The timer mechanism is mainly used to differentiate data reporting orders (another usage is discussed in SinkTrail- S protocol); so the clock on each sensor node doesn't need to be perfectly synchronized. Since the right-most element in a node's trail reference is the latest hop count information from this node to a mobile sink, the inverse proportional timers ensure that nodes faraway from S have shorter timer durations than those close to S, thus will start data reporting first. When a node's timer expires, it initiates the data reporting process.

Every sensor node in the network maintains a routing table of size $O(b)$ consisting of all neighbors' trail references. This routing table is built up by exchanging trail references with neighbors, as described in Algorithm 3; and it is updated whenever the mobile sink arrives at a new trail point. Although trail references may not be global identifiers since route selection is conducted locally, they are good enough for the SinkTrail protocol. Because each trail reference has only 3 numbers, the size of exchange message is small. When a node has received all its neighbors' trail references, it calculates their distances to the destination reference, [2, 1, 0], according to *2-norm* vector calculation, then greedily chooses the node with the smallest distance as next hop to relay data. If there is a tie the next hop node can be randomly selected. The complete procedure of greedy forwarding is presented in Algorithm 3. when node $n8$ decides to report its data, it compares $n4$, $n5$, and $n7$'s vector distance with [2, 1, 0]. Since $n5$ and $n7$'s distance to [2, 1, 0] is$\sqrt{133}$ and$\sqrt{249}$respectively, and $n4$'s distance is$\sqrt{90}$, $n4$ is chosen as thenext hop of $n8$.

3.3 SinkTrail protocol with multiple mobile sinks



Example execution snapshot of SinkTrail of multiplemobile sinks scenario.

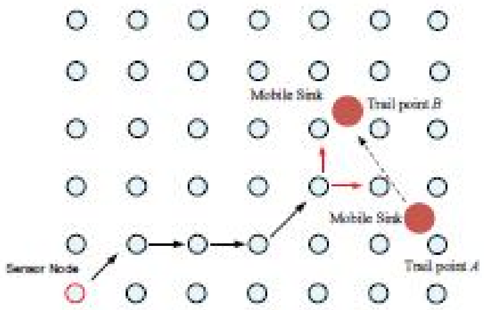
The proposed SinkTrail protocol can be readily extended to multi-sink scenario with small modifications. When there is more than one sink in a network, each mobile sink broadcasts trail messages following Algorithm 1. Different from one sink scenario, a sender ID field, *msg.sID*, is added to each trail message to distinguish them from different senders.Algorithms executed on the sensor node side should be modified to accommodate multi-sink scenario as well.

Instead of using only one trail reference, a sensor node maintains multiple trail references that each corresponds to a different mobile sink at the same time. Two trail references, colored in black and red, coexist in the same sensor node. In this way, multiple logical coordinate spaces are constructed concurrently, one for each mobile sink. When a trail message arrives, a sensor node checks the mobile sink's ID in the message to determine if it is necessary to create a new trail reference. The procedure is summarized in Algorithm 4. In SinkTrail trail references of each node represent node locations in different logical coordinate spaces, when it comes to data forwarding, because reporting to any mobile sink is valid, the node can choose the neighbor closest to a mobile sink in any coordinate space. Sink location in each logical coordinate space is still [2, 1, 0], as we use K = 1, dv = 3. If each mobile sink has a different K value, sensor nodes will calculate neighbors' distances to multiple destination references and select route accordingly. Detailed description is in Algorithm 5. It is well-known that geographic routing and virtual coordinate based routing ensure loop-free routes [6], [18], so does SinkTrail gives us an example of data gathering in multiplecoordinate spaces.

For node n5, its neighbor node n2's vector distance to [2, 1, 0] with regard to the red mobile sink on the left is 2, and√43 to the right grey mobile sink. And all other neighbors of n5 has larger vector distance to the two sinks. So n2 is used as the next hop to route to the red mobile sink.

