# IMPLEMENTATION OF TASK DISSEMINATION IN WIRELESS SENSOR NETWORKS USING MESH TOPOLOGY

[1]B.Maheswari
II$^{nd}$ M.E (CSE)
SNS College of Technology
Coimbatore, Tamilnadu, India

[2]Dr.S.Karthik
Professor & Head, Department of CSE
SNS College of Technology
Coimbatore, Tamilnadu, India

[3]Prof. T.Rajendran
Assistant Professor, Department of CSE
SNS College of Technology
Coimbatore, Tamilnadu, India

***Abstract*-- Wireless sensor networks, trend of the past few years involves in deploying a large number of small nodes. The nodes then sense environmental changes and report them to other nodes over flexible network architecture. A prototype was developed with a mesh topology of wireless sensor network for analysis of various message costs as well as the overall cost of data messages. Here a new task of task dissemination scheme is used, that can disseminate tasks into a subset of the sensor network. In which it uses a tree which is formed when the source node broadcasts the code version messages. The source node keeps the tasks and sends it based on the requests. This scheme also supports newly emerging wireless sensor network architectures with layered structure and mobile sensor nodes.**

***Keywords*- wireless sensor network, task dissemination, mobile sensor nodes.**

## I.INTRODUCTION

### *A Sensor Networks*

Sensor networks consist of very small nodes that are deployed in some geographical area. Each node is equipped with a sensor to perform monitoring, tracking, or surveillance and reports its finding to some central node. Most of the time the batteries in the nodes are not rechargeable, the networks operates as long as the power supply is O.K. when the power is off, the network ceases to operate. Thus low power is important in sensor networks.

Sensor Networks are used to measure temperature or pressure, or it could be used for target tracking or border surveillance. It could be also deployed in factories in order to monitor toxic or hazardous materials. It is also used to measure the weakness in building structures, or in vehicles and airplanes.

There are three different types of wireless networks are, cellular, ad-hoc, or sensor networks. Cellular networks, best exemplified by the cellular phones consist of mobile devices roaming an area that is divided into cells, with a base station located in every cell in order to serve the devices in that cell. The cell radius ranges from few kilometers (in old networks) to few tens of meters for modern networks. The mobile devices communicate by establishing a connection to the base station; all the base stations are connected to the phone network. The base station acts as a gateway to make and receive phone calls. Traditionally, the cellular networks use circuit-switching mode of operation. However, recently a movement towards packet switching is gaining acceptance.

Ad-hoc networks are networks that are deployed without an existing infrastructure. Mobile devices communicate among themselves by relaying the message over many devices. In this case, each mobile device works as a user and a routing switch at the same time. Usually, ad-hoc networks are networks that are established on a small geographical area in emergency situation. However, there are some proposals for wide area ad-hoc networks. Since both cellular and ad-hoc networks use mobile devices, low power circuits are very important. However, the mobile devices are rechargeable. Sensor nodes may not be rechargeable, the network works as long as the power supply is working, and then it ceases to work when the power supply is drained off.

## 2. IMPLEMENTATION OF WIRELESS SENSOR NETWORKS

### *B. Communication Networks*

#### *1. Network Topology*

The basic issue in communication networks is the transmission of messages to achieve a prescribed message throughput and Quality of

Service (QoS). QoS can be specified in terms of message delay, message due dates, bit error rates, packet loss, economic cost of transmission, transmission power, etc. Depending on QoS, the installation environment, economic considerations, and the application, one of several basic network topologies may be used. A communication network is composed of nodes, each of which has computing power and can transmit and receive messages over communication links, wireless or cabled.

***2. Fully connected networks***

It is a mesh network in which each of the nodes is connected to each other. A fully connected network doesn't need to use switching nor broadcasting.It suffers from problems of NP-complexity as additional nodes are added, the number of links increases exponentially. Therefore, for large networks, the routing problem is computationally intractable even with the availability of large amounts of computing power.

