ISSN:0975-9646

A Novel Dynamic Data Collection Paradigm for Wireless Sensor Networks (WSNs) by Mobile Users

[1]Zara Sumaiya, [2]Rekha Patil

[1]P.G.Student, Department of Computer Science & Engineering,
VTU RO PG Centre, Gulbarga, Karnataka, India.

[2]Associate Professor & H.O.D., Department of Computer Science & Engineering,
PDACE, Gulbarga, Karnataka, India.

Abstract– **A novel dynamic data collection paradigm enables the mobile users to access the network wide data from anywhere and anytime along their walks. In this paper, we propose a new and powerful protocol for mobile users to collect their dynamic data in a ubiquitous way. As the mobile user moves, the routing structure of data collection (tree) is additively updated. Here, a little modification is done to the existing routing structure while the routing performance is confined and assured with regard to the optimal value. The proposed protocol is easy to carry out. We provide strong analysis and extensive simulations to evaluate the efficiency and scalability of our protocol. The results showed that our protocol can scale absolutely well with the addition of mobile users, delivers the correct data continuously during the user movement and performs efficiently in the routing performance.**

Keywords: **protocol, wireless sensor networks, dynamic data collection, mobile users.**

I. INTRODUCTION

A wireless sensor network comprises large number of spatially distributed and dedicated sensor nodes. Each node is equipped with processor, memory, radio transceiver, battery and sensors. These sensor nodes form a short-range wireless communication network after being deployed.

In recent years, wireless sensor networks are widely deployed and used for environmental monitoring to build a affirm society. WSN are widely used in many domains like emergency operations, security surveillance, habitat monitoring, precision agriculture, logistic, information enquiry, and transmission, [1], [2], [3], [4], etc. With the progress of handheld devices, ubiquitous data collection is a way for mobile users to collect the data from the wide network via nearby sensors by using their handheld devices like PDAs, mobile phones, etc. This enhances the ease of sensor network deployment and offers a facilitated means of sensor data collection. Mobile users can endorse data collection in variety of sensing applications such as environmental monitoring, health care, transportation, security, etc. For example, the Personal Environment Impact Report (PEIR) is a mobile sensing application that uses geographical data from mobile phones to calculate how we impact the environment and also how the environment impact on us [5].

Mobile users (Mobile phones) can collect variety of sensing data by communicating with wireless sensors in their environment. For instance, in the GreenOrbs project [6], forest rangers patrol around the Tianmu Mountain to collect scientific data such as humidity, temperature, amount of carbon dioxide, and so on, from the sensors deployed in the forest. This data helps the forest ranger to detect fire, vegetation damage in the forest.

II. RELATED WORK

Data collection is the basic function in wireless sensor network, a typical data collection protocol provides for the creation and sustenance of one or more data collection routing tree with each sink as their root [7]. Thus, the sink can upload the data to the internet or other databases for further analyzing.

The Collection Tree Protocol (CTP) is a link and loop detection based data collection protocol. It provides two routing mechanism, namely, adaptive beaconing and data path validation to route the packets from sensor nodes to (one or more) sink nodes in the presence of highly dynamic link topology. To attain reliability, robustness, efficiency, and hardware independence in CTP, standard implementation has been proposed by [9] and evaluated by [10], [8]. Implementation of CTP [9] consists of three major subcomponents: link estimator, routing engine, and forwarding engine. [8] further indicated out the link dynamics and transient loops are the dominant causes for only 2–68% of delivery ratio.

Mobile sinks and mobile relays have been suggested for improving the performance of data collection in sensor networks [11], [12]. The most existing works focused on how to plan the moving trajectory for the mobile user or sink to collect the data efficiently i.e., it compensate the time cost of data collection transitions [13], [14]. R. Tan et al. [13] exploits reactive mobility to better the target detection and performance. Mobile sensors and static sensors work together to move reactively in [13]. The above works centered upon controlling the mobility of mobile sinks for data collection, which is different from the mobile users with independent and uncontrollable movement of users in our work.

Some recent work focus on mobile users or sinks without any fixed trajectory. Kusy et al. [15] presented an algorithm to predict the future data collection position of the mobile sinks from the training data. They computed and maintained the mobility graph of the mobile sinks to improve routing reliability in data collection process. Similarly, Lee et al. [16] presented a routing scheme that exploits the mobility pattern of the mobile sinks to optimize the prediction accuracy. The above works mainly focus on predicting the mobile user's movement to improve the routing efficiency.

