İSTANBUL TECHNICAL UNIVERSITY ★ INSTITUTE OF SCIENCE AND TECHNOLOGY

POWER AWARE MANY TO MANY ROUTING PROTOCOL IN WIRELESS SENSOR AND ACTUATOR NETWORKS

M.Sc. Thesis by
Özlem EMİROĞLU
(514031103)

Date of submission: **6 July 2005**
Date of defence examination: **21 July 2005**

Supervisor (Chairman): **Prof. Dr. Emre HARMANCI**

Members of the Examining Committee: **Assoc. Prof. Dr. Erdal ÇAYIRCI**

Prof. Dr. Bülent ÖRENCİK (İ.T.Ü.)

JULY 2005

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

ALGILAYICI VE EYLEMCİ TELSİZ BİLGİSAYAR AĞLARINDA GÜÇ DENETİMLİ ÇOKTAN ÇOĞA YÖNLENDİRME PROTOKOLÜ

YÜKSEK LİSANS TEZİ

Özlem EMİROĞLU

(514031103)

Tezin Enstitüye Verildiği Tarih : 6 Temmuz 2005
Tezin Savunulduğu Tarih : 21 July 2005

Tez Danışmanı: **Prof. Dr. Emre HARMANCI**

Diğer Jüri Üyeleri **Assoc. Prof. Dr. Erdal ÇAYIRCI**

Prof. Dr. Bülent ÖRENCİK (İ.T.Ü.)

TEMMUZ 2005

PREFACE

I would like to express my deepest gratitude and appreciation to my advisors Assoc. Prof. Erdal ÇAYIRCI and Prof. Dr. Emre HARMANCI for their valuable guidance and contributions. I would also like to express my appreciation to Assoc. Prof. Erdal ÇAYIRCI for his unlimited patience, constant encouragement, vision, inspiration and endless source of knowledge and ideas throughout my study. Without him, the thesis would not have been possible.

I wish to thank to my husband and my parents for their love, encouragement, assistance, understanding and support. I am very grateful to them.

July, 2005 Özlem EMİROĞLU

CONTENTS

Page No

ABBREVIATIONS **vii**
LIST OF TABLES **viii**
LIST OF FIGURES **ix**
LIST OF ALGORITHMS **xi**
SUMMARY **xii**
ÖZET **xiii**

1. INTRODUCTION **1**
1.1 Contribution of the Thesis 2
1.2 Structure of the Thesis 3

2. BACKGROUND INFORMATION **5**
2.1 Physical Architecture of WSANs 6
2.2 Differences between WSNs and WSANs 7
2.3 Technical Challenges and Design Issues of WSNs and WSANs 8
2.3.1 Fault tolerance 9
2.3.2 Scalability / Network size 9
2.3.3 Node deployment 9
2.3.4 Production costs 10
2.3.5 Network topology 10
2.3.6 Node / Link heterogeneity 10
2.3.7 Transmission media / Communication modality 10
2.3.8 Connectivity 11
2.3.9 Coverage 11
2.3.10 Mobility 11
2.3.11 Power consumption 11
2.3.12 Quality of service / Real-time requirement 12
2.3.13 Data aggregation / Data fusion 12
2.3.14 Data reporting 12
2.3.15 Coordination 12
2.3.16 Data delivery 12
2.3.17 Network lifetime 13
2.4 Protocol Architecture of WSANs 13
2.4.1 Coordination plane 13

2.4.2 Management plane 14
2.4.3 Communication plane 14
2.4.3.1 Physical layer 14
2.4.3.2 Medium access layer 15
2.4.3.3 Routing layer 16
2.4.3.4 Transport layer 16
2.4.3.5 Application layer 17

3. RELATED WORK **18**
3.1 Routing Protocols in WSNs 18
3.1.1 Classification of routing protocols 18
3.1.2 Data centric routing protocols 19
3.1.2.1 SPIN (Sensor protocols for information via negotiation) 19
3.1.2.2 Directed diffusion 21
3.1.2.3 Rumor routing 23
3.1.2.4 Minimum cost forwarding algorithm (MCFA) 24
3.1.2.5 Gradient based routing (GBR) 25
3.1.2.6 Constrained anisotropic diffusion routing (CADR) and information driven sensor querying (IDSQ) 25
3.1.2.7 COUGAR 26
3.1.2.8 ACQUIRE 27
3.1.2.9 Energy aware routing 27
3.1.2.10 Routing protocols with random walks 28
3.1.3 Hierarchical based routing protocols 28
3.1.4 Location based routing protocols 29
3.2 Power Control in WSNs 29
3.2.1 Power control methods 29
3.2.2 Power control algorithms in routing layer 30
3.2.2.1 LMA and LMN 30
3.2.2.2 COMPOW 31
3.2.2.3 Energy efficient routing 31

4. PAMR, CPAMR AND PCPAMR IN WSAN **33**
4.1 Power Aware Many to Many Routing (PAMR) Protocol 33
4.1.1 Task registration process 36
4.1.2 Task deregistration process 39
4.1.3 Data dissemination 40
4.2 Centralized PAMR (CPAMR) 42
4.3 Power Controlled PAMR (PCPAMR) 42

5. PERFORMANCE EVALUATION **44**

6. EXPERIMENTAL RESULTS **48**

7. CONCLUSIONS AND FUTURE WORK **60**

REFERENCES **61**
BIOGRAPHY **64**

ABBREVIATIONS

WSN	: Wireless Sensor Networks
WSAN	: Wireless Sensor and Actuator Networks
PAMR	: Power Aware Many to Many Routing Protocol
PCPAMR	: Power Controlled PAMR
CPAMR	: Centralized PAMR
NS	: Network Simulator
NAM	: Network Animator

LIST OF TABLES

Page No

Table 4.1. The Registration Table of Node c in Figure 4.2 35
Table 4.2. The Routing Table of Node c in Figure 4.2 36
Table 4.3. The Registration Table of Node c in Figure 4.2 42
Table 6.1. Inputs Used in Our Simulations 48
Table 6.2. Sensitivity of PAMR for the Changes in Weighting Parameters. 57

LIST OF FIGURES

Page No

Figure 2.1. (a) Automated vs. (b) Semi-automated Architecture 2

Figure 2.2. The Physical Architecture of WSANs 6

Figure 2.3. The Components of (a) Sensors and (b) Actors 7

Figure 2.4. WSAN Protocol Stack 13

Figure 3.1. SPIN Protocol.. 20

Figure 3.2. Directed Diffusion.. 20

Figure 3.3. Rumor Routing 23

Figure 3.4. Query Plan at a Leader Node in COUGAR 27

Figure 4.1. The Packet Types in PAMR 34

Figure 4.2. Task Dissemination in PAMR 35

Figure 4.3. Echelons 36

Figure 4.4. The Total Power Used from Node X to Y 42

Figure 6.1. Average Data Packet Transmission Rate for Varying Network Size .. 49

Figure 6.2. Average Control Packet Transmission Rate for Varying Network Size. 49

Figure 6.3. Average Power Consumption Ratio for Varying Network Size.......... 50

Figure 6.4. Average End to End Delay for Varying Network Size........................ 50

Figure 6.5. Average Data Packet Transmission Rate for Varying Number of Actuators. 51

Figure 6.6. Average Control Packet Transmission Rate for Varying Number of Actuators. 51

Figure 6.7. Power Consumption Rate for Varying Number of Actuators.............. 52

Figure 6.8. Average End to End Delay for Varying Number of Actuators............ 52

Figure 6.9. Average Data Packet Transmission Rate for Varying Unsuccessful Delivery Rate.. 53

Figure 6.10. Average Control Packet Transmission Rate for Varying Unsuccessful Delivery Rate. 53

Figure 6.11. Power Consumption Ratio for Varying Unsuccessful Delivery Rate. 54

Figure 6.12. Average End to End Delay for Varying Unsuccessful Delivery Rate.54

Figure 6.13. Average Data Packet Transmission Rate for Varying Network Size and Different Weighting Parameter Values. 55

Figure 6.14. Average Control Packet Transmission Rate for Varying Network Size and Different Weighting Parameter Values. 55
Figure 6.15. Power Consumption Ratio for Varying Network Size and Different Weighting Parameter Values.. 56
Figure 6.16. Average End to End Delay for Varying Network Size and Different Weighting Parameter Values.. 56
Figure 6.17. Display of 30 Nodes in Network Animator 58
Figure 6.18. Display of 100 Nodes in Network Animator 59

LIST OF ALGORITHMS

Page No

Algorithm 4.1. Task Registration Algorithm 38

Algorithm 4.2. Data Dissemination Algorithm... 41

POWER AWARE MANY TO MANY ROUTING PROTOCOL IN WIRELESS SENSOR AND ACTUATOR NETWORKS

SUMMARY

In this thesis, a new power aware many-to-many routing protocol (PAMR) for wireless sensor and actuator networks is introduced. The protocol has two versions adapted for three cases. These protocols are PAMR, Centralized PAMR (CPAMR) and Power Controlled PAMR (PCPAMR). The first version is designed for networks where every node transmits at the same power level. The transmission power level may not be adjustable, or can be adjusted according to a deployment scenario and changed before or after the deployment. The second version is for the case where nodes can individually adjust the transmission power according to the channel conditions and communications distance.

The main objective of these protocols is conveying the sensed data from multiple sensor nodes to multiple actuators by a power efficient way. The number of packet transmission and end to end delay required for transmit packets to their destinations are minimized.

In the PAMR protocol, every sensor node has enough knowledge in its routing and registration tables to transmit the sensed data to the actuators. Each sensor node knows how to transmit the data to the related actuators through which neighbor node. If they have not this knowledge, they can obtain this from their neighbor nodes. A newly defined route selection function is used to send the packet from the best route. Min hop route is the most power efficient route in this protocol.

In the CPAMR protocol, transmission power of nodes is calculated in a centralized manner according to the average hop distance. Average hop distance can be found from node density and connectivity. By using this global value, a proper transmission range can be chosen and assigned to the whole sensor and actuator nodes. After calculating average hop distance between two nodes, a global transmission power is used in each sensor node communication.

In the PCPAMR protocol, the minimum hop route is not essentially the most power efficient route as in the first. It provides a tradeoff mechanism between the end-to-end delay and the total power used. The route selection function is extended to reach this goal. To determine the required power to transmit a packet, transmission powers of each node along the route are added to calculation of this function.

ALGILAYICI VE EYLEMCİ TELSİZ BİLGİSAYAR AĞLARINDA GÜÇ DENETİMLİ ÇOKTAN ÇOĞA YÖNLENDİRME PROTOKOLÜ

ÖZET

Bu tez çalışmasında, algılayıcı ve eylemci telsiz bilgisayar ağlarında yeni bir güç denetimli çoktan çoğa yönlendirme protokolü geliştirilmiştir. Bu protokolün üç duruma göre uyarlanmış iki versiyonu vardır. Bu protokoller PARM, merkezileştirilmiş PAMR (CPAMR) ve güç denetimli PAMR (PCPAMR) olarak adlandırılır. İlk versiyon, tüm algılayıcı düğümlerin aynı miktarda güç tüketimi ile haberleştiği telsiz bilgisayar ağları için tasarlanmıştır. Bu versiyonda, veri iletim güç düzeyi ayarlanamaz veya algılayıcı düğümlerin çevreye yayılım senaryosuna göre sürekli veya algılayıcı düğümlerin yayılımından önce ya da sonra ayarlanabilir. İkinci versiyonda algılayıcı düğümler kanal durumuna ve haberleşme uzaklığına göre iletişim güçlerini kendileri ayarlayabilirler.

Bu protokollerin esas amacı algılanan verinin algılayıcı düğümlerden eylemci düğümlere en az güç tüketimi ile iletilmesidir. Paket iletim sayısı ve paketlerin hedeflenen eylemci düğümlere iletilmesi için oluşan bir uçtan bir uca gecikme azaltılmıştır.

Birinci protokolde (PAMR), tüm algılayıcı düğümler yönlendirme ve kaydetme tablolarında algılanan verinin eylemci düğümlere iletilmesi için gerekli bilgiye sahiptirler. Her algılayıcı düğüm elindeki veriyi ilgili eylemci düğme hangi komşu düğüm üzerinden göndereceğini bilir. Eğer bu bilgi kendinde yoksa bunu komşu düğümlerinden elde edebilirler. Paketi en iyi yoldan göndermek için yeni tanımlanan yol seçme fonksiyonu kullanılır. Bu protokolde, en kısa yol güç tüketiminde en verimli yoldur.

İkinci protokolde (Merkezileştirilmiş PAMR), algılayıcı düğümlerin paket iletim güçleri ortalama sekme uzaklığına göre merkezi olarak hesaplanır. Ortalama sekme uzaklığı düğüm yoğunluğu ve düğümler arası bağlantıya bakılarak bulunur. Bu global değer kullanılarak her bir düğüm için paket iletim uzaklığı seçilir ve algılayıcı ve eylemci düğümlere atanır. İki düğüm arasındaki ortalama sekme uzaklığı hesaplandıktan sonra, algılayıcı düğümlerin iletişiminde global bir paket iletim gücü kullanılır.