3.4 SinkTrail-S protocol

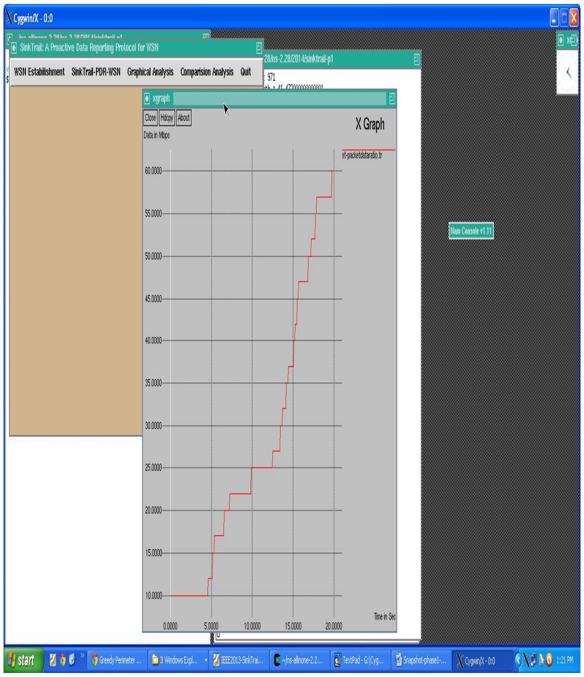
In SinkTrail, flooding trail messages to the whole network can be nontrivial in terms of energy consumption. To further optimize the energy usage and eliminate unnecessary control messages in the network, we propose SinkTrail-S algorithm as an improvement to the original SinkTrail. SinkTrail-S algorithm is mainly based on the following two observations. First, in a large-scale sensor An illustrative example to show that mobile sink's movementhas less impact on remote sensor nodes than immediateones. Network, the sensor nodes that are far away from a mobile sink may not be significantly affected by a single movement of the mobile sink. Take the sensor network shown, when the mobile sink moves from trail point A to trail point B, the yellow sensor node at the left bottom corner may still have the same hop count distance to the mobile sink, and the routing path chosen from last *"move"* of the mobile sink may still be valid. In this case, the trail messages can be suppressed with high probability. Second, when a node has finished data reporting and forwarding, trail reference updating becomes meaningless and results in huge waste of energy, especially for peripheral sensor nodes. To properly handle these two situations, we propose a message suppression policy at a small cost of extra state storage at each sensor node. Each sensor node will compare the current hop count distance to a mobile sink with the most recently received one. If these two are same, it indicates the path length through the node to the mobile sink is still same, making it unnecessary to rebroadcast this trail message. In case of the second situation, each node maintains a state variable in its memory. When a node finishes data reporting, it marks itself as "*finished*", and informs all its neighbor nodes. A node stops trail reference updating and trail message rebroadcasting whenever itself and all its neighbors are "*finished*". Again, this method is guaranteed by the timer mechanism that ensures sequential data packets reporting



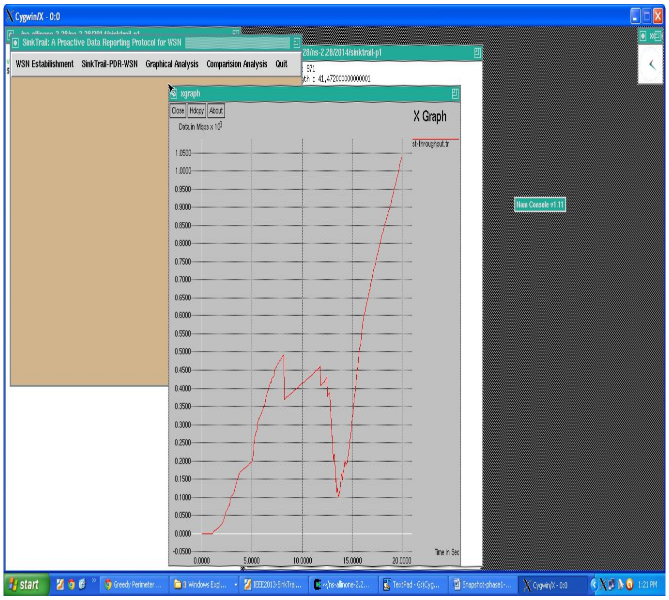
order from network peripheral to a mobile sink's current location. For accidental situations due to timer failure, a new data packet may arrive at a node that has already stopped trail reference updating. In that case old trail references are used. This may cause a longer routing path but the result is still acceptable for data reporting. Algorithm 6 presents a detail description of SinkTrail-S protocol.

IV.SIMULATION RESULTS

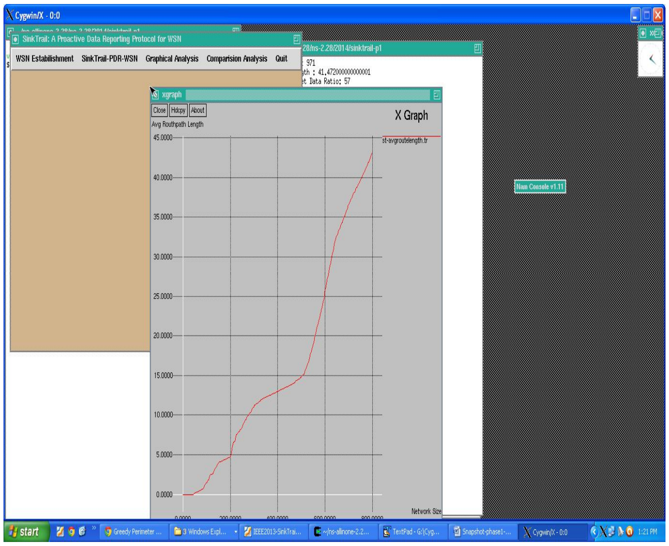
To demonstrate the effectiveness of message suppressionin SinkTrail-S, we simulated SinkTrail-S with
circular and linear sink moving patterns and comparethe result with the basic SinkTrail protocol. It is worthnoting that energy cost for informing neighbors is alsocounted in implementation.



```
971
```



```
971
th : 41.472000000000001
```



```
971
th : 41.472000000000001
t Data Ratio: 57
```