In this paper, we solve the problem of ubiquitous data collection process for mobile phone users in a wireless sensor network by locally modifying the previously constructed data collection tree with the movement of mobile users in the sensor network.

III. EXISTING SYSTEM

A. Traditional Data Collection

It is based on static settings. Usually an optimal data collection tree is built in static sensor network to collect the network wide-data. The data collection tree is pre-determined and serves to effectively deliver data to the static sink. As the mobile user moves, it requires the ubiquitous data access. Thus, the data collection tree created at one point is not enough in the presence of user mobility. This demand for creating and updating the data collection tree, i.e., the routing structure must be updated regularly with the movement of mobile users.

B. Disadvantages

1) Creation of independent data collection trees at different positions of mobile users.
2) Introduces large volume of communication overheads and non-negligible delay.
3) There is no guarantee of data delivery due to frequent changes in mobile user's location.

IV. PROPOSED SYSTEM

Different from traditional WSNs, dynamic data collection does not rely on a stationary sink to collect sensing data from the whole network. Instead, mobile users collect sensing data from their surrounding sensors pervasively using their mobile devices. The uncontrollable mobility of the mobile users and the limited wireless communication range pose new challenges in dynamic data collection. In particular, the contact time between the mobiles and the sensors can be very short given the continuous and potentially fast movement of mobile users.

In this paper, we design and implement dynamic data collection protocol, which updates the data collection tree from time to time with the user's movement. Here, a little change is done to the existing data collection tree which results in the creation of a new data collection tree which is light-weighted in terms of both time efficiency and communication overheads. Moreover, the proposed protocol is easy to carry out and the resulting routing performance on the new data collection tree is confined and assured with regard to the optimal value.

The proposed protocol serves the mobile users with the continuous data streams within the sensor network data by keep on updating the data collection tree during routing transitions.

Fig. 1 shows the basic architecture of the proposed system. Mobile users start their data collection journey. A virtual sink 1 is formed by the node detecting mobile user. The other sensor nodes deliver the data packet to the virtual sink 1. Mobile users collect all the data from the virtual sink 1. Whenever mobile users move away from the virtual sink 1, a new virtual sink 2 is formed. The sensor nodes in the previous cluster send the data through new virtual sink 2.

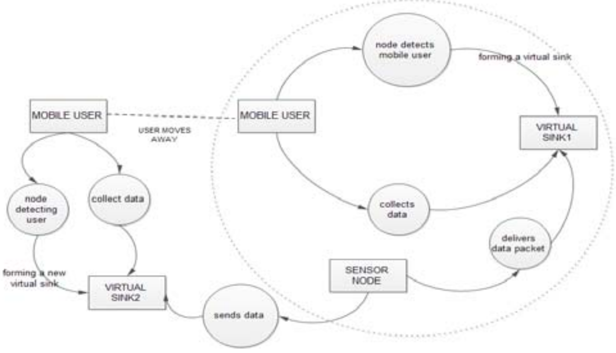


Fig. 1: Basic Architecture

A. Advantages

1) Ascertains that data collection delay is low.
2) Provides the ability of acquiring the real-time data to the mobile users with unlimited mobility paths.
3) The mobile user probing process does not introduce extra communication costs.
4) The movement of mobile users where used to optimize routing transitions.
5) Proposed protocol is compatible with existing mobility prediction mechanisms.
6) Maintains the routing structure in very large scale.
7) The data collection process is fluent.

B. Modules

1) Wireless Sensor Network Creation and Routing
2) Construction of Data Collection Tree
3) Performance Analysis
4) Implementation of Ubiquitous Data Collection Method
5) Performance Analysis and Result Comparison

Modules Description:

1) Wireless Sensor Network Creation and Routing:

In this module, a wireless sensor network is created. Sensor nodes are randomly deployed in the network area. Base station / Sink node is configured using PHENOM agent. All the sensor nodes are connected with wireless links. The sensor nodes can act as router and forward the data packets to the next nodes. The sensor nodes send their data to the base station node by constructing the best path among them and forward the data packets.