### *C. Routing Protocols for Mesh Networks*

Different routing protocols may impose different requirements on the design of their routing metrics. Hence, it is necessary to first understand what routing protocols best mesh networks to understand the necessary properties of routing metrics to support effective routing in mesh networks. Depending on when routes are calculated, the possible routing protocols for mesh networks can be divided into two categories: on-demand routing and proactive routing. Based on how packets are routed along the paths, proactive routing can further be divided into two subcategories: source routing and hop-by-hop routing. All of these different routing protocols have different costs in terms of message overhead and management complexity.

***1. On-demand Routing***

On-demand or reactive routing protocols only create a route between a pair of source and destination nodes when the source node actually needs to send packets to the destination. Network wide flooding is usually used to discover routes whey they are needed. For ad hoc networks, since there are frequent link breaks caused by the mobility of nodes, flooding-based route discovery provides high network connectivity and relatively low message overhead compared to proactive routing protocols. However, in mesh networks, links usually have much longer expected lifetimes due to the static nature of nodes.

Since the frequency of link breaks is much lower than the frequency of low arrivals in mesh networks, flooding-based route discovery is both redundant and very expensive in terms of control message overhead. Therefore, on-demand routing protocols are generally not scalable or appropriate for mesh networks.

***2. Proactive Routing***

In proactive routing protocols, each node maintains one or more tables containing routing information to every other node in the network. All nodes update these tables to maintain a consistent and up-to-date view of the network. When the network topology changes, the nodes propagate update messages throughout the network to maintain consistent and up-to-date routing information about the whole network. These routing protocols differ in the method by which packets are forwarded along routes.

- ***Source Routing:*** Source routing, imposes minimal burden on relaying nodes since the source node calculates the route for a low and puts the entire path of the low in the packet headers. Intermediate nodes only need to relay packets based on the paths in the packet headers. However, considering that the packet size in mesh networks is usually very small to cope with the high bit error rate of wireless channels, putting the entire path in the packet header imposes expensive message overhead.
- ***Hop-by-hop Routing:*** In hop-by-hop routing, every node maintains a routing table that indicates the next hops for the routes to all other nodes in the network. For a packet to reach its destination, it only needs to carry the destination address. Intermediate nodes forward the packet along its path based only on the destination address. Due to its simple forwarding scheme and low message overhead, hop-by-hop routing is dominant in wired networks. Similar reasons also make hop-by-hop routing the most preferable for mesh networks. However, despite its benefits, hop-by-hop routing requires careful design of its routing metrics to ensure loop-free packet forwarding. Due to the fact that hop-by-hop routing is most suitable for mesh networks, the requirements for designing routing metrics for hop-by-hop routing will be especially emphasized

### *D. Routing in WSNs*

Routing in wireless networks has been an active area of R&D for many years. Routing techniques rooted in computer data communications have been thoroughly explored for use in wireless networks, resulting in the emergence of many self-organizing, self-healing models in commercial implementations.

The reason for all this activity is that robust operation within changing propagation conditions and under energy and communication bandwidth constraints precludes the use of traditional IP-based protocols and creates a difficult challenge for dedicated WSN routing algorithms. The task of finding and maintaining routes in WSNs is nontrivial because energy restrictions and sudden changes in node status cause frequent and unpredictable changes.

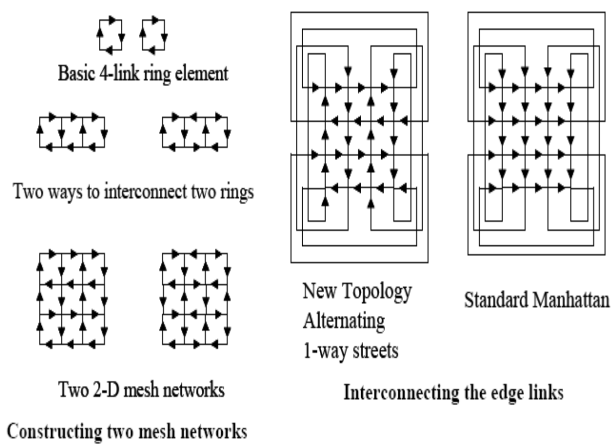


**Fig. 1 Construction of Mesh Networks**

Building and propagating automatic routing through the network requires powerful node processors, large amounts of memory, and additional dedicated routers, as well as network downtime until alternative routing is established. Routing tables with alternate routing while using low-cost, low-power processors proves to be a formidable challenge, which is amplified when the size and number of hops increase.