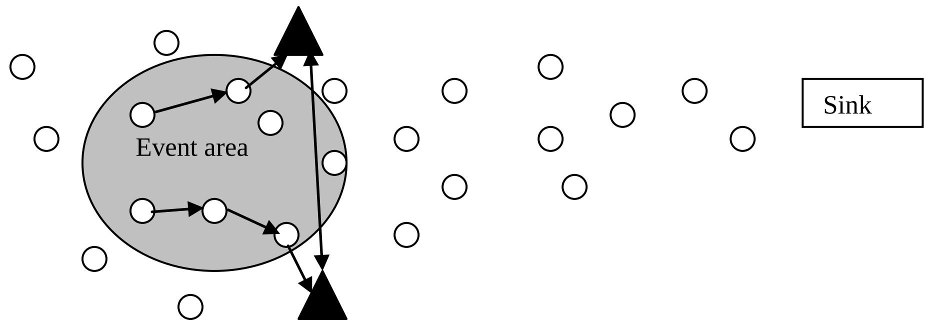
Üçüncü protokolde (Güç denetimli PAMR), PAMR'da olduğu gibi en az sekme sayısına sahip yol en çok güç tasarrufu sağlayan yol değildir. Bir uçtan bir uca gecikme ve toplam kullanılan güç azaltılır. Yol seçme fonksiyonu bu amaç için genişletilmiştir. Bir paketi iletmek için gerekli güç miktarını belirlemek için, kaynak düğümden gönderilecek düğüme kadar olan yol boyunca bulunan her bir düğümün iletim güçleri toplanarak bu fonksiyonun hesabına katılır.

1. INTRODUCTION

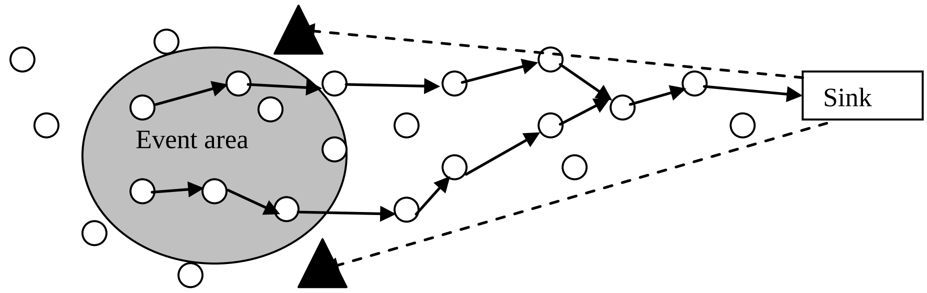
Wireless Sensor and Actuator Networks (WSANs) are distributed systems that adapt and react the ambient conditions reported through the collaborative effort of sensor nodes and actuators. In WSANs, a group of sensor nodes and actuators are distributed over a certain region with a wireless medium to perform distributed sensing and acting tasks and networking functionalities with a high degree of cooperation and coordination.

Sensor nodes have sensing, computation and wireless communication capabilities to collect information from the environment, process it based on the given task and command processed data to the appropriate actuators. To take decisions and execute actions based on the pre-specified user preferences, actuators have also acting capabilities and more hardware resources (especially more power and memory resources). They control several devices attached to them based on the sensed data obtained from sensor nodes. These actuators can be deployed inside the sensor field, and they may be collocated or dispersed.

The sensed data can be conveyed from sensor nodes to actuators either by an automated architecture or by a semi-automated architecture. In the automated architecture the sensed data are routed directly to actuators that stimulate the appropriate actions. In the semi automated architecture the sensed data are first routed to a sink the same as in the conventional many to one regime of wireless sensor networks (WSNs). Then the sink processes the gathered data and relays the fused data to the related actuators. These two architectures are given in Figure 2.1 [12]. One of these architectures is used depending on the application. Semi-automated architecture is similar to the architecture already used in WSN applications. To perform communication and coordination is easier than automated architecture.



(a)



(b)

Figure 2.1: (a) Automated vs. (b) Semi-automated Architecture [12]

We focus on the automated architecture in this thesis, which provides low latency and low energy consumption. Latency is an important performance metric for WSANs because they are generally real time systems that should re-act the detection of an event. Energy consumption is another key metric for WSANs and it is important to prolong the lifetime of the network. Each sensor node operates on limited battery energy consumed mostly in transmission and reception of data.

1.1 Contribution of the Thesis

Our contribution in this thesis is the design and evaluation of a new power aware, many to many, data centric routing protocol PAMR and its extensions Centralized PAMR (CPAMR) and Power Controlled PAMR (PCPAMR) for wireless sensor and actuator networks (WSANs). In PAMR, every node can transmit at the same power level. PAMR can also be used for the case where the transmission power of the nodes can be controlled but every node transmits still at the same power level that is centrally determined. The most energy efficient protocol among these new protocols is PCPAMR in which power is controlled by adjusting transmission powers of each node according to the channel conditions and communications distance.

We evaluate the energy efficiency and latency of PAMR, CPAMR and PCPAMR analytically and experimentally. We analyze energy savings and delay minimizations for our three protocol and directed diffusion. Our analytical framework finds out the appropriate communications distance for a given average connectivity, i.e., the average number of nodes that can receive the transmissions of a node, and node density. Our results suggest that the lifetime of the network in PAMR is higher than in directed diffusion and our scheme is more scalable and has high performance gains comparing to directed diffusion. End to end delay which is an important quality of service (QoS) metric in many WSANs applications is also minimized with our solutions while directed diffusion give less importance to delay.

To verify and complement our analytic evaluation, we implement PAMR in the ns-2 simulator. Particularly, we compare the performance of PAMR against directed diffusion and study the sensitivity of PAMR performance to the choice of parameters. We validate results with an actual implementation of our application.

Other contributions in this dissertation include evaluation metrics, tradeoffs, simulation platforms, test suites, an application, requirements, challenges, and insights into the design of data dissemination systems for wireless sensor networks. Specifically, our evaluation metrics are average dissipated energy, average delay, and distinct event delivery ratio. These metrics indicate the overall lifetime of sensor nodes, the temporal accuracy of the estimates, and the robustness of the system. The challenge is to design a system that is long-lived but still accurate and robust.

1.2 Structure of the Thesis

In Section 2, background information is given to understand the environment that new routing protocol will be used in and to determine the technical challenges and design issues.

Related work is studied in two subsections in Section 3. First, current routing protocols for WSNs are summarized. Their deficiencies are described according to our perspective. Then power control techniques for WSNs in literature are surveyed and criticized.

In Section 4, the newly proposed routing protocol PAMR is described in detail. After explaining the general architecture, task registration and deregistration, and data dissemination of PAMR is explained. In the last part of the section, implementation of PAMR protocol is discussed. The figures and the pseudo code explaining our new algorithm for PAMR are given.

The analytical models for the performance evaluation of the proposed scheme are provided in Section 5. Mathematical models for calculating the average hop distance between two sensor nodes from node density and connectivity are developed under different deployment conditions. The uplink node selection functions for PAMR and PCPAMR and weighting parameters to evaluate them are also formulized to determine the analytical bounds.

Simulation results are presented in Section 6, which verify the mathematical models introduced in Section 5. The improvements of PAMR and PCPAMR on the number of transmitted data and control packets, power consumption ratio and end to end delay of the overall network are evaluated. The results are interpreted for changing three weighting parameters, deployment conditions and other system factors.

Finally, Section 7 concludes the thesis by grouping future projections.

2. BACKGROUND INFORMATION

Modern research on wireless sensor networks (WSNs) are started around 1980 with the Distributed Sensor networks (DSN) program at the Defense Advanced Research Projects Agency (DARPA). Technology components for a DSN were identified in a distributed Sensor Nets workshop in 1978. These include sensors, communication, processing techniques and algorithms and distributed software. In the 1980s, planners of military systems quickly recognized the benefits of the sensor networks, which become a crucial component of network-centric warfare [2]. In platform-centric warfare, sensors and weapons are mounted with and controlled by separate platforms that operate independently. In network centric warfare, sensors do not necessarily belong to weapons or platforms. Instead, they collaborate with each other over a communication network, and info is sent to the appropriate shooters. An example of network centric warfare is Cooperative Engagement Capability (CEC) [3] developed by the U.S. Navy. Other examples to military sensor networks are FDS (Fixed Distributed System), ADS (Advanced Deployable System), UGS (Unattended Ground Sensors), REMBASS (Remote Battlefield Sensor system), and TRSS(Tactical Remote Sensor System).

DARPA Sensor Information Techno (SensIT) program [4] pursued two key research and development thrusts. First, the program develops networking techniques suitable for highly dynamic ad hoc environments. Second was networking information processing (how to extract useful, reliable, and timely information from the deployed sensor networks).

In Tactical Automated Security System (TASS) [5], which is currently in use SensIt networks have new capabilities. The networks are interactive and programmable with dynamic tasking and querying. The software and the overall system design support low latency, energy-efficient operation, built-in autonomy and survivability and low probability of detection of operation.

In today’s wireless sensor networks; sensors, processors and communication devices are much smaller and cheaper. Ember, Crossbow and Sensoria are now building and deploying small sensor nodes and systems. Small and inexpensive sensors based upon micro-electromechanical system (MEMS) technology, wireless networking and inexpensive low-power processors allow the deployment of wireless ad-hoc networks.

Sensing, computing and communications can now be performed on a single chip, further reducing the cost and allowing deployment in ever larger numbers. In the future, sensors will be more capable and versatile. For example, SmartDust [6] research project at the University of California, Berkeley is building MEMS sensors that can sense and communicate and they are very tiny [7].

2.1 Physical Architecture of WSANs

In WSANs, sensor nodes and actuators are scattered in the sensor/actor field while the sink which monitors the overall network and communicates with the task manager node and sensor/actor nodes, if necessary is separated from the sensor/actor field. The physical architecture of WSANs is given in Figure 2.2 [12].

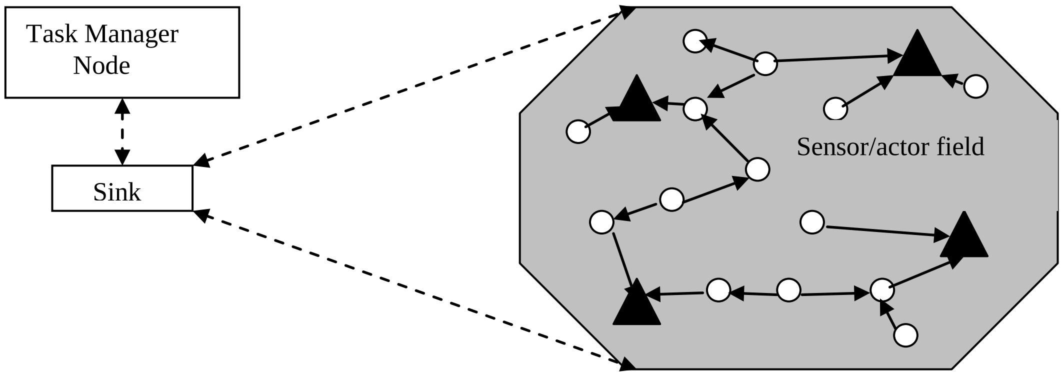
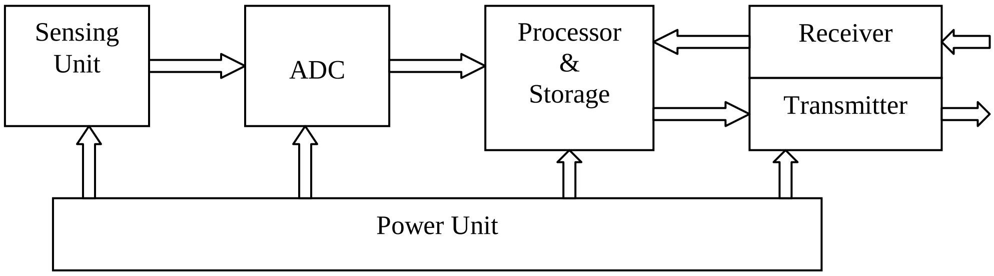


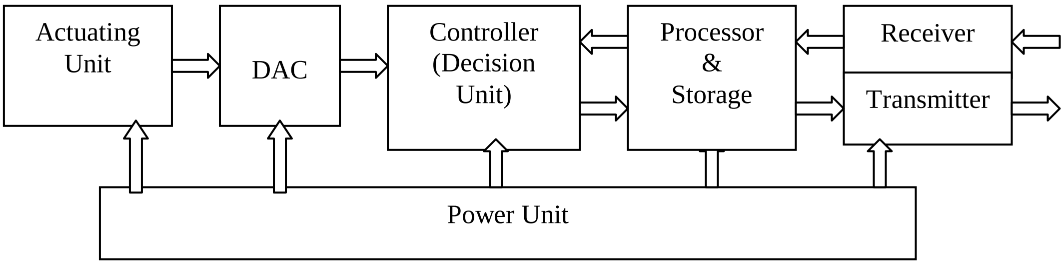
Figure 2.2: The Physical Architecture of WSANs [12]

Instead of initiating an action, actor node may communicate with other actor nodes and send messages come from sensor nodes to other actors in the network. The type of these messages depends on the coordination of actor nodes. It may be the whole event information or a part of the event information, or a specific command.

The components of sensor and actor nodes used in WSAN applications can be seen in Figure 2.3(a) and (b), respectively [12].



(a)



(b)

Figure 2.3: The Components of (a) Sensors and (b) Actors [12]

Each sensor node consists of four components: sensing unit, analog digital converter (ADC), processor and storage unit, communication (receiver and transmitter) unit. They may also have additional application-dependent components such as a location-finding system, power generator and mobilizer. The sensing unit observes phenomena and returns analog data to ADC which converts it to digital form. CPU processes the received data, makes computation about routing, controls and monitors power. Communication unit (transceiver unit) receives the command from CPU and transmits the data to the network.