2) Construction of Data Collection Tree:

A node is randomly selected and configured as mobile user. The mobile user can move across the network and collect the data from the sensor nodes. A data collection tree is formed by the sink node. The mobile user can collect the data packets from the constructed tree and deliver them to the base station.

3) Performance Analysis:

In this module, the performance of Data aggregation method is analyzed. Based on the analyzed results X-graphs are plotted. Throughput, delay, energy consumption are the basic parameters are considered here and X-graphs are plotted for these parameters.

4) Implementation of Ubiquitous Data Collection Method:

In this module, the data collection tree is updated dynamically according with the movement of the mobile user. Once the mobile user moves from one collection point (virtual-sink) to another, the data collection tree updates the tree to communicate with mobile nodes through newly formed collection-point.

5) Performance Analysis and Result Comparison:

In this module, the performance of the proposed architecture is analyzed. Based on the analyzed results X-graphs are plotted. Throughput, delay, energy consumption are the basic parameters considered here and X-graphs are plotted for these parameters.

Finally, the results obtained from this module is compared with third module results and comparison X-graphs are plotted. Form the comparison result, final RESULT is concluded.

V. SYSTEM DESIGN

Our dynamic data collection protocol employs the spatial correlation to effectively construct and modify the data collection tree. Whenever the mobile user changes the location i.e. virtual sink, the existing data collection tree is modified locally and thus provides new data collection tree efficiently. Therefore, here we present the detailed design of three elements in our protocol:

A) Data Collection Tree Construction
B) Data Collection Tree Modification
C) Data Routing

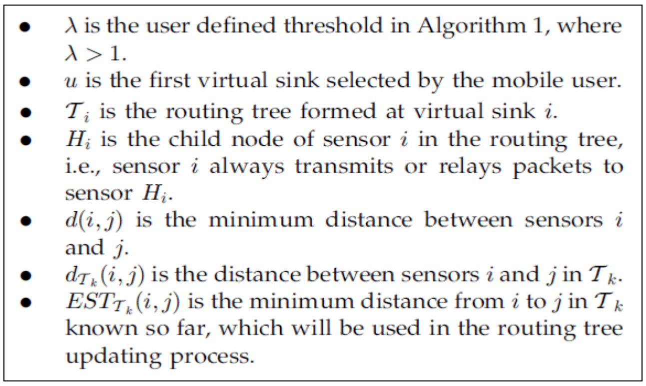
A. Data Collection Tree Construction

Initially data collection tree is constructed. Here we consider full sensor network as a graph G = {V, E}, where the vertex set V denotes the static sensor and the edge set denotes the communicational links. At the start out, a virtual sink, u ∈ V is formed by the mobile user to access the data from the sensor network.

An optimal routing tree in this phase is made as follows: The virtual sink node broadcasts a control message for the construction of routing tree and the initial cost (e.g. hop-count distance) at each sensor node to the virtual sink node is set to infinity. Universally, by interchanging information, sensor i sets up its child node H_i to be the neighbor with the minimum cost to the virtual sink compared with all other neighbors. Once H_i is updated, sensor i will notify its neighbors and they can update their own configurations accordingly.

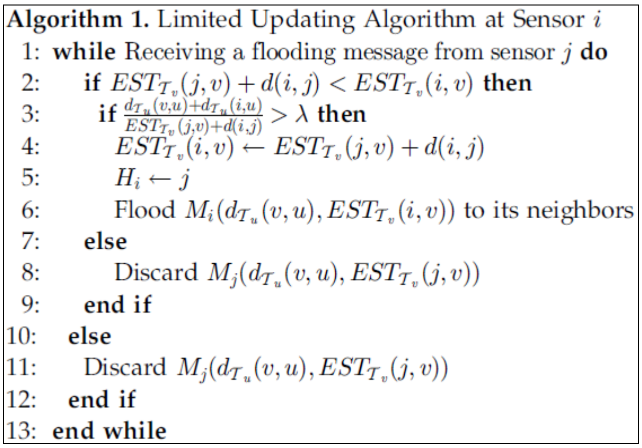
B. Data Collection Tree Modification

The notations used in this algorithm are shown in Fig. 2. An implemental account of our proposed protocol (Fig. 3) is as follows: The protocol is generally triggered by a series of flooding messages. One flooding message contains two types of information: 1) d_{T_u}(v, u) and 2) EST_{T_v}(j, v), where j is the sender of this message. Such a message is denoted as $M_j(d_{T_u}(v, u), EST_{T_v}(j, v))$. If sensor i receives it, d_{T_u}(v, u) can be used to calculate the distance from i to v in T_u, i.e., d_{T_u}(i, u) + d_{T_u}(v, u), and EST_{T_v} (j, v) can be used to update EST_{T_v} (i, v) ← EST_{T_v} (j, v) + d (i, j).