The resulting routing schemes take into consideration the inherent features of WSNs along with application and architecture requirements. To minimize energy consumption, routing techniques employ some interesting techniques special to WSNs, such as data aggregation and in-network processing, clustering, different node role assignment, and data-centric methods.

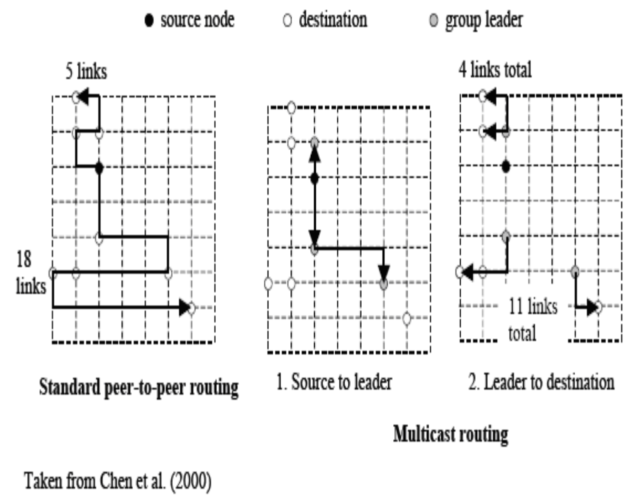


**Fig. 2 Multicast Routing improves efficiency and reduces message path length**

These routing techniques seek balance between simple solutions with limited robustness and sophisticated solutions. Even in sophisticated solutions, there is still the risk that in large networks or when messages are short, the routing overhead will consume valuable resources such as bandwidth and power and sometimes cause packet collisions. Worst case, these factors combine to finally degrade network robustness, throughput, and end-to-end delay.

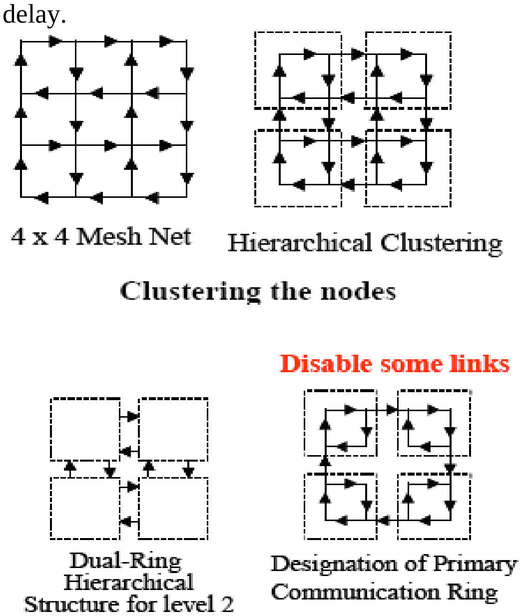


**Fig.3 Reducing Complexity**

One basic routing attribute related to the dynamic nature of an RF environment has yet to be solved: One moment after the routing table is created;

it is already obsolete because the RF conditions have changed.

## 3. DESIGN GOALS

The system is designed with a new application of task dissemination. It is designed to support task dissemination targeted at a subset of sensor nodes in both static and mobile WSNs. The source node initializes dissemination by broadcasting a code version message. The sink nodes form a group to receive the task image. A routing tree formed by the code version message and code request messages are used to route the code data. Each sensor node keeps a node and a request table reflecting the routing tree.