Inside an actor node there is a controller unit that takes sensor readings as input and generates action commands as output. After the digital data is processed, it reaches the controller unit where the decision is made and the action command is constructed. This digital command is converted to an analog signal by digital to analog converter (DAC). Lastly, this analog signal is transformed into an action via actuation unit.

2.2 Differences between WSNs and WSANs

Wireless Sensor Networks (WSNs) consist of sensor nodes scattered in a sensor field with sensing, computation, and wireless communications capabilities. Sensor nodes coordinate among themselves and collect and route data either to other sensor nodes or back to an external sink(s). A sink may be a fixed or mobile node capable of connecting the sensor network to an existing communications infrastructure or to the internet where user can have access to the reported data.

There are common factors and constraints influencing the network design for WSANs and WSNs [8] like power consumption, scalability, mobility, bandwidth, and capacity. However, WSANs, especially the automated architecture, differs from WSNs mainly due to the following issues:

- Many-to-many relation among the nodes: Actuator nodes can act according to the sensed data coming from a number of sensor nodes, and a sensor node can send data to multiple actuators.
- End-to-end delay constraint: In WSNs the power consumption is the primary concern. However, in some WSAN applications end-to-end delay may be the most prominent design factor.
- Node heterogeneity: There are actuator and sensor nodes in WSANs. This introduces greater node heterogeneity comparing to WSNs. Actuators consume more energy than sensors. Therefore they are equipped with better resources.

The most significant difference between automated WSANs and WSNs is related to topology. More specifically, WSANs need power aware, fault tolerant, self organizing communication protocols tailored to many-to-many relation between sensor and actuator nodes, which is different from the many to one regime of WSNs. In conventional sensor networks, the sensed data is collected by a single node, which is often called as sink [8]. Instead, the sensed data is forwarded to multiple actuators in the automated WSANs where one actuator may fetch data from multiple sensors, and one sensor node may feed multiple actuators. Therefore, new power aware many-to-many data centric routing protocols are required for WSANs where multicast communication is very important because an efficient implementation of multicasting permits much better use of the available power by reducing the number of packets transmitted for the same sensed data [8, 9].

2.3 Technical Challenges and Design Issues of WSNs and WSANs

Depending on the actual needs of the application, different architectures and design goals/constraints have been considered for sensor networks. Since the performance of a routing protocol is closely related to the architectural model, in this section we mention many challenges and design issues in WSNs and WSANs [8, 10, 11, 12].

2.3.1 Fault tolerance

When some of sensor nodes or actuators fail, it should not affect the overall task of the sensor network. Sensor networks should continue functioning without any interruption due to sensor node failures. MAC or routing protocols must provide new links and routes. This may require adjusting transmit powers and signaling rates on the existing links to reduce energy consumption or rerouting packets where more energy is available.

2.3.2 Scalability / Network size

This size of a sensor network can very from one to thousands of nodes. The number of nodes participating in a sensor network is mainly determined by requirements relating to network connectivity and coverage and by the size of the area of interest.

2.3.3 Node deployment

Sensor nodes are densely deployed either very close or directly inside the phenomenon to be observed. The deployment of sensor nodes in the physical environment can be either manual or randomized. In manual deployment, the sensors are manually placed and data is routed through predetermined paths. However, in random deployment, the sensor nodes are scattered randomly. Deployment may be a one-time activity or continuous process. The type of deployment affects node density, node locations and degree of network dynamics.

However, such a dense deployment is not necessary for actor nodes due to the different coverage requirements and physical interaction methods of acting task. In WSAN, the number of actors is much less than that of sensors. There must be an actor deployment such that at least one actor is able to act on every point of the environment. In the deployment scenario of actor nodes, it must be considered that actors should be coordinate with each other to achieve given task and they can be act on the same area concurrently. Tradeoff between connectivity and acting coverage is important.

2.3.4 Production costs

For the deployment of sensor nodes in large numbers, a sensor node and actuator should be inexpensive. Limited size and cost limits energy available as well as computing, storage and communication resources.

2.3.5 Network topology

A sensor network forms a single-hop network, in which every sensor node able to directly communicate with every other node. In a multihop network, there are number of hops between any two nodes. Topology affects latency, robustness and capacity. There are 3 phases in topology maintenance and changes:

- Pre-deployment and deployment phase: Sensor nodes are placed in the sensor field.

- Post-deployment phase: After deployment, sensor nodes' position can be changed due to available energy, task details, etc. This causes topology change.

- Redeployment of additional nodes phase: Additional sensor nodes can be deployed in the sensor filed.

2.3.6 Node / Link heterogeneity

Depending on the application, a sensor node can have a different role or capability. Nodes may differ in the type and number of attached sensors. Special sensors can be deployed independently or the different functionalities can be included in the same sensor nodes. More powerful nodes may collect, process, route sensory data from many more limited sensing nodes. Therefore, they can be chosen as actors. Actors makes more complicated and energy consuming activity than sensor nodes. Actors are resource rich nodes equipped with better processing capabilities, stronger transmission powers and longer battery life. Degree of heterogeneity affects the complexity of the software executed on the sensor nodes and management of whole system.

2.3.7 Transmission media / Communication modality

Communication links of sensor nodes and actuators can be formed by radio, infrared, optical media, diffuse light, inductive and capacity coupling or even sound. Both infrared, light and optical media require a line of sight between the sender and receiver. Radio waves do not require a free line of sight. Light beams allow for much smaller and more energy-efficient transceivers than radio waves. Inductive and capacity work over small distances. Sound is generally used for communication under water. Related to the transmission media is design issue of MAC.

2.3.8 Connectivity

Connectivity depends on the communication ranges and physical locations of individual sensor nodes and actuators. If there is always a network connection between any two nodes, the network is said to be connected. Connectivity may be decrease due to node failures. In addition, if nodes enter the communication range of other nodes, it affects the connectivity between them.

2.3.9 Coverage

Coverage area of sensor node and actuator is determined by the effective range of sensors attached to it. Network coverage means the degree of coverage area of interest by nodes. The area of interest can be completely or partially covered by nodes. The degree of coverage influences information processing algorithms. Network lifetime may be extended by switching redundant nodes to sleep modes when high coverage exists.

2.3.10 Mobility

In many applications, both the sensor nodes and actuators can be mobile. Routing messages is more challenging when mobile nodes are used. Route and topology stability, control of energy and bandwidth become important issues. Also phenomenon can be mobile. Dynamic events require periodic reporting to actors.

Mobility of a sensor node and actuator can be interchangeable. Nodes can be mobile or fixed depending on remoteness of the phenomenon. When they are mobile, they move to interesting physical locations. Mobility influences the design of networking protocols and distributed algorithms.

2.3.11 Power consumption

Sensor node and actuator lifetime is depending on battery lifetime. Performing computations, transmitting information and sensing the environment are three domain causes of power consumption of nodes. Power failure of nodes may require topology changes, rerouting of packets and reorganization of the network.

2.3.12 Quality of service / Real-time requirement

For time constraint applications (real time applications) bounded latency for data delivery is another challenge. As energy is reduced the network may be give loss importance for quality than lifetime of network. In WSANs, there may be a need to rapidly respond to sensor input. Sensor data must be valid at the time of acting. Time between the sensing occurs and acting started should be minimized in order to perform right action.

2.3.13 Data aggregation / Data fusion

Data aggregation is combination of data from different sources to reduce the number of transmissions. This technique is used to achieve energy efficiency and data transfer optimization in number of routing protocols. Data aggregation is also feasible through signal processing techniques. In that case, it is referred as data fusion where a node is capable of producing a more accurate signal by reducing the noise and using some techniques such as beamforming to combine the signals.

2.3.14 Data reporting

Data reporting can be categorized as time-driven, event-driven or hybrid of all these methods. In the time-driven delivery methods, sensor nodes or actuators periodically switch on their sensors and transmit the data of interests. In event-driven and query-driven methods, nodes react sudden changes in the value of sensed attribute or respond to a query generated by the actuators or another node. Data reporting methods influences energy consumption and route calculation.

2.3.15 Coordination

While in WSNs, the main communication problem is between sensor nodes and the sink, in WSANs, sensor-actor and actor-actor communications are the main problems. In order to provide effective sensing and acting, a distributed local coordination mechanism is necessary among sensor nodes and actuators.

2.3.16 Data delivery

The data delivery model to the sink or actuator can be continuous, event-driven, query-driven and hybrid depending on the application. In the continuous delivery model, each sensor node sends data periodically. In event-driven and query-driven models, the transmission of data is triggered when an event occurs or a query is generated by the sink. To minimize energy consumption and stabile route, data delivery model should be chosen attentively.

2.3.17 Network lifetime

The necessary lifetime has a high impact on the required degree of energy efficiency and robustness of the nodes. Sensor nodes within one hop from the actuators may have a higher load of relaying packets and more likely to fail than other nodes because of consuming more energy than others. If for each event, different actuators are triggered, this implies that relaying sensor nodes is different for each event. This causes fairness between nodes and longer lifetime of network.

2.4 Protocol Architecture of WSANs

The protocol stack in WSANs consists of physical layer, medium access layer, routing layer, transport layer and application layer and three planes which are management, communication and coordination planes given in Figure 2.4 [12]. These planes help the sensor nodes and actuators to coordinate the sensing and acting task respectively and lower overall power consumption.

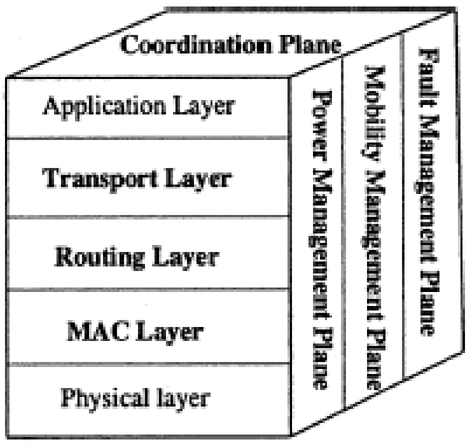


Figure 2.4: WSAN Protocol Stack [12]

2.4.1 Coordination plane

It uses coordination and negotiation techniques to make correct and on-time actions. Node behavior is determined according to the data received from communication plane and management plane. After sensing an event, sensors communicate with each other to share their readings. The coordination plane makes decisions about exchanged data. To determine nodes which will not transmit data, to perform multihop routing and data aggregation and to select actor(s) to which sensor data will be transmitted are done by the coordination of sensor nodes and actuators.

2.4.2 Management plane

It monitors and controls sensor nodes and actuators so that it operates properly. Also it can provide information to the coordination plane to make decisions. The functions performed by the management layer can be categorized into the following three areas [12]:

- Power management plane manages how a node uses its power. For example, when the power level of a node is low, this plane informs the coordination plane so that the node will not participate in sensing, relaying, or acting activities.
- Mobility management plane detects and registers the movements of nodes so that network connectivity is always maintained.
- Fault management plane refers to the detection and resolution of node problems. For example, when the sensitivity of sensing unit or the accuracy of the actuation unit degrades, fault management plane informs the coordination plane about this situation.

2.4.3 Communication plane

It performs information exchanging among the sensor nodes and actuators and produces a change in the state of the network. It receives commands from coordination plane and provides the link relation between nodes. Specially, the communication plane deals with the construction of physical channels, the access of the node into the medium (MAC), the selection of routing paths through which the node transmits its data and the transport of packets from one node to another. The layers in communication plane are explained in the following subsections [8, 12].

2.4.3.1 Physical layer

Frequency selection, carrier frequency generation, signal detection, data encryption, modulation, transmission and receiving techniques are addressed in this layer. Minimum required power to transmit a signal over distance d is proportional to d^n, where $2 <= n < 4$. Power starts to drop off with higher exponents at smaller distances for low antenna heights. There are also shadowing and path loss effects in multihop sensor networks. Propagation losses and higher node density may affect data reliability. In addition, energy efficient physical layer design is an important issue in WSANs.

2.4.3.2 Medium access layer

To organize large number of sensor nodes, there is a need for MAC protocol. MAC layer is responsible for creation of the network infrastructure (establishing communication links for data transfer) and fair and efficient communication between sensor nodes. MAC protocol should also maintain network connectivity between sensor nodes and actuators. Timely detection, processing and delivery of information are the main requirements of real time applications. The main goal in designing a MAC protocol for WSANs is to minimize energy consumption while limiting latency and loss of data throughput. MAC protocol has to support QoS issues such as efficient bandwidth utilization. Bandwidth should be reallocated efficiently as sensors' data rates change and sensor nodes move. In addition, handling of mobility is needed to be examined for WSANs while designing MAC layer.

Contention base protocols are not suitable for real-time applications. Handshaking in contention-based channels increases the latency of the data. SMAC decreases energy consumption by using a random wake-up schedule during connection phase and by turning the radio off during idle time slots. But it should also consider QoS issues. In SMAC, neighbor discovery and channel assignment phases are combined. A communication link consists of a pair of time slots with fixed frequency. There is no need for network wide synchronization, although communication neighbors in a subnet need to be time-synchronized.