- λ is the user defined threshold in Algorithm 1, where $\lambda > 1$.
- u is the first virtual sink selected by the mobile user.
- $\mathcal{T}_i$ is the routing tree formed at virtual sink i.
- H_i is the child node of sensor i in the routing tree, i.e., sensor i always transmits or relays packets to sensor H_i.
- $d(i,j)$ is the minimum distance between sensors i and j.
- $d_{\mathcal{T}_k}(i,j)$ is the distance between sensors i and j in $\mathcal{T}_k$.
- $EST_{\mathcal{T}_k}(i,j)$ is the minimum distance from i to j in $\mathcal{T}_k$ known so far, which will be used in the routing tree updating process.

Fig. 2: Notations used



```
Algorithm 1. Limited Updating Algorithm at Sensor $i$
while Receiving a flooding message from sensor $j$ do
    if $EST_{\mathcal{T}_v}(j,v) + d(i,j) < EST_{\mathcal{T}_v}(i,v)$ then
        if $\frac{d_{\mathcal{T}_u}(v,u) + d_{\mathcal{T}_u}(i,u)}{EST_{\mathcal{T}_v}(j,v) + d(i,j)} > \lambda$ then
            $EST_{\mathcal{T}_v}(i,v) \leftarrow EST_{\mathcal{T}_v}(j,v) + d(i,j)$
            $H_i \leftarrow j$
            Flood $M_i(d_{\mathcal{T}_u}(v,u), EST_{\mathcal{T}_v}(i,v))$ to its neighbors
        else
            Discard $M_j(d_{\mathcal{T}_u}(v,u), EST_{\mathcal{T}_v}(j,v))$
        end if
    else
        Discard $M_j(d_{\mathcal{T}_u}(v,u), EST_{\mathcal{T}_v}(j,v))$
    end if
end while
```

Fig. 3: Pseudo code of the Limited Updating Algorithm at Sensor i

Virtual sink v reverses the path v ⇒ u (in T_u) to u ⇒ v first, and then launches the Data Collection Tree Modification process by broadcasting $M_v(d_{T_u}(v, u), EST_{T_v}(v, v))$ to all its neighbors. Note that EST_{T_v} (v, v) = 0 and the initial value of EST_{T_v} (i, v) for any i ≠ v equals +∞. In general, after sensor i receives $M_j(d_{T_u}(v, u), EST_{T_v}(j, v))$, sensor i calculates

$$\frac{d_{T_u}(v, u) + d_{T_u}(i, u)}{EST_{T_v}(j, v) + d(i, j)}$$

If EST_{T_v}(j, v) + d (i, j) < EST_{T_v} (i, v) and

$$\frac{d_{T_u}(v, u) + d_{T_u}(i, u)}{EST_{T_v}(j, v) + d(i, j)} > \lambda$$

sensor i updates the value of $EST_{T_v}(i, v)$ as $EST_{T_v}(j, v) + d(i, j)$ changes H_i to j and broadcasts $M_i(d_{T_u}(v, u), EST_{T_v}(i, v))$ to its neighbors; otherwise, sensor i simply discards the flooding message from neighbor j.

C. Data Routing

A new routing structure is built after the data collection tree modification process is completed. If sensor i has data to transmit or helps other sensors to pass along data, it just transfers data to the neighbor indicated by H_i. Data are ensured to be handed over to the mobile users. The routing delays of those sensors are confined and controllable, and the mobile user can easily adjust the routing efficiency according to his/ her requirement.

Advantages:

1) The cost (time and communication) of building a data collection tree is significantly reduced.

2) The lightweight communication cost leads to less energy consumption of sensors.

3) The rapid updating process results in smooth routing transitions.