### *E. Task dissemination*

The dissemination time and message cost or different request intervals are shown. When the request interval is too small, the message cost is very high because too many request messages are sent and the probability of collisions and retransmissions of messages is high. This will increase the dissemination time. When the request interval increases, the message cost and the dissemination time will decrease. The increase in request interval will also increase the retransmission time for lost data messages, so after the dissemination time reaches the lowest value, it will start increasing.

## 4. SYSTEM FUNCTIONALITIES

The system includes the implementation of all the features of task dissemination. It also includes the analysis for different types of structures in WSNs and the analysis for mobile WSN with the mesh topology. Mesh has sets of servers that are managed together and participate in workload management. It enables enterprise applications to scale beyond the amount of throughput capable of being achieved with a single application server. Mesh also enables enterprise applications to be highly available because requests are automatically routed to the running servers in the event of a failure. The servers that are members of a mesh can be on different host machines.

The implementation is done with a prototype on TelosB nodes with a Stargate acting as the micro server. In this system the Experimentations were conducted to test the performance with an analysis of the energy consumption and a simulation for a grid structured static WSN. The performance results were measured by assuming a city area environment with moving sensors nodes.

The dissemination time with longer beacon interval, the repair time of broken links will increase, which will make the code dissemination time to increase. The dissemination time does not change much with the increasing of beacon interval when the sensors are static or the speed is low. For fast moving sensors, the increasing of beacon interval will increase the dissemination time sharply.

## 5. RELATED WORK

Multi-hop Over-the Air Programming (MOAP) [10] is a tasking application for TinyOS [12] WSNs. In MOAP, the source nodes broadcast advertisements to the neighboring nodes and broadcast the data based on the received requests from neighboring nodes. NACKs are used if the data messages are lost. Each receiving node maintains a sliding window for receiving code segments. The code images are disseminated neighbor by neighbor and finally reach the entire network.

A suppression protocol called Trickle [4] is used to reduce the number of broadcast messages. Trickle dynamically scales its suppression intervals to detect inconsistencies. Deluge uses Trickle to suppress the broadcasting of version advertisements. Maté [5][6] is a TinyOS application which uses application specific virtual machines (ASVMs) to reprogram WSNs. Maté stores tasks in capsules. The dissemination process is similar to Deluge[10]. The only difference is that after receiving a request message, only one packet randomly chosen from the requested capsule is broadcast with Maté while in Deluge, once a page is requested, the whole page is unicasted to the requesting node.

The wireless links in WSN usually have a low bandwidth and a high loss rate. Since the TinyOS system uses active messages, the traditional transport layer protocols such as TCP cannot be used in the TinyOS system. A reliable data transfer mechanism is needed for task dissemination in TinyOS. PSFQ is a reliable data transfer protocol designed for TinyOS system [3]. It is a hop-by-hop reliable transport protocol. PSFQ uses high rate, NACK-based error recovery which allows nodes to request missing packets from neighboring nodes aggressively. RMST(Reliable Multi-Segment Transport) [9] is a reliable transport layer protocol for TinyOS. It is used to disseminate large pieces of data to all subscribed nodes. RMST supports fragmentation and guaranteed delivery [5]. The receivers detect the loss and request missing packets from the source. Both PSFQ and RMST are transport layer protocols, in order to disseminate tasks, an application layer task dissemination protocol is still needed.

## 6. CONCLUSION

Wireless sensor networks (WSNs) provide low-cost, embedded sense-and-respond capability, and are therefore an integral part of the vision of pervasive computing. Most research on WSNs to date has focused on the development of efficient protocols for infrastructure establishment.

This work proposes a prototype with a Mesh topology of wireless sensor network for analysis of various message costs as well as the overall cost of data messages**.** A new application of task dissemination is designed to support task dissemination targeted at a subset of sensor nodes in both static and mobile WSNs. The source node initializes task dissemination by broadcasting a code version message. The sink nodes form an Multicast group to receive the task image. A routing tree formed by the code version message and code request messages are used to route the code data. Each sensor node keeps a node and a request table reflecting the routing tree have been implemented.