In TDMA approaches, TDMA time scheduling is a problem in WSANs. In hybrid TDMA/FDMA, there is a centralized frequency and time division. Optimum number of channels calculated for minimum system energy. CSMA based approaches has listening mechanism and back-off scheme. Constant listening time is used for energy efficiency.

Collision-free protocols can be suitable for WSANs. They reduce the delay (QoS issue) and provide real-time guarantee. They also save power by eliminating collisions. The only problem is use of multiple channels. Also the complexity of the protocol is another concern.

2.4.3.3 Routing layer

In WSANs, multiple actors cause challenges in terms of routing solutions. First selecting an actor node to send source data with a power efficient way is one of the challenges for a source sensor node. While the source data is transmitted through relaying sensors towards to an actor node, it may be aggregated or forwarded in order to achieve high efficiency.

Data aggregation is important because it reduces the communications overhead. Congestion in the network can be prevented by aggregating the data coming from multiple sensors at one sensor. The types of data aggregation are temporal or spatial, snapshot or periodical, centralized or distributed early or late data aggregations. In data aggregation approach, there is a problem deciding the location of data aggregation. It depends on the application requirements. Network lifetime and reliability needs of applications determine the choice. Also the use of data aggregation reduces congestion, redundant data and energy consumption. There is also a need for congestion control mechanism when two sensor nodes transmit their packets through different paths to the same sensor node. Congestion can also be prevented by data aggregation algorithms.

In addition to determining the path selection and data delivery, routing protocol should support real-time communication and power efficiency. Moreover, the routing protocol should also consider the issue of prioritization and should provide data with low delay bounds to reach the actor on time. Routing protocols also should deal with reliable event transmission. One of the ways of decreasing the delivery failure may be to provide the data of sensors in each cluster to flow through different paths.

2.4.3.4 Transport layer

The transport protocols must support conventional reliability and realtime requirements in WSANs. When the transport protocol for sensor-actor communication detects low reliability, transport protocol for actor–actor communication regulates the traffic between actors so that the actor receiving low reliable event information can inform the other nearby actors about this situation as soon as possible. Since sensor-actor and actor–actor communications occur consecutively in WSANs, a unified transport protocol is needed which works well for both cases.

The main goal of WSANs is the detection of specified events of interest. The same event is reported to actuators by multiple sensor nodes to decrease the loss of data packet. Event may be lost due to environmental characteristics of sensor fields and power limitations of the nodes. It may be also lost although multiple sensor nodes report it. End-to-end reliable event transfer schemes are needed to overcome this problem.

2.4.3.5 Application layer

Three known application layer protocols are SMP (sensor management protocol), TADAP (task assignment and data advertisement protocol) and SQDDP (sensor query and data dissemination protocol). SMP performs some management tasks such as moving sensor nodes, exchanging data related to the location finding algorithms, querying the senor data related to the location finding algorithms, querying the sensor data related to the location finding algorithms, querying the sensor network configuration and the status of nodes, reconfiguring the sensor network.

SQDDP responds to queries and collect incoming replies. Main topic of SQDDP protocol is how to query available data on nods. Attribute or location based naming is preferred for queries. TADAP provides the user software with efficient interfaces for interest dissemination. Users send their interest to sensor nodes or sensor nodes advertise the available data to the users and the user query the data in which they are interested.

3. RELATED WORK

In this part of the thesis, previous works about routing protocols and power control algorithms which were carried out by other researchers are presented for comparison with PAMR and its extensions CPAMR and PCPAMR.

3.1 Routing Protocols in WSNs

In this section, we classify the routing protocols for WSNs first and then analyze the existing data centric routing protocols and give advantages, disadvantages and drawbacks of these protocols [11, 13, 14]. This analysis helps in identifying open issues in the area of routing in WSNs. Because of our routing protocol PAMR is in the class of data-centric protocols, only data centric protocols are analyzed to compare.

3.1.1 Classification of routing protocols

Depending on the network structure, routing protocols in WSNs can be divided into 3 categories:

- Data centric routing (flat based routing): all nodes in the network have same roles and functionalities. They are query based and depend on the naming of desired data. Naming helps elimination of redundant transmissions.

- Hierarchical based routing (clustering routing): nodes have different roles in the network. Clustering the nodes so that cluster heads can do some aggregation and reduction of data in order to save energy.

- Location based routing: location of nodes is important to make routing decisions. It utilizes the position information to relay the data to the desired regions rather than the whole network.

Routing protocols can also be classified depending on how the source finds a route to the destination.

- Proactive protocols: All routes are computed before they are really needed.

- Reactive protocols: Routes are computed on demand.

- Hybrid protocols: It uses a combination of these two ideas.

- Cooperative routing protocols: Nodes send data to the central node where data can be aggregated and processed if necessary.

Another classification type is based on the protocol operation such as multipath based, query based, negotiation based, QoS based, coherent based, energy aware, timing based routing protocols.

In the rest of this section, we present network structure based routing protocols.

3.1.2 Data centric routing protocols

Data centric routing is different from traditional address-based routing where routes are created between addressable nodes. Sink sends queries to certain regions and waits for data from their sensors located in the selected regions. Attribute-based naming is used specify the properties of data. A set of sensor nodes are selected to route a data for a specific task and data aggregation method is utilized during the relaying of data in order to eliminate redundant data and save energy. SPIN, Directed Diffusion, Rumor Routing, GBR, MCFA, CADR, COUGAR, ACQUIRE, EAR are examples of flat routing protocols.

3.1.2.1 SPIN (Sensor protocols for information via negotiation)

SPIN [15] is based on advertisement of data available in sensor nodes. SPIN family of protocols uses data negotiation and resource adaptive algorithms. Metadata negotiations are performed before any data is transmitted. Each node upon receiving new data, advertises it to its neighbors by broadcasting advertisement packet. Advertisement packets contain metadata. The nodes interested in this data reply back by a request packet. Data packets are only sent to sensor nodes which send request packets. Hence the data is delivered to every node that may have not an interest. Figure 3.1 [13] summarizes the SPIN protocol.

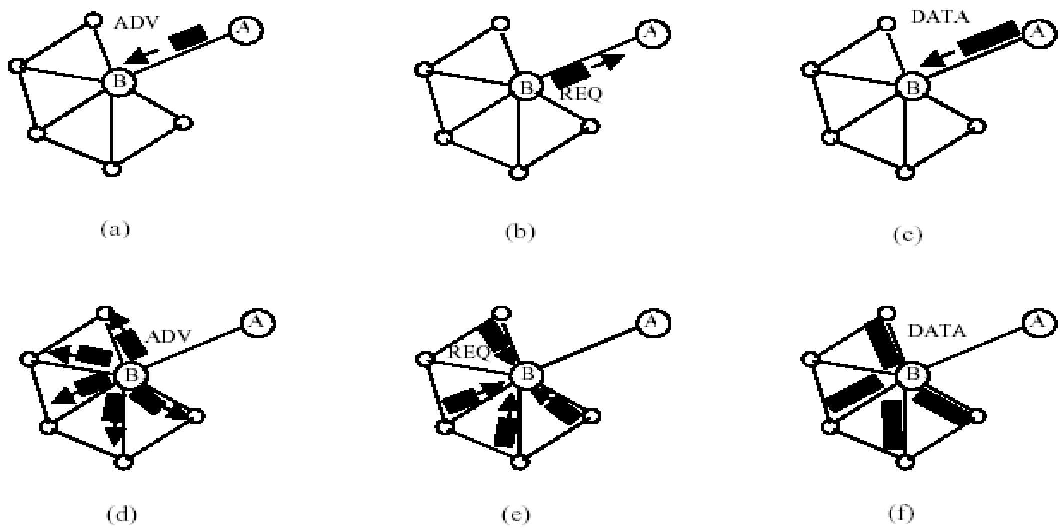


Figure 3.1: SPIN Protocol. Node A Starts by Advertising Its Data to Node B (a). Node B Responds by Sending a Request to Node A (b). After Receiving the Requested Data (c). Node B then Sends Out Advertisements to Its Neighbors (d). who in Turn Send Requests Back to B (e-f) [13]

Advantages:

- SPIN solves the classic problems of flooding such as redundant data transmission, overlapping of sensing areas and resource blindness.

- SPIN achieves energy and bandwidth efficiency without sending extra and unnecessary copies of data.

- Two nodes sensing the same region does not send similar packet to the same neighbor.

- SPIN has access to the current energy level of the node and adapts the protocol it is running based on how much energy is remaining.

- When an advertisement packet is come to a sensor node with low level energy which does not have this data, instead of retrieving the data by sending a request message, to save energy it may not request the data. The saved energy may be used with high important events.

- Sensor nodes operate more efficiently and conserve energy by sending description of data instead of sending all the data.

- Topology changes are localized since each node needs to know only its single-hop neighbors.

Disadvantages and Drawbacks:

- SPIN's data advertisement mechanism cannot guarantee the delivery of data. If the nodes that are interested in the data are located for away from the source node and the nodes between source and destination are not interested in that data, such data will not be delivered to the destination at all.

- SPIN is not scalable. The nodes around a sink could deplete their battery quickly if the sink is interested in too many events.

3.1.2.2 Directed diffusion

Directed diffusion [16, 17] uses attribute-value pairs for the data. In order to create a query, an interest is defined using a list of attribute value pairs. The interest is broadcasted by a sink through its neighbors. Each node has cache to save interests for later use. The interests in the caches are used to compare the received data with the values in the interests.

Neighbor node that sends the interest is called gradient. As the interest is propagated throughout the network, gradients are set up to draw data satisfying the query toward the requesting node. Gradient is characterized by the data rate, duration, and expiration time derived from the received the interest's fields. By utilizing interest and gradients, paths are established between sink and sources. One of the good paths is selected by reinforcement to send the original interest message.

The goal is to find a good aggregation tree that gets the data from source nodes to the sink. The sink periodically refreshes and resends the interest when it starts to receive data from the source(s) because of the unreliability of the network. Figure 3.2 [13] summarizes the directed diffusion protocol.

In directed diffusion, the sink queries the sensor nodes if a specific data is available by flooding some tasks, while in SPIN sensors advertise the availability of data allowing interested nodes to query that data. There is an on-demand data querying mechanism in directed diffusion.

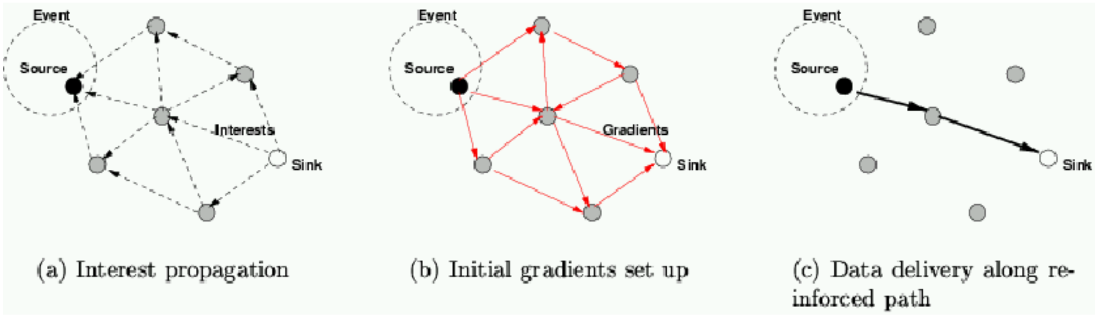


Figure 3.2: Directed Diffusion [13]

Advantages:

- When a path between a source and the sink fails, a new or alternative path is chosen to send interest.
- Caching can increase the efficiency, robustness, and scalability of coordination between sensor nodes.
- All communication is neighbor to neighbor, there is no need for node addressing mechanism.
- Each node can do sensing, aggregation and caching.
- Caching of interests minimizes energy consumption and delay.
- There is no need for maintaining global network topology; this also makes SPIN an energy efficient protocol.

Disadvantages and Drawbacks**:**

- The applications that require continues data delivery to the sink do not work efficiently with directed diffusion which has a query-driven on demand routing protocol model.
- Naming schemes are application dependent and each time should be defined a priori. Matching process for data and queries require some extra overhead.
- To implement data aggregation, it employs time synchronization technique, which is not easy to realize in a sensor network.
- The overhead involved in recording information in data aggregation is too much.
- All of these may lead to increasing the cost of a sensor node.

3.1.2.3 Rumor routing

Rumor routing [18] combines query flooding and event flooding protocols in a random way. The key idea is to route the queries to the nodes that have observed a particular event rather than flooding entire network to retrieve information about the occurring events. Arbitrary paths are discovered instead of the shortest paths from an event source to a sink. Each node maintains a list of neighbors and an event table with forwarding information for all the events it is aware of. In order to flood events through the network, the rumor routing algorithm employs long-lived packets called agents. When a node detects an event, it adds such event to its local table and generates an agent. Agents travel on a random path with related event information. When a node generates a query for an event, the nodes that know the route, can respond to the query by referring its event table. Only one path is maintained between source and destination. In Figure 3.3 [18], path establishment in Rumor routing is shown.

To make the protocol more efficient, agent aggregates event information stored in the nodes on the random path and the visited nodes updates their event information if better routes are found in the agent’s event information.