VI. EVALUATION

A. Investigation on system parameter λ

Fig. 4 demonstrates that region U formed by our approach is under control of λ. As λ increases, both node-count ratio (i.e. the ratio of the number of updated sensor nodes over the total number of sensor nodes in the network) and hop-count ratio (i.e. the ratio of the average hop distance of region U over the average hop distance of the entire network) decreases accordingly, i.e., the size of region U is reverse proportional to λ. The mobile user can thus control the system overhead through adjusting λ instantly. Fig. 3 also shows that our approach can efficiently update the routing tree, which is clearly indicated by the formation-time ratio (is the ratio of the formation time of region U over the formation time of a global optimal routing tree).

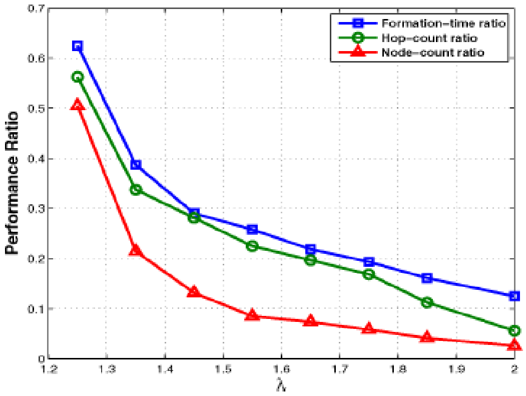


Fig. 4: Updating overhead versus

B. System Performance as the mobile user's movement

Fig. 5 stimulates the movement of mobile user. The mobile user moves from the left-bottom corner of the network, roughly following three-quarter of an ellipse inscribed in the field, to the middle point of the bottom line of the network.

Fig. 5.a., presents the system performance in two aspects. The three metric ratios are presented in the upper figure to show the affected area in updating the collection tree. The lower figure depicts the sensors path lengths in the data collection tree during the movement of mobile user.

The upper figure in Fig. 5.a depicts that a protocol migrates to build an optimal routing tree when the mobile user moves sufficiently far away (say 100 meters) from the original virtual sink. Whereas the lower figure shows that the routing efficiently approximately optimal as both average path length and longest path length get close to 1.

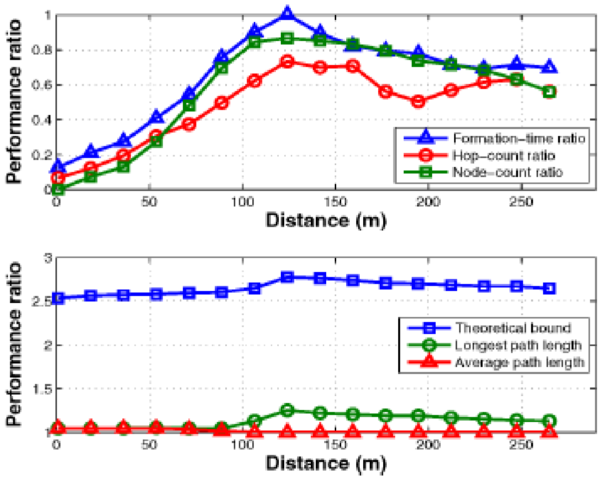


Fig. 5.a: System Performance versus Distance (Fixed λ)

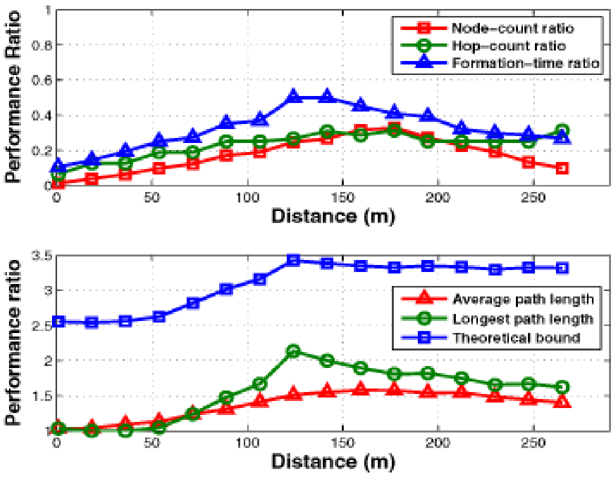


Fig. 5.b: System Performance versus Distance (Adjusted λ)

In Fig. 5.b, we adjust the λ value during the user's movement. Here, we apply a linear updating policy for λ:

$$\lambda_{v_k} \leftarrow \lambda_{v_{k-1}} + SNG * c * d_{T_{u(v_k,u)}}$$

where SNG = +1, if $d_{T_{u(v_k,u)}} \geq * \, d_{T_{u(v_{k-1},u)}}$; SNG=-1, otherwise. C is chosen from [0.1, 0.3] uniformly at random and v_k is the current virtual sink. From the upper figure in Fig. 5.b, we see that it ensures a low routing tree formation cost and a short construction time. As the lower figure shows that the delay distortion introduced by our approach is not excessively long.