V.CONCLUSION

We presented the SinkTrail and its improved version, SinkTrail-S protocol, two low-complexity, proactive data reporting protocols for energy-efficient data gathering. SinkTrail uses logical coordinates to infer distances, and establishes data reporting routes by greedily selecting the shortest path to the destination reference. In addition, SinkTrail is capable of tracking multiple mobile sinks simultaneously through multiple logical coordinate spaces. It possesses desired features of geographical routing without requiring GPS devices or extra landmarks installed. SinkTrail is capable of adapting to various sensor field shapes and different moving patterns of mobile sinks. Further, it eliminates the need of special treatments for changing field situations. We systematically analyzed energy consumptions of Sink- Trail and other representative approaches and validated our analysis through extensive simulations. The results demonstrate that SinkTrail finds short data reporting routes and effectively reduces energy consumption. The impact of various design parameters used in SinkTrail and SinkTrail-S are investigated to provide guidance for implementation.

REFERENCES

[1] S. Basagni, A. Carosi, E. Melachrinoudis, C. Petrioli, and Z. M.Wang. Controlled sink mobility for prolonging wireless sensornetworks lifetime. *ACM/Elsevier Wireless Networks*, 2007.

[2] C. Chou, K. Ssu, H. Jiau, W. Wang, and C. Wang.A dead-endfree topology maintenance protocol for geographic forwarding inwireless sensor networks.*IEEE Transactions on Computers*, 2010.

[3] D. Coffin, D. Van Hook, S. McGarry, and S. Kolek. Declarativead-hoc sensor networking.In *Proceedings of SPIE*, volume 4126,page 109, 2000.

[4] M. Demirbas, O. Soysal, and A. Tosun. Data salmon: A greedymobile basestation protocol for efficient data collection in wirelesssensor networks. *Distributed Computing in Sensor Systems*, pages267–280, 2007.

[5] K. Fodor and A. Vid´acs. Efficient routing to mobile sinks inwireless sensor networks. In *Proceedings of the 3rd internationalconference on Wireless Internet (WICON)*, pages 1–7, 2007.

[6] R. Fonseca, S. Ratnasamy, J. Zhao, C. T. Ee, D. Culler, S. Shenker,and I. Stoica. Beacon vector routing: Scalable point-to-pointrouting in wireless sensornets. In *Proceedings of NSDI*, pages 329–342, 2005.

[7] Q. Huang, C. Lu, and G. Roman.Spatiotemporal multicast insensor networks. In *Proceedings of the 1th ACM Conference onEmbedded Networked Sensor Systems (SenSys)*, page 217. ACM, 2003.

[8] C. Intanagonwiwat, R. Govindan, and D. Estrin.Directed diffusion:A scalable and robust communication paradigm for sensornetworks. In *Proceedings of the 6th annual international conference on Mobile computing and networking (MobiCom)*, pages 56–67. ACMNew York, NY, USA, 2000.

[9] M. Keally, G. Zhou, and G. Xing. Sidewinder: A predictive dataforwarding protocol for mobile wireless sensor networks. In *6thAnnual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, pages 1–9, June2009.

[10] P. Levis, N. Lee, M. Welsh, and D. Culler.Tossim: accurate andscalable simulation of entire tinyos applications. In *Proceedingsof the 1th ACM Conference on Embedded Networked Sensor Systems(SenSys)*, pages 126–137, 2003.

[11] Z. Li, N. Wang, A. Franzen, P. Taher, C. Godsey, H. Zhang, andX. Li. Practical deployment of an in-field soil property wirelesssensor network,.*(in press) Computer Standards & Interfaces, Elsevier*,2011.

[12] B. Liu, W. Ke, C. Tsai, and M. Tsai.Constructing a messagepruningtree with minimum cost for tracking moving objects inwireless sensor networks is np-complete and an enhanced dataaggregation structure.*IEEE Transactions on Computers*, pages 849–863, 2008.

[13] J. Luo and J.-P.Hubaux.Joint mobility and routing for lifetimeelongation in wireless sensor networks.In *Proceedings of IEEEINFOCOM*, volume 3, 2005.

[14] M. Ma and Y. Yang. Data gathering in wireless sensor networkswith mobile collectors. In *Proceedings of the IEEE InternationalSymposium on Parallel and Distributed Processing (IPDPS)*, pages1–9, April 2008.

[15] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson.Wireless sensor networks for habitat monitoring. In*Proceedings of the 1st ACM international workshop on Wireless sensornetworks and applications (WSNA)*, pages 88–97, New York, NY,USA, 2002. ACM.

[16] T. Moscibroda, R. O'Dell, M. Wattenhofer, and R. Wattenhofer.Virtual coordinates for ad hoc and sensor networks. In *Proceedingsof the joint workshop on Foundations of mobile computing (DIALMPOMC)*,pages 8–16, New York, NY, USA, 2004. ACM.