## REFERENCES

[1] A. Bakshi, A. Pathak, and V. K. Prasanna, “System-level support for macroprogramming of networked sensing applications," InternationalConf. Pervasive Systems Computing (PSC), June 2005.

[2] A. T. Campbell, S. B. Eisenman, N. D. Lane, E. Miluzzo, and R. A. Peterson, “People-centric urban sensing," in Proc. Proc. 2nd AnnualInternational Workshop Wireless Internet, New York, NY, USA, 2006, p. 18.

[3] Gummadi. R, O. Gnawali, and R. Govindan, “Macro-programming wireless sensor networks using kairos," in Proc. International Conf. Distributed Computing Sensor Systems (DCOSS), June 2005.

[4] Levis . P, N. Patel, D. Culler, and S. Shenker, “Trickle: a self-regulating algorithm for code propagation and maintenance in wireless sensor networks," in First USENIX/ ACM Symposium Networked Systems Design Implementation (NSDI 2004), San Francisco, CA, Mar. 2004.

[5] Levis. P, D. Gay, and D. Culler, “Active sensor networks," in SecondUSENIX/ACM Symp. Networked System Design Implementation (NSDI 2005), Boston, MA, May 2005.

[6] Levis. P, S. Madden, J. Polastre, R. Szewczyk, K. Whitehouse, A. Woo, D. Gay, J. Hill, M. Welsh, E. Brewer, and D. Culler, “Tinyos: an operating system for wireless sensor networks." Ambient Intelligence, Springer-Verlag, 2004.

[7] Parker. A, S. Reddy, S. Thomas, G. S. K. Chang, J. Burke, M. Hansen, M. Srivastava, D. Estrin, V. Pasxon, and M. Allman, “Network system challenges in selective sharing and verification for personal, social, and urban-scale sensing applications," in Proc. Fifth Workshop Hot Topics Networks (HotNets-V), HotNets-V, Nov. 2006.

[8] Stann. F and J. Heidemann, “Rmst: reliable data transport in sensor networks," in Proc. First International Workshop Sensor Net Protocols Applications, Anchorage, AK, USA, Apr. 2003, pp. 102-112. [Online]. Available: http://www.isi.edu/ johnh/PAPERS/Stann03a.html

[9] tathopoulos. T, J. Heidemann, and D. Estrin, “A remote code update mechanism for wireless sensor networks," University of California, Los Angeles, Center for Embedded Networked Computing, Tech. Rep. CENS-TR-30,Nov.2003.[Online].Available:http://www.isi.edu /johnh/PAPERS/Stathopoulos03b.html

[10] Xiao Zheng ,Behcet Sarikaya , “Task Dissemination with MDeluge in Sensor Networks”, IEEE Transactions on Wireless Communications, Vol.8, No.5, May 2009

AUTHORS PROFILE

B.Maheswari received her Bachelor of Engineering degree in Computer Science and Engineering in 2007 and currently she is pursuing her Master of Computer Science and Engineering degree in SNS College of Technology, Coimbatore, India.

Dr.S.Karthik is presently Professor & Head, Department of Computer Science & Engineering, SNS College of Technology, affiliated to Anna University- Coimbatore, Tamilnadu, India. He received the B.Sc degree from Bharathiar University in 1998 and M.Sc degree from the Periyar University in 2000, M.E Degree from Anna University Chennai and Ph.D degree from International University, Washington. His research interests include network security, web services and wireless systems. In particular, he is currently working in a research group developing new Internet security architectures and active defense systems against DDoS attacks. Dr.S.Karthik published many papers in international journal and conference papers and has been involved many international conferences as Technical Chair and tutorial presenter. He is a active member of IEEE, ISTE and Indian Computer Society.

T. Rajendran is a PhD Scholar in the Department of Computer Science and Engineering at Kongu Engineering College, affiliated to Anna University, Coimbatore, Tamilnadu, India. His research interest includes Web Services, Network Security and Web Technology. He has obtained ME degree in Computer Science and Engineering. He is a life member of ISTE. He has published more than 34 articles in International/ National Journals/Conferences.