C. CDF of delay performance

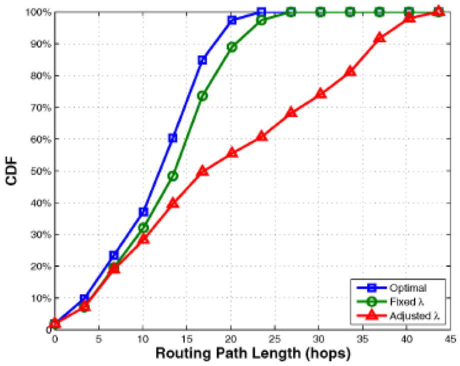


Fig. 6: CDF of routing path lengths

Fig. 6 indicates the CDF of sensor's routing path lengths. With the fixed λ approach, the CDF of sensor's routing path lengths is similar to that in the optimal routing tree. As with the adjusted λ approach, most of sensors still have shot routing paths and only a small number of sensors suffer a relatively long routing delay.

D. System performance versus network size

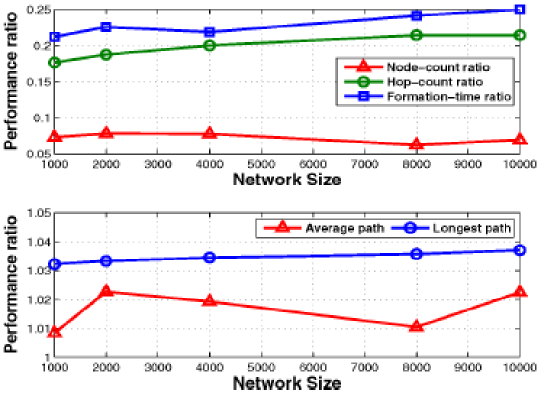


Fig. 7: System performance versus Network size

Fig. 7 demonstrates that our protocol has an effective scalability with regard to the network size variation from 1,000 to 10,000 and uses the adjusted λ policy.

VII. RESULTS

A. Simulations of Proposed Architecture

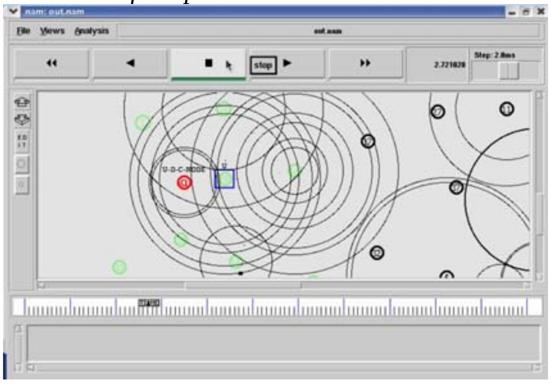


Fig. 8.a: Initial virtual sink U

The node 41 as shown in Fig. 8.a is mobile user which is accessing the data from the virtual sink U (node 29) in the sensor network during its motion. The dark big rectangle boxes indicate the dropped data packets and light dots indicate the transmitting data packets.

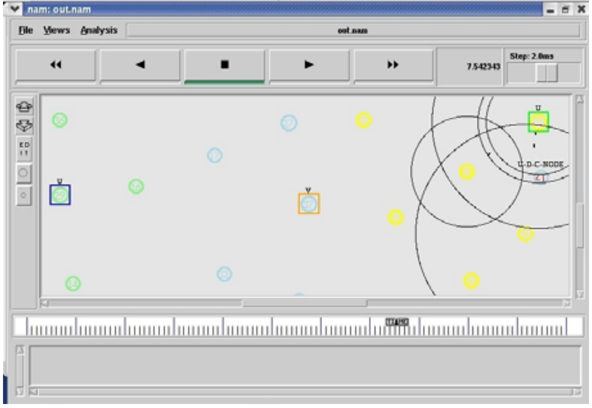


Fig. 8.b: New virtual sink V

In above fig. 8.b, nodes color green, blue and yellow denotes different area while the mobile node (41) is in movement. Fig. 8.b also depicts that the node 41 is in yellow region where the new virtual sink is yellow circled green box node u.