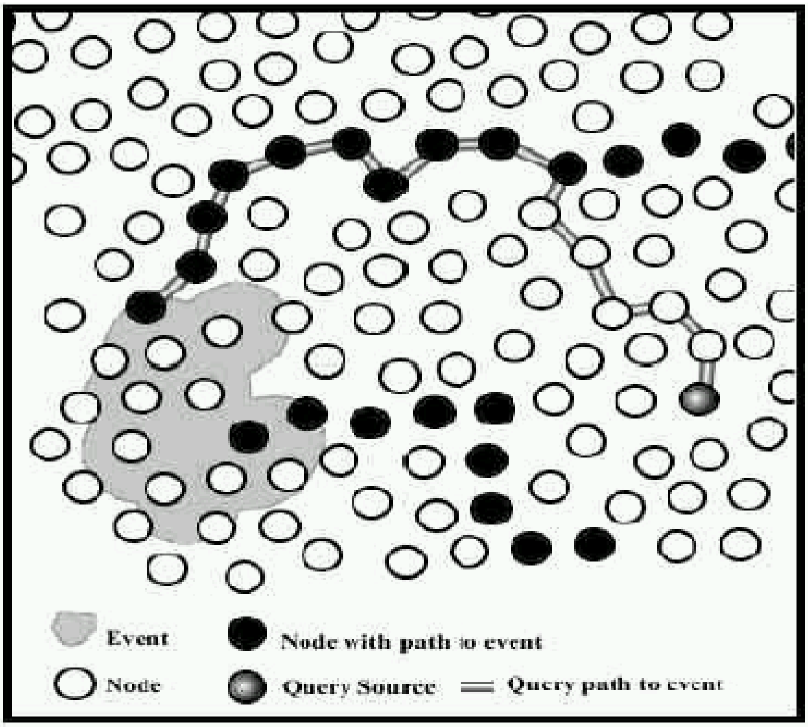


Figure 3.3: Rumor Routing [18]

Advantages**:**

- Rumor routing achieves energy saving and can also handle node's failure.

Disadvantages and Drawbacks:

- Rumor routing is attractive only when the number of queries is large and the number of events is small. For other situations, query flooding or event flooding are more efficient.

- Overhead to maintain agents and event tables.

- Overhead should be tuned through adjusting parameters used in the algorithm such as time-to-live for queries and agents.

3.1.2.4 Minimum cost forwarding algorithm (MCFA)

In MCFA [19], each node maintains the least cost estimate from itself to the sink. Each node stores its cost to the sink. The sink broadcast its own cost (0 initially) to its neighbors with an ADV message. Each node sets a back-off time when receives the message. Basically, back-timer expires if the new cost is less than the old one. The new cost is the sum of the cost of its immediate previous node and the cost consumed during the previous transmission. Once the times expire, the node changes its cost to the new one and rebroadcast the ADV message containing the new cost.

When a source has data to send to the sink, it simply broadcast it. Only nodes, which have a cost that matches the difference between the cost contained in the message and the consumed cost, rebroadcast the data. This process is continued until the data arrive at their destination.

Advantages:

- The cost values for each node are same as flooding.

- Optimal forwarding is achieved with minimum number of advertisement messages.

- The average number of advertisement messages in flooding is reduced using the back-off based algorithm.

Disadvantages and Drawbacks:

- Delays, channel errors, node failures should be considered.

- If there are many sinks, nodes have to store large amount of cost information related to those sinks.

- When the network size is too large, the time to set the cost field becomes intolerable.

- Nodes with lower cost to the sink may deplete their energy very soon.

3.1.2.5 Gradient based routing (GBR)

In GBR [20], when the interest is diffused through the whole network, number of hops is memorized. Each node can calculate the minimum number of hops to the sink, which is called height of the node. The difference between a node’s height and its neighbor’s height is considered the gradient on that link. A packet is forwarded on a link with the largest gradient on that link.

GBR uses data aggregation and traffic spreading in order to uniformly divide the traffic over the network. Three different data dissemination techniques in GBR are:

- Stochastic Scheme: When there are more than one hop with the same gradient, one of them are chosen at random.

- Energy based Scheme: When a node’s energy is dropped below a certain threshold, it increases its height. With this method, other sensors are discouraged from sending data to that node.

- Stream based Scheme: New streams are not routed through nodes that are currently part of other streams.

Advantages:

- To obtain balanced distribution of the traffic in the network causes increase in the network lifetime.

3.1.2.6 Constrained anisotropic diffusion routing (CADR) and information driven sensor querying (IDSQ)

In CADR [28], the key idea is to query sensors and route data in the network such that information gain is maximized while latency and bandwidth are minimized. Only the sensors that are close to a particular event are activated. Each node evaluates an information/cost objective and routes data based on the local information/cost gradient and end-user requirements. The major difference from directed diffusion is the consideration of information gain in addition to communication gain.

Estimation theory was used to model information utility. CADR is more energy efficient than directed diffusion.

In IDSQ, queries are diffused in an isotropic fashion and reach nearest neighbors first. Querying node can determine which node can provide the most useful information while balancing the energy cost. It provides a way of selecting the optimal order of sensors to achieve maximum information gain.

Advantages:

- Information gain is maximized while latency and bandwidth are minimized

- It is more energy efficient than directed diffusion.

3.1.2.7 COUGAR

COUGAR [22] provides a network layer independent solution that abstract, "query processing" from the network layer functions such as selection of relevant sensors and utilizes in-network data aggregation to save energy. Sensor nodes select a leader node to perform aggregation and transmit to the gateway (sink). Gateway generates query plan which gives information about data flow, makes in-network computation for the incoming query and sends it to the relevant nodes. Figure 3.4 [13] summarizes the COUGAR protocol.

Advantages:

- The ability of in-network computation also ensures energy efficiency.

Disadvantages and Drawbacks:

- It provides an additional query layer which means an extra overhead to sensor nodes in terms of energy consumption and storage.

- Synchronization is required to provide in-network data computation from several nodes.

- Leader nodes should be prevented from failure.

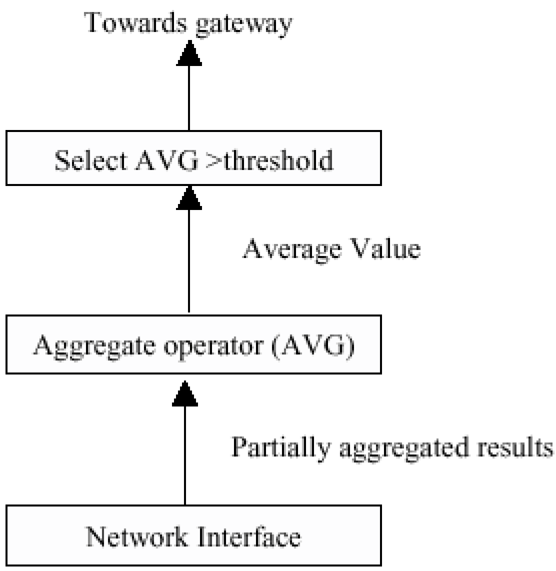


Figure 3.4: Query Plan at a Leader Node in COUGAR [13]

3.1.2.8 ACQUIRE

In ACQUIRE [23], complex queries can be further divided into several sub-queries. Sink sends a query and each node that receives this query tries to respond to the query partially by using its pre-cached information and then forwards it to another sensor node. If the pre-cached information is not up-to-date, the nodes gather information from their neighbors within d hops. At the end, resolved query is sent to the sink using reserve or shortest path. ACQUIRE deals with complex queries by allowing many nodes to send responses. A mathematical modeling is used to find an optimal value of the parameter d. To forward the query, next node is selected randomly or selection is based on max potential query satisfaction.

Advantages:

- Provide solution to one-shot, complex queries with an energy efficient way.

3.1.2.9 Energy aware routing

Energy aware routing [24] is destination initiated reactive protocol which increases the network lifetime. It maintains a set of paths. A path is chosen based on the energy consumption levels of each. Paths are chosen at different times, so the energy of any single does not deplete quickly. This causes increment in network lifetime.

Class based addressing is used for each node. Costs of each route are found, routing tables are build using localized flooding. Forwarding tables are built choosing low cost paths, and they are used to send data to the destination.

Advantages:

- Compared to directed diffusion, this protocol provides an overall improvement of energy saving and increase in network lifetime.

Disadvantages and Drawbacks:

- Single path usage hinders the ability of recovering from a node or path failure as opposed to directed diffusion.

- Requires gathering the location information and setting up the addressing mechanism for the nodes, which complicate route setup compares to the directed diffusion.

3.1.2.10 Routing protocols with random walks

Random walks approach [25] achieves multi-path routing as well as some kind of load balancing in a statistical sense. The location information or lattice coordination is obtained by computing distances between nodes using the distributed asynchronous version of the Bellman-Ford algorithm. For each intermediate node, it selects one of its neighbors which are closer to the destination according to a computed probability as next hop. Some kind of load balancing is assured if the probability is well computed.

Advantages**:**

- It balances routing or communication load and little state information need to be kept by nodes.

Disadvantages and Drawbacks:

- The topology of the network may not be practical.

3.1.3 Hierarchical based routing protocols

Low energy adaptive clustering hierarchy (LEACH) [26], power efficient gathering in sensor information systems (PEGASIS) [27] are in this category. These techniques tackle with scalability factor by clustering nodes for routing. For example, in LEACH any sensor node can elect itself as a cluster head at any time with a certain probability. Sensor nodes access the network through the cluster head that requires minimum energy to reach.

3.1.4 Location Based Routing Protocols

Location based algorithms such as minimum energy communication network (MECN) and small MECN (SMECN) [28] make routing decisions based on geographic locations of sensor nodes. In SMECN it is assumed that the exact locations of sensor nodes are known. Based on these locations, a sensor network is represented as a graph. Then the sub-graph that connects all nodes with minimum energy cost is computed by using a graph theoretic approach.

3.2 Power Control in WSNs

In the literature, there are many researches about routing problems in sensor networks. But there are limited researches about power control in sensor and actuator networks.

3.2.1 Power Control Methods

Power control in WSNs provides multiple benefits. It allows the large number of sensor nodes to efficiently share the wireless medium with minimal interference achieving required quality of service (QoS) levels, reduces the power consumption of individual sensor nodes thus increasing the battery mean life time, reduces the overall energy consumption of the network, maintains the network connectivity and increases the network capacity with spatial reuse improvement of the wireless channel by letting more users transmit at the same time. Power control algorithms can be classified in many different ways [29]:

a) Open Loop/Closed Loop/Combined Closed and Open Loop: For open loop power control a node adjusts its transmission power level inversely proportional to the averaged received power. In closed loop power control mechanism, the receiver node sends a measurement of received power back to the sender node which makes the sender node to adjust its transmission power based on the feedback provided by the receiver node. Both open loop and closed loop power control mechanisms help to combat with path loss and shadowing while only closed loop mechanisms overcomes multipath fading.

b) Centralized/Decentralized: In centralized power control mechanism a centralized controller manages the transmission power level of the nodes in the network. A decentralized power control algorithm controls only the transmission power of one single node depending on the local information.

c) Strength-based/ SIR-based/ BER-based: The measured quantity for power control can be the strength of a signal arriving at the receiver node, the SIR (Signal to Interference Ratio) or the BER (Bit Error Rate).

d) Fixed Step Size/Adaptive Step Size: The transmission power update strategy can be either fixed (fixed step size algorithm) or can be made adaptive to the channel variations. Power control command in fixed step size algorithms is a simple 1-bit command while with the adaptive step size approach it is possible to increase or decrease the transmission power by the actual difference between the received signal power and the desired received signal power.

e) Continuous power/Discrete power: Transmission power level can be controlled in the continuous or discrete power domain.

f) Common Power Control (CPC)/Independent Power Control (IPC) [30]: The transmission power is determined based on the dynamic network conditions. While in CPC all nodes use the same transmission power, IPC allows nodes to use independent transmission powers.

Power control mechanisms can be applied in any layer of the networking protocol stack. Low power modes, where for example the devices are powered on periodically, can be used in physical layer. Power controlled data link layer protocols can be implemented in order to limit the amount of unnecessary retransmissions or reduce collisions. Network layer can make use of power-aware routing protocols. Among these the most suitable layer to apply a power control mechanism seems to be the MAC layer since it determines the state of the radio (transmit, receive or sleep modes) effecting the overall energy consumption more than other layers of protocol stack.

3.2.2 Power control algorithms in routing layer

3.2.2.1 LMA and LMN

Two algorithms for dynamically adjusting transmission power level for each node of the fixed wireless sensor and actuator networks are proposed in [31]. In an indoor sensor environment, these local algorithms outperform fixed power level assignment.

The same power is used when sending to any neighbor regardless its distance to its neighbors. In the “local mean algorithm” (LMA), if the numbers of reachable neighbors of a node are less than the minimum number of neighbors required to ensure connectivity, the node increases its transmission power level by a fixed amount. If the number of reachable neighbors is more than the maximum number of neighbors required, the node decreases its transmission power. The transmission power must be in the range of minimum and maximum transmission powers. In the “local mean of neighbors algorithm” (LMN), the number of neighbors of its neighbors are also used to adjust transmission power of a node. If the average number of neighbors of neighbors is less than the minimum threshold, the node increases its transmission power. If it is more than maximum threshold, the node decreases its transmission power. In these two algorithms, all nodes converge very quickly. The connectivity achieved by LMN is higher comparing to the one achieved by LMA. The lifetime of the network is achieved by these local algorithms are about two times more than the global algorithms which assigns fixed transmission powers to each node.

3.2.2.2 COMPOW

The feedback power control algorithm COMPOW (Common Power) [32] finds the minimum power level, at which all nodes are connected. The network layer has a routing table for each power level. A routing table is built by sending routing control packets at specified power level. A routing daemon corresponding to each power level is run at different ports. The power control agent decides the current network power level that data packets will send. Each routing daemon maintains the routing table for its own power level. Power level field is required in the packets. This algorithm makes a centralized solution to power problem in the network layer. But a common power may cause an increase in the overall power consumption at the network.

3.2.2.3 Energy efficient routing

A spectrum of new techniques to enhance the routing in sensor networks is developed in [33]. Their first approach aggregates packet streams in a robust way, resulting in energy reductions. Second, more uniform resource utilization is obtained by shaping the traffic flow. Several techniques, which rely only on localized metrics, are proposed and evaluated. The network lifetime is increase up to an extra 90%

beyond the gains of the first approach. a number of practical algorithms that handles exceptions when nodes are critical in the overall network connectivity. DCE combining scheme reduces the overall energy, while spreading approaches aim at distributing the traffic in a more balanced way.

4. PAMR, CPAMR AND PCPAMR IN WSAN

The main objective of PAMR is conveying the sensed data from multiple sources to multiple destinations by the minimum number of transmissions. While fulfilling this objective, PCPAMR provides also a tradeoff mechanism between end-to-end delay and power efficiency for the case where nodes can individually control their transmission power. Please note that end-to-end delay is an important quality of service (QoS) metric in many WSAN applications. The salient features of WSANs where our protocols can be used are as follows:

- Several actuators, and a high number of sensor nodes that may be attached multiple sensors are densely deployed in a sensing field.
- The sensor nodes communicate with each other through short-range radios and they are constrained in communication bandwidth, memory capacity, and available power.
- An actuator evaluates the sensed data received from the sensor nodes in order to make a decision for actuating its attached devices.
- A sensor node may feed many actuators with the sensed data.
- An actuator may fetch the sensed data from many sensor nodes.
- Each actuator has a unique identifier.
- Both actuators and sensor nodes may be mobile.

4.1 Power Aware Many to Many Routing (PAMR) Protocol

In PAMR actuators register their interest for data to the nodes in the sensor network by broadcasting a *registration message*. The *registration message* contains fields, which are updated by every node that relays the message. While the *registration message* is being disseminated through the network, the sensor nodes relaying the *registration message* build up their registration table by inserting a registration record with the fields extracted from the registration message. Sensor nodes derive a routing table from the registration table where there is a single record for every unique uplink node and sensing task pairs. When a sensed data packet is received, that is forwarded to the uplink nodes looked up from the routing table. Hence a many-to-many multicast tree is obtained for each sensing task, i.e., the sensed data measured by a specific type of sensor. Actuators can also deregister from a sensing task by broadcasting a *deregistration message*. A node that receives a *deregistration message* updates its registration and routing tables accordingly. We also provide procedures that make our scheme adapt itself to the changes in the number, locations, and the sensing task interests of the actuators, as well as power available in the nodes.

a. Task registration

header	node _id	actuator _id	echel on	minPA	totalPA	task(s)

b. Task deregistration

header	node_id	actuator_id	task(s)

c. Update message

header	node_id	task

d. Reroute message

header	node_id	forward_node_id	task

e. Deactivation message

header	node_id

f. Data message

header	uplink_node_id(s)	time	task(s)	data(s)

Figure 4.1: The Packet Types in PAMR

Six types of packets shown in Figure 4.1 are used in PAMR protocol. Note that (a-e) are control packets where (f) is a data packet. An actuator broadcasts a *task registration message (TAREM)* to inform the network about its interest for the sensed data. When a node does not need a previously registered sensing task anymore, it deregisters itself from that task by broadcasting a *task deregistration message (TADREM)*.

An *update message* is used to construct a new route to replace a failed route. A *reroute message* is used to inform the node that forwards a message when the message cannot be relayed towards the intended destination. A *deactivation message* indicates that the source node will be unavailable soon due to energy depletion. *Data messages* convey the sensed data between sensor nodes and actuators.

An actuator may run based on multiple types of sensed data such as temperature, humidity and proximity. Similarly several actuators may use the same type of sensed data. Therefore, every actuator must inform the sensor network about the type of sensed data that it is interested in. This is achieved by broadcasting a TAREM.

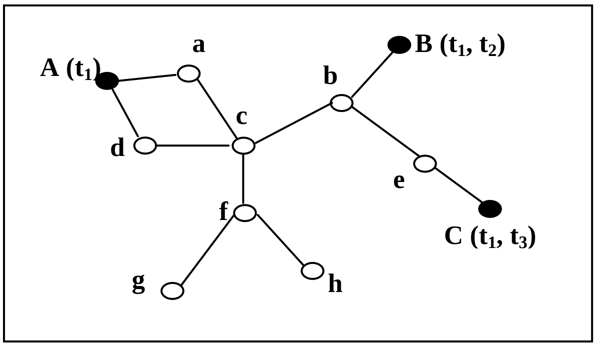


Figure 4.2: Task Dissemination in PAMR

While a TAREM is being disseminated in the network, the nodes maintain two tables: a registration table and a routing table. Examples for registration and routing tables are given in Tables 4.1 and 4.2 respectively for the WSAN shown in Figure 4.2.

Table 4.1: The Registration Table of Node c in Figure 4.2

Actuator Id	Uplink Node Id	Echelon	MinPA	TotalPA	Task(s)
A	a	2	5	5	t_1
A	d	2	4	4	t_1
B	b	2	7	7	t_1,t_2
C	b	3	3	10	t_1,t_3

The registration table of a node is the list of actuators that have registered at least one task to the network. The routing table is the list of neighboring nodes that the sensed data is forwarded to, i.e., uplink nodes. When a new sensed data packet is received, this table is looked up for the uplink (i.e., next hop) nodes. The same type of sensed data may be relayed to multiple nodes. Therefore, there may be multiple records for the same task in a routing table. However, there is a single record for every unique uplink node and sensing task pair. The routing table is derived from the registration table.

Table 4.2: The Routing Table of Node c in Figure 4.2

Task(s)	Uplink Node Id
t_1	a
t_1	b
t_2	b
t_3	b

4.1.1 Task registration process

An actuator broadcasts a TAREM for the tasks that it is interested in. A TAREM is composed of the fields shown in Figure 4.1.a: node identification (node_id), actuator identification (actuator_id), echelon, minimum power available (minPA), total power available (totalPA) and task(s). Node_id is the identification of the sending node. When an actuator broadcasts a TAREM, it initializes the node_id field with its own id, and the nodes that repeat the message update this field. Every node that repeats a TAREM replaces the node_id field with its own id. Echelon means the minimum number of hops required to reach a node from an actuator.

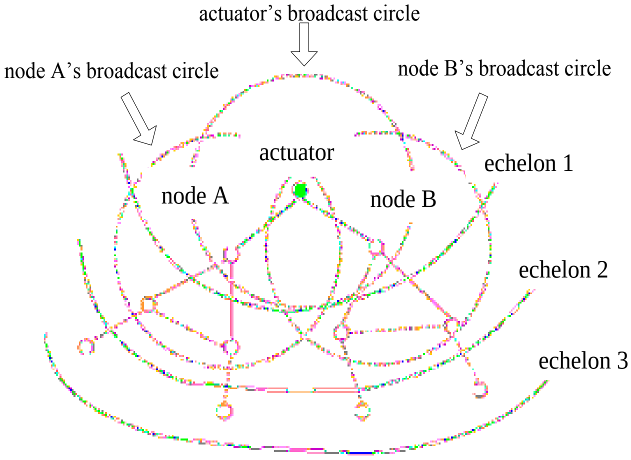


Figure 4.3: Echelons

In Figure 4.3 where circles represent the coverage areas of nodes, echelons are shown for the actuator. Since only nodes A and B are in the range of the actuator, they are the only nodes in echelon 1. The nodes that are in the range of A and B make echelon 2. The totalPA is found by summing up the power available in every node along the route. The minPA is the power available in the node that has minimum power along the route. A node that relays a TAREM adds its power available (ownPA) to the totalPA. It also replaces minPA field with its ownPA, if the ownPA is lower than the minPA value. Before transmitting a TAREM, the actuator initializes the echelon and totalPA as 0 and the minPA as the maximum possible PA value.

Since PAMR is designed for the case where every node transmits at the same power level, the minimum echelon has the minimum cost and minimum delay. However, both minPA and totalPA fields in a TAREM may have a crucial role to extend the overall network lifetime. Therefore a parameterized selection function formulated below is developed for PAMR.

Lets assume that we have a sensor node *s* which has data to send, and N = {n_1, n_2, …..n_n} is the set of uplink nodes in the routing table of *s*. The general formula for the selection function is:

$$f_i = (w_1 \times \alpha_i) + (w_2 \times \beta_i) + (w_3 \times \phi_i) \qquad \textbf{(4.1)}$$

where w_1, w_2, w_3 are the weighting parameters which satisfy $w_1+w_2+w_3 = 1$ and $0 \leq w_1, w_2, w_3 \leq 1$, and

$$\alpha_i = \frac{\sum_{k=1}^{n} e_k - e_i}{\sum_{k=1}^{n} e_k} \qquad \textbf{(4.2)}$$

$$\beta_i = \frac{m_i}{\sum_{k=1}^{n} m_k} \qquad \textbf{(4.3)}$$

$$\phi_i = \frac{t_i}{\sum_{k=1}^{n} t_k} \qquad \textbf{(4.4)}$$

where e_i is the echelon of the uplink node *i*, m_i is the minimum power available along the route via the uplink node *i*, and t_i is the total power available along the route via the uplink node *i*.

After calculating *f* values for all neighboring nodes, the node that has the maximum *f* value is selected as the uplink node to route an incoming data packet.

Sensor nodes use Algorithm 4.1 given below to relay TAREMs. If a node receives a TAREM, it first checks if the TAREM is from a new route. A route, which is one of the following, is accepted as a new route:

- The registration table does not have any entry for the actuator in the TAREM.
- The registration table has at least one entry for the actuator, but none of these entries for the actuator is from the uplink node in the TAREM.
- The registration table has an entry for the actuator and uplink node in the TAREM. However, at least one of the tasks in the TAREM is not indicated in the related registration table entry.

```
while(1)
{
        if receive (message)
        {
              if (message.type==TAREM)
              {
                    if (newRoute(message) or (betterRoute(message))
                    {
                     updateRegistrationTable (message);
                         if (updateRoutingTable)
                         {
                               modify(message);
                               broadcast (message);
                         }
                 }
                 else
                        discard(message);
            }//end if TAREM
            else
            {
                   .......
             }
}//end message
}//end while
```

Algorithm 4.1: Task Registration Algorithm

If the TAREM is not for a new route, then it is checked to determine if it is about a better route based on echelon, minPA and totalPA fields as explained before. If the TAREM is for a better route or a new route, the registration table is updated accordingly. Otherwise, the message is discarded. After the registration table is updated, the routing table is updated if it is a better route. If the routing table is updated, the uplink_node_id, echelon, minPA and totalPA fields in the TAREM are modified as explained above, and the modified TAREM is broadcasted. Note that our task registration scheme is different from flooding because sensor nodes relay only selected TAREMs.

We can explain how TAREM dissemination process runs by using the example shown in Figure 4.2. The TAREM sent by Actuator A is relayed by both Nodes a and d. Lets assume that Node c receives the TAREM relayed by Node a before the TAREM relayed by Node d. As soon as Node c receives the TAREM from Node a, it checks its registration table and finds out that this is the first TAREM from Actuator A. Therefore it inserts a record into its registration table, i.e., the first record in the registration table shown in Table 4.1. Then it inserts a new record for Task 1 through Node a into its routing table because Actuator A registers for Task 1 and the best route available in the registration table for Actuator A is through Node a. Since the received TAREM triggers a routing table update, the TAREM is modified and relayed by Node *c*. The TAREM relayed by Node *d* is also received by Node *c*. Although an actuator already in the registration table sends this TAREM, and the task registered by the actuator is also in the registration table, the relaying node is *c*, which is a new uplink node for Actuator A. Therefore, a new record is inserted into the registration table. However, we do not need to update the routing table because the current route for Actuator *A* and Task *1* is better than the new route. Therefore, Node *c* does not relay the TAREM received from Node *d*.

4.1.2 Task deregistration process

Task deregistration process is very similar to task registration process. A task deregistration message is made up of three fields as shown in Figure 4.1.b.: node_id, actuator_id and task fields. When an actuator does not need data from a task registered before, it broadcasts a task deregistration message. A node that receives a TADREM deletes the records related to the specified actuator and task from the registration table. Then the routing table is updated according to the new registration table. If there is a need to change the routing table, the TADREM is repeated. Otherwise it is discarded. Therefore, some deregistration messages may not be relayed even by the first node.

4.1.3 Data dissemination

When a sensor node senses data related to a task in its routing table, it creates a data packet and forward it to its neighbors in the routing table for this task. Before forwarding the data message, the node replaces the uplink_node_id field with the id(s) of the new uplink node(s) that the message will be forwarded to.

When a node receives a data message and its id is in the uplink_node_id field of the message, it looks up its routing table, and relays the data message to the uplink node(s) for the task(s) in the *task* field of the message. Before forwarding the data message, the node replaces the *uplink_node_id* field with the id(s) of the new uplink node(s). For example lets assume that Node *c* receives a data message, and finds out that Node *c* is in the uplink node list, and this data message is for Task 1. In its routing table shown in Table 4.2, there are two records for Task *1*: one is for Uplink Node *a*, and the other is for Uplink Node *b*. Therefore, it first replaces the ids in the *uplink_node_id* field of the data message with *a* and *b*, and then broadcasts the received data message, i.e., relays it.

If the data packet cannot be forwarded to an uplink node(s) (when acknowledgement for the data packet is not come), the record related to the uplink node(s) is removed from the registration and the routing tables. Then, a new uplink node whose route selection function value is more than the others is selected from the updated registration table, routing table is rearranged and the data packet is resent to the new uplink node. This process is repeated until either the uplink node receives the data packet or all records of this task(s) are removed from the registration table. Please refer to Algorithm 4.2 for further details about data dissemination process.

Since sensor nodes are prone to failure, it is possible to remove all uplink nodes for the task from the registration and routing tables at the end of this process. In this case, the node broadcasts an *update message*, which consists of two fields as shown in Figure 4.1.c: *node_id*, which broadcasts this message and *task_id* that the node requires to send data packet about. If there is a neighboring node that has an uplink node for the related task_id in its registration table, it prepares a TAREM and broadcasts it. After this everything is the same as the task registration process.

If the node is unsuccessful in the route reestablishment process (if the neighboring nodes that take update message has no information about requested task), it sends back a reroute message shown in Figure 4.1.d to the node that has forwarded the data packet. When a node receives a reroute message from an uplink node for a previously forwarded data packet, it behaves the same as the uplink node does not receive the forwarded data packet.

As a result, every sensor node has a enough knowledge in its routing and registration tables to send the sensed data to the actuators. Each sensor node knows how to transmit the data to the related actuator(s) through which neighbor node. Minimum hop route is the most power efficient route.

```
if (packet.type==DATA)
{
        if (packetIsForMe(packet))
        {
                if (isInRoutingTable(packet.task))
                {
                        sendDataPacket(uplinkNode);
                        if (not ACKComes)
                        {
                                routingTable.remove(uplinkNode);
                                registrationTable.remove(uplinkNode);
                                if (foundNewInRegistrationTable())
                                {
                                        routingTable.add(newUplinkNode);
                                        reSendDataPacket(newUplinkNode);
                                }
                                else
                                {
                                        sendUpdatePacket();
                                        if (not receiveTAREM())
                                                sendReroutePacket();
                                }
                        }
                }
                else
                {
                        sendUpdatePacket();
                        if (not receiveTAREM())
                                sendReroutePacket();
                }
        } // if packetIsForMe(packet)
        else
                discard(packet);
}
```

Algorithm 4.2: Data Dissemination Algorithm

4.2 Centralized PAMR (CPAMR)

Transmission power of nodes is calculated in a centralized manner according to the average hop distance. Average hop distance can be found from node density and connectivity. Using global average hop distance value, a proper transmission range (r) can be chosen and assigned to the whole sensor and actuator nodes. After calculating average hop distance between two nodes, a global transmission power is used in each sensor node communication.

4.3 Power Controlled PAMR (PCPAMR)

When nodes can adjust their transmission powers individually, the minimum hop route is not essentially the most power efficient route. Therefore, we need to modify our route selection function for PAMR such that it also fits the requirements of the case where nodes can control their transmission power. Moreover, we need to provide a tradeoff mechanism between the end-to-end delay and the total power used because they are the conflicting parameters. PCPAMR is the slightly modified version of PAMR that satisfies these conditions.

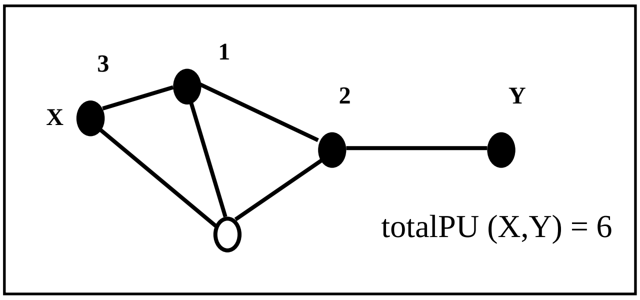


Figure 4.4: The Total Power Used from Node X to Y

Table 4.3: The Registration Table of Node c in Figure 4.2

Actuator Id	Uplink Node Id	Echelon	minPA	total PA	total PU	Task
A	a	2	5	5	3	t_1
A	d	2	4	4	4	t_2
B	b	2	7	7	2	t_1, t_2
C	b	3	3	10	3	t_1, t_3

In PCPAMR, a new field named totalPU (total power used) is added into the TAREM packet format. TotalPU indicates the total power used to convey a data packet through a given route. This is shown in Figure 4.4 where the totalPU for the highlighted route between nodes X and Y is 6 when the nodes transmit at the power levels given in the figure. Every node that relays a TAREM updates the totalPU field by adding the transmission power that it uses to repeat the TAREM. Hence the totalPU for the route that a TAREM is received from can easily be found out, and maintained in the registration table as shown in Table 4.3. Apart from the addition of totalPU field in the TAREM packets and the registration tables, we need only to modify the selection function for the uplink nodes in PCPAMR as follows:

$$f_i = (w_1 \times \alpha_i) + (w_2 \times \beta_i) + (w_3 \times \phi_i) + (w_4 \times \tau_i) \quad \textbf{(4.5)}$$

where w_1, w_2, w_3, w_4 are the weighting parameters which satisfy $w_1+w_2+w_3+w_4 = 1$ and $0 \leq w_1, w_2, w_3, w_4 \leq 1$, and

$$\tau_i = \frac{\sum_{k=1}^{n} u_k - u_i}{\sum_{k=1}^{n} u_k} \quad \textbf{(4.6)}$$

where u_i is the total power used along the route via the uplink node i.

5. PERFORMANCE EVALUATION

PAMR can be used without any modifications in conjunction with a network wide power control scheme where the transmission power of nodes is calculated in a centralized manner according to the average hop distance. This is the idea in the CPAMR protocol. In this subsection we provide the formulations for the calculation of the average hop distance from node density and connectivity. Using global average hop distance value, a proper transmission range (r) can be chosen and assigned to the whole sensor and actuator nodes.

The average hop distance m. can be found out by

$$m = \int_{0}^{\varphi} z\ f_z(z)\,dz \qquad \textbf{(5.1)}$$

where φ is the maximum possible distance between two sensor nodes, and given by

$$\varphi = \sqrt{h^2 + w^2} \qquad \textbf{(5.2)}$$

h is the height and w is the width of the sensor field. In Equation (5.3) f_Z(z) is the probability density function of the distance between a pair of nodes in a sensor field.

$$f_Z(z) = P\left(\sqrt{(x_n - x_e)^2 + (y_n - y_e)^2}\right) \qquad \textbf{(5.3)}$$

where (x_n, y_n) are the coordinates of the first node, (x_e, y_e) are the coordinates of the second node. We find f_Z(z) in two steps. First, we compute the probability density functions (pdf) of $X = X_n - X_e$ and $Y = Y_n - Y_e$; then we compute the pdf of $Z = \sqrt{X^2 + Y^2}$. At the first step by substituting $X_n = X + X_e$ and $Y_n = Y + Y_e$, we find

$$f_X(x) = \int_{-\infty}^{\infty} f_{X_n X_e}(x + x_e, x_e)\,dx_e \qquad \textbf{(5.4)}$$

$$f_Y(y) = \int_{-\infty}^{\infty} f_{Y_n Y_e}(y + y_e, y_e)\,dy_e \qquad \textbf{(5.5)}$$

Since X_n and X_e, Y_n and Y_e are independent random variables,

$$f_X(x) = \int_{-\infty}^{\infty} f_{X_n}(x + x_e) f_{X_e}(x_e) dx_e \quad \textbf{(5.6)}$$

$$f_Y(y) = \int_{-\infty}^{\infty} f_{Y_n}(y + y_e) f_{Y_e}(y_e) dy_e \quad \textbf{(5.7)}$$

At the second step to formulate the pdf of Z, an auxiliary random variable T, as T = X, is introduced. This will enable us to use the general formula of finding f_{ZT} from two functions of two random variables with n real roots, given below

$$f_{ZT}(z,t) = \sum_{i=1}^{n} f_{XY}(x_i, y_i) \left| \tilde{J}_i \right| \quad \textbf{(5.8)}$$

The equations

$$\begin{aligned} Z - \sqrt{X^2 + Y^2} &= 0 \\ T - X &= 0 \end{aligned} \quad \textbf{(5.9)}$$

have two real roots, for $|t| < z$, namely $x_1 = t$ and $x_2 = t$

$$y_1 = \sqrt{z^2 - t^2} \quad \textbf{(5.10)}$$

$$y_2 = -\sqrt{z^2 - t^2} \quad \textbf{(5.11)}$$

At both roots, $\left| \tilde{J} \right|$ has the same value:

$$\left| \tilde{J}_1 \right| = \left| \tilde{J}_2 \right| = \frac{z}{\sqrt{z^2 - t^2}} \quad \textbf{(5.12)}$$

Since X and Y are independent random variables, a direct application of Equation (5.8) yields

$$f_{ZT}(z,t) = \frac{z}{\sqrt{z^2 - t^2}} \left[f_X(x_1) f_Y(y_1) + f_X(x_2) f_Y(y_2) \right] \quad \textbf{(5.13)}$$

We get $f_x(x)$ and $f_y(y)$ in Equation (5.6) and (5.7), so we can find $f_Z(z)$:

$$f_Z(z) = \int_{-\infty}^{\infty} f_{ZT}(z,t) dt \quad \textbf{(5.14)}$$

Equation (5.14), that gives the probability that the distance between two nodes is less than z can be extended for the Gaussian distributions, where X_n, X_e, Y_n and Y_e are distributed according to N(0,$\sigma_{X_n}{}^2$), N(0,$\sigma_{X_e}{}^2$), N(0,$\sigma_{Y_n}{}^2$), N(0,$\sigma_{X_e}{}^2$) respectively, under $\sigma_{X_n} = \sigma_{X_e}$, $\sigma_{Y_n} = \sigma_{Y_e}$ condition. If we substitute functions in Equation (5.12), we get

$$f_{ZT}(z,t) = \begin{cases} \dfrac{1}{\pi\sigma^2}\dfrac{z}{\sqrt{z^2 - t^2}}, & z > 0, |t| < z \\ 0, & otherwise \end{cases} \quad \textbf{(5.15)}$$

$$f_z(z) = \int_{-\infty}^{\infty} f_{ZT}(z,t)dt = \frac{z}{\sigma^2} e^{-z^2/2\sigma^2} \left[\int_0^z \frac{dt}{\sqrt{z^2 - t^2}} \right] \frac{2}{\pi} u(z) \quad \textbf{(5.16)}$$

Since the term in parenthesis has value π/2, $Z = \sqrt{X^2 + Y^2}$, $f_z(z)$ is the Rayleigh density function where standard deviation σ is,

$$f_Z(z) = \frac{z}{\sigma^2} e^{-z^2/2\sigma^2} u(z) \quad \textbf{(5.17)}$$

Equation (5.12) can also be extended for uniform random variables X_n(0,w), X_e(0,w), Y_n(0,h) and Y_e(0,h) where w and h are the width and height of the sensor field respectively. If we solve Equation (5.4) for these random variables, we get

$$f_X(x) = \begin{cases} \displaystyle\int_0^{x+w} \frac{1}{w^2} d_{X_n} = \frac{w + x}{w^2}, & -w \le x \le 0 \\ \displaystyle\int_x^{w} \frac{1}{w^2} d_{X_n} = \frac{w - x}{w^2}, & 0 \le x \le w \end{cases} \quad \textbf{(5.18)}$$

$$f_Y(y) = \begin{cases} \displaystyle\int_0^{y+h} \frac{1}{h^2} d_{x_n} = \frac{h + y}{h^2}, & -h \le y \le 0 \\ \displaystyle\int_y^{h} \frac{1}{h^2} d_{y_n} = \frac{h - y}{h^2}, & 0 \le y \le h \end{cases} \quad \textbf{(5.19)}$$

Same steps are followed from Equation (5.6) to Equation (5.9) and then $f_X(x_1)$, $f_X(x_2)$, $f_Y(y_1)$ and $f_Y(y_2)$ are substituted in Equations (5.10) and (5.11).

$$f_{ZT}(z,t)=\frac{z}{\sqrt{z^2-t^2}}\begin{cases}(\frac{w+t}{w^2})(\frac{h+\sqrt{z^2-t^2}}{h^2})+(\frac{w+t}{w^2})(\frac{h-\sqrt{z^2-t^2}}{h^2}),-w\le x\le 0,-h\le y\le 0\\(\frac{w+t}{w^2})(\frac{h-\sqrt{z^2-t^2}}{h^2})+(\frac{w+t}{w^2})(\frac{h+\sqrt{z^2-t^2}}{h^2}),-w\le x\le 0,0\le y\le h\\(\frac{w-t}{w^2})(\frac{h-\sqrt{z^2-t^2}}{h^2})+(\frac{w-t}{w^2})(\frac{h+\sqrt{z^2-t^2}}{h^2}),0\le x\le w,0\le y\le h\\(\frac{w-t}{w^2})(\frac{h+\sqrt{z^2-t^2}}{h^2})+(\frac{w-t}{w^2})(\frac{h-\sqrt{z^2-t^2}}{h^2}),0\le x\le w,-h\le y\le 0\end{cases} \quad \textbf{(5.20)}$$

$$f_{ZT}(z,t)=\frac{z}{\sqrt{z^2-t^2}}\begin{cases}(\frac{2}{h^2})(\frac{1}{w}+\frac{t}{w^2}) & ,-w\le x\le 0,-h\le y\le h\\(\frac{2}{h^2})(\frac{1}{w}-\frac{t}{w^2}) & ,0\le x\le w,-h\le y\le h\end{cases} \quad \textbf{(5.21)}$$

where $z>0,|t|<z$ conditions must be satisfied.

$$f_Z(z)=\int_{-\infty}^{\infty} f_{ZT}(z,t)dt \quad \textbf{(5.22)}$$

$$f\ (z)=\begin{cases}\frac{2z}{hw}\int_0^z\frac{dt}{\sqrt{z^2-t^2}}u(z)+\frac{2z}{hw^2}\int_0^z\frac{tdt}{\sqrt{z^2-t^2}}u(z),-w\le x\le 0,-h\le y\le h\\\frac{2z}{hw}\int_0^z\frac{dt}{\sqrt{z^2-t^2}}u(z)-\frac{2z}{hw^2}\int_0^z\frac{tdt}{\sqrt{z^2-t^2}}u(z),-w\le x\le 0,-h\le y\le h\end{cases} \quad \textbf{(5.23)}$$

If we substitute v as $v=z^2-t^2$, so dv becomes dv= -2wdw and solve the integrals, since $z\ge 0$ and $|w|<z$, the probability density function of the distance between two sensor nodes becomes:

$$f_Z(z)=\begin{cases}(\frac{2z}{hw}\left[\arcsin\ \frac{t}{z}\right]_0^z+\frac{2z}{hw^2}\left[-\sqrt{z^2-t^2}\right]_0^z)u(z),-w\le x\le 0,-h\le y\le h\\(\frac{2z}{hw}\left[\arcsin\ \frac{t}{z}\right]_0^z-\frac{2z}{hw^2}\left[-\sqrt{z^2-t^2}\right]_0^z)u(z),-w\le x\le 0,-h\le y\le h\end{cases} \quad \textbf{(5.24)}$$

$$f_Z(z)=\begin{cases}\frac{2z}{hw}(\frac{\pi}{2}+\frac{z}{w})u(z), & -w\le x\le 0,-h\le y\le h\\\frac{2z}{hw}(\frac{\pi}{2}-\frac{z}{w})u(z), & -w\le x\le 0,-h\le y\le h\end{cases} \quad \textbf{(5.25)}$$

6. EXPERIMENTAL RESULTS

In this section, simulation results are presented which verify the mathematical models introduced in the previous section. We evaluate the performance of PAMR in terms of average power consumption, average end to end delay, and average number of data and control packet transmissions through simulations in ns-2.

Table 6.1: Inputs Used in Our Simulations

PARAMETERS	VALUES
Number of nodes	30, 60, 100
Number of actuators	1 – 8
Size of sensor field	800 x 800
Node distribution	Random (uniform)
Packet size	10 bytes
Unsuccessful delivery rate	0% - 70%
Sensor nodes' initial energies	1000 joules
Actuators' initial energies	5000 joules
Required energy to transmit a packet	0,2 joules
Required energy to receive a packet	0,1 joules
MAC Protocol	IEEE 802.11
w1 (weighting parameter for echelon)	0,1, 0.3, 0.4, 0.8
w2 (weighting parameter for minPA)	0.1, 0.3, 0.4, 0.8
w3 (weighting parameter for totPA)	0.1, 0.3, 0.4, 0.8

The factoring parameters in our simulations are number of nodes, number of actuators, network congestion ratio and weighting parameters. The values assigned to our simulation parameters are depicted in Table 6.1.

a) Impact of the changes in network size

We first examine the sensitivity of PAMR and PCPAMR to the changes in the number of nodes in the network. In these experiments the number of actuators is always 3 and *w1*, *w2* and *w3* are 0.3, 0.3 and 0.4 respectively.

In Figure 6.1, we show the average number of data packet transmissions per node for varying network sizes, i.e., the number of nodes in the sensor field. PAMR is not sensitive for the changes in network size. We observe slight reduction in the number of packets transmitted as the network size increases. When more nodes are deployed, PAMR can find out lower hop routes, and the number of packet transmissions decreases.

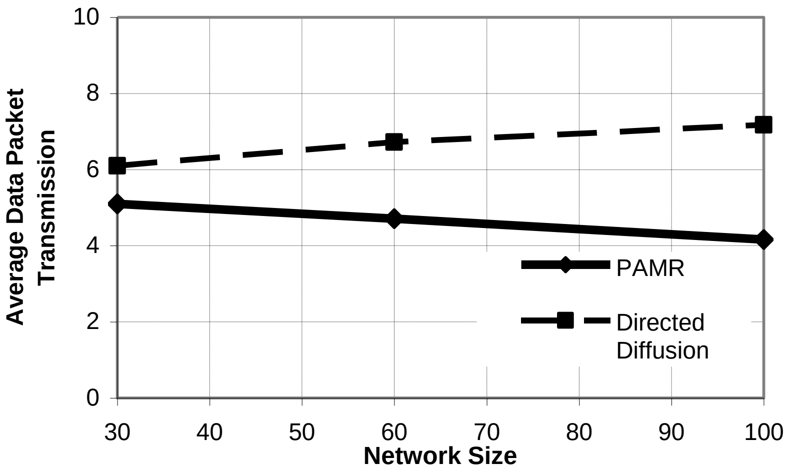


Figure 6.1: Average Data Packet Transmission Rate for Varying Network Size

In Figure 6.2, the average number of control packet transmissions is shown for varying network sizes. Similar to the data packet transmissions, the control packet traffic is not sensitive to varying network sizes in PAMR, and the number of packet transmissions is approximately 50% more in directed diffusion.

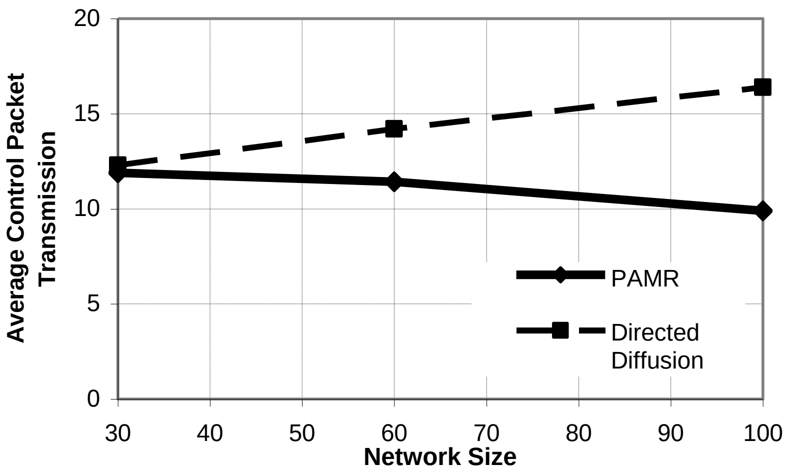


Figure 6.2: Average Control Packet Transmission Rate for Varying Network Size

In Figure 6.3, we show the power consumption in PAMR for varying network size. In PAMR and directed diffusion, the increase in the network size makes the nodes consume more power. The average power consumption in PAMR is 50% less than average power consumption in directed diffusion.

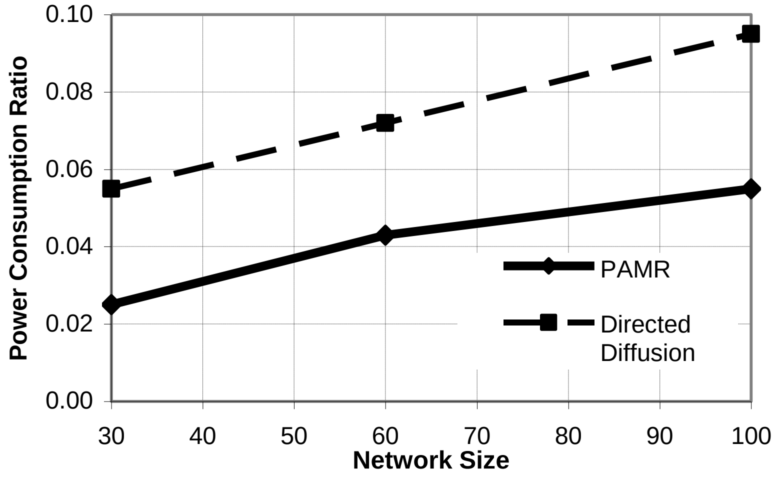


Figure 6.3: Average Power Consumption Ratio for Varying Network Size.

In Figure 6.4, the end to end delay performance of PAMR is shown. Average end to end delay increases as the network size gets higher both in PAMR and directed diffusion.

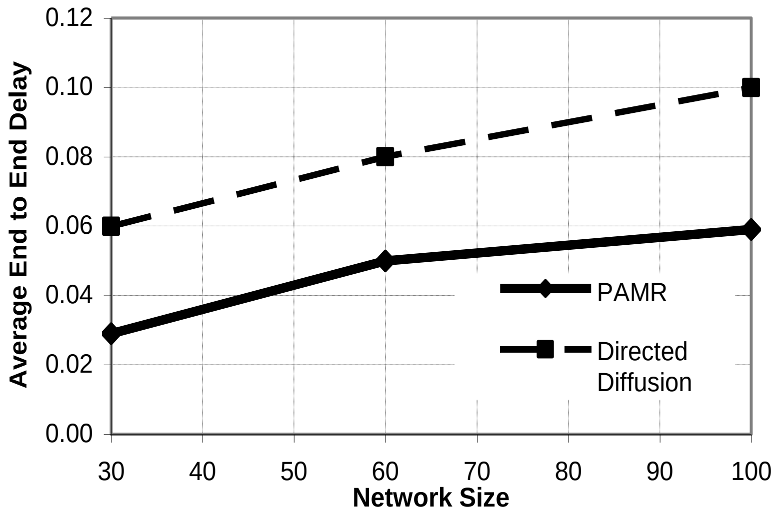


Figure 6.4: Average End to End Delay for Varying Network Size

b) Impact of the changes in the number of actuators

We also evaluate the impact of the changes in the number of actuators for the same network size. We deploy 30 nodes and varying number of actuators in these experiments. In these experiments, *w1*, *w2* and *w3* are 0.3, 0.3 and 0.4 respectively.

In Figure 6.5, the number of data packets transmitted in PAMR is depicted for varying number of actuators. Data traffic is not sensitive to the number of actuators.

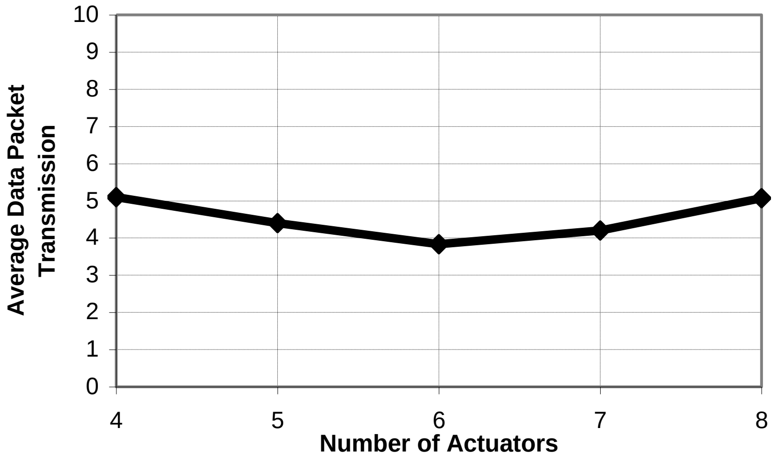


Figure 6.5: Average Data Packet Transmission Rate for Varying Number of Actuators

As shown in Figure 6.6, the average number of control packets per node increases for the higher number of actuators. This is also intuitively clear because actuators generate the control packets. However, please note that the number of control packets is not doubled when the number of actuators is doubled.

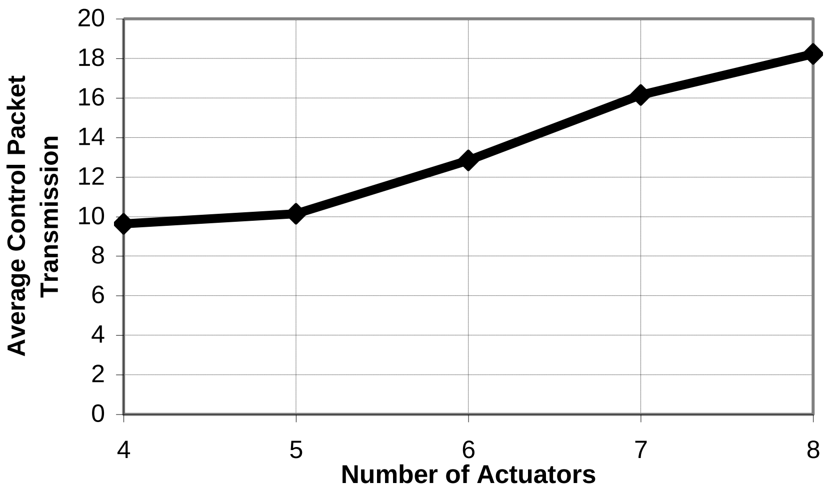


Figure 6.6: Average Control Packet Transmission Rate for Varying Number of Actuators