B. Performance Analysis (X Graphs)

In performance analysis, we plot the X Graphs for the performance comparison between existing (E in red line) and proposed (P in green line) system considering the 3 major parameters i.e., threshold, delay and energy consumption.

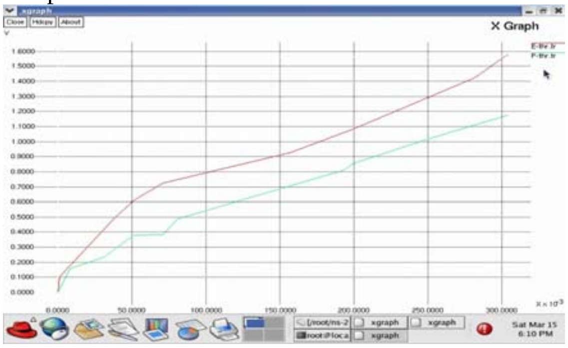


Fig. 9.a: Threshold X Graph

Fig. 9.a shows that the threshold for proposed system is less than the existing system.

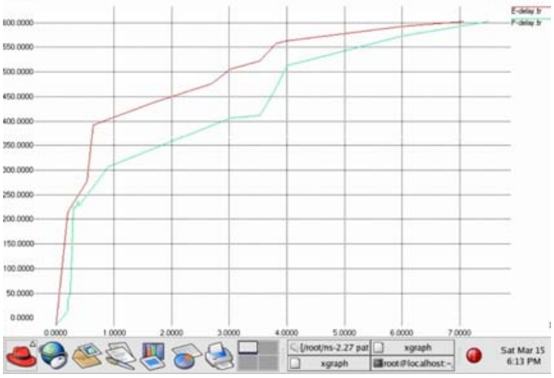


Fig. 9.b: Delay X Graph

Fig. 9.b depicts that the delay is very low for the proposed system as compared to existing system.

Fig. 9.c: Energy X Graph

Fig. 9.c reveals that the energy consumption is uniform in the proposed system, whereas it varies in the existing system.

VIII. CONCLUSION

In this study, we consider the data collection for mobile users in wireless sensor network to be enabled at anywhere and anytime. Here, we employed the spatial correlation to effectively construct the data collection trees dynamically according to the mobile user's movement. This approach reduces unnecessary packet losses and retransmissions due to disconnection with the mobile users.

Whenever the mobile user changes the location i.e. virtual sink, the existing data collection tree is modified locally and thus provides new data collection tree efficiently. Moreover, the proposed protocol is easy to carry out and the resulting routing performance on the new data collection tree is confined and assured with regard to the optimal value.

The cost of updating the routing structure (data collection tree) is significantly reduced, ensuring the low data collection delay and also provides the ability to acquire the real-time data for the mobile users. Extensive simulation results demonstrated that our protocol can improve the efficiency and scalability of our approach.

IX. FUTURE ENHANCEMENT

So far, we have taken the single user to introduce the basic design principle and curial properties of our proposed protocol. However, the solution in this paper would further be extended to support multiple users for collecting data simultaneously. To this end, in the system, multiple independent user data collection trees will be formed which would be distinguished based on their user IDs. User IDs are also labeled for flooding messages and sensory data for different mobile users. For each formed routing tree (denoted as T), the sensor (e.g., sensor i) maintains a corresponding $\mathbf{H}_i$(T). When the packet needs to be transmitted on the routing tree T, sensor i just forwards the packet to the neighbor notified by H_i(T).

ACKNOWLEDGEMENTS

This work has been partially supported by the NSFC Major Program 61190110, NSFC Grant No. 60970123, National High-Tech R&D Program of China (863) under grant No. 2011AA010100, Singapore MOE AcRF Tier 2 grant MOE2012-T2-1-070, and NAP M4080738.020. A preliminary version of this study has been presented in IEEE INFOCOM 2011 [17]. Any view, findings and conclusions expressed in this paper are those of the authors.

REFERENCES

[1] V. Dyo, S. A. Ellwood, D. W. Macdonald, A. Markham, C. Mascolo, B. Pásztor, S. Scellato, N. Trigoni, R.Wohlers, and K. Yousef, "Evolution and sustainability of a wildlife monitoring sensor network," in Proc. of Sensys, 2010, pp. 127–140.

[2] T. He, J. Stankovic, T. Abdelzaher, and C. Lu, "A Spatiotemporal Communication Protocol for Wireless Sensor Networks," IEEE Trans. Parallel and Distributed Systems, vol. 16, no. 10, pp. 995-1006, Oct. 2005.

[3] Y. Kim, T. Schmid, Z. M. Charbiwala, J. Friedman, and M. B. Srivastava, "Nawms: nonintrusive autonomous water monitoring system," in Proc. Of Sensys, 2008, pp. 309–322.

[4] I. Stojmenovic and X. Lin, "Loop-Free Hybrid Single-Path/ Flooding Routing Algorithms with Guaranteed Delivery for Wireless Networks," IEEE Trans. Parallel and Distributed Systems, vol. 12, no. 10, pp. 1023-1032, Oct. 2001.

[5] M. Mun, S. Reddy, K. Shilton, N. Yau, J. Burke, D. Estrin, M. Hansen, E. Howard, R. West, and P. Boda, "PEIR, the personal environmental impact report, as a platform for participatory sensing systems research," in Proc. Of MobiSys, 2009, pp. 55–68.

[6] L. Mo, Y. He, Y. Liu, J. Zhao, S.-J. Tang, X.-Y. Li, and G. Dai, "Canopy closure estimates with greenorbs: sustainable sensing in the forest," in Proc. of Sensys, 2009, pp. 99–112.

[7] U. Colesanti and S. Santini, "A performance evaluation of the collection tree protocol based on its implementation for the castalia wireless sensor networks simulator," month, 2010.

[8] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis, "Collection tree protocol," in Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems. ACM, 2009, pp. 1–14.

[9] R. Fonseca, O. Gnawali, K. Jamieson, S. Kim, P. Levis, and A. Woo, "The collection tree protocol (ctp)," TinyOS TEP, vol. 123, p. 2, 2006.

[10] O. Gnawali, R. Fonseca, K. Jamieson, and P. Levis, "Ctp: Robust and efficient collection through control and data plane integration," Technical report, The Stanford Information Networks Group (SING), 2008. available at: http://sing. stanford. edu/pubs/sing-08-02. pdf, Tech. Rep., 2008.

[11] M. Di Francesco, S. K. Das, and G. Anastasi, "Data collection in wireless sensor networks with mobile elements: A survey," ACM Trans. Sen. Netw., vol. 8, no. 1, pp. 7:1–7:31, Aug. 2011.

[12] S. Jain, R. C. Shah, W. Brunette, G. Borriello, and S. Roy, "Exploiting mobility for energy efficient data collection in wireless sensor networks," Mob. Netw. Appl., vol. 11, no. 3, pp. 327–339, Jun. 2006.

[13] R. Tan, G. Xing, J. Wang, and H. So, "Exploiting Reactive Mobility for Collaborative Target Detection in Wireless Sensor Networks," IEEE Trans. Mobile Computing, vol. 9, no. 3, pp. 317-332, Mar. 2010.

[14] G. Xing, T. Wang, Z. Xie, and W. Jia, "Rendezvous planning in wireless sensor networks with mobile elements," IEEE Trans. on Mobile Computing, vol. 7, no. 12, pp. 1430–1443, 2008.

[15] B. Kusy, H. Lee, M. Wicke, N. Milosavljevic, and L. Guibas, "Predictive QoS routing to mobile sinks in wireless sensor networks," in Proc. of IPSN, 2009, pp. 109–120.

[16] H. Lee, M. Wicke, B. Kusy, O. Gnawali, and L. Guibas, "Data stashing: energy-efficient information delivery to mobile sinks through trajectoryprediction," in Proc. of IPSN, 2010, pp. 291–302.latency balancing," in *Proc. Adv. Comput. Entertain. Technol.*, 2005, pp. 117–124.

[17] Z. Li, M. Li, J. Wang, and Z. Cao, "Ubiquitous Data Collection for Mobile Users in Wireless Sensor Networks," Proc. IEEE INFOCOM, pp. 2246-2254, 2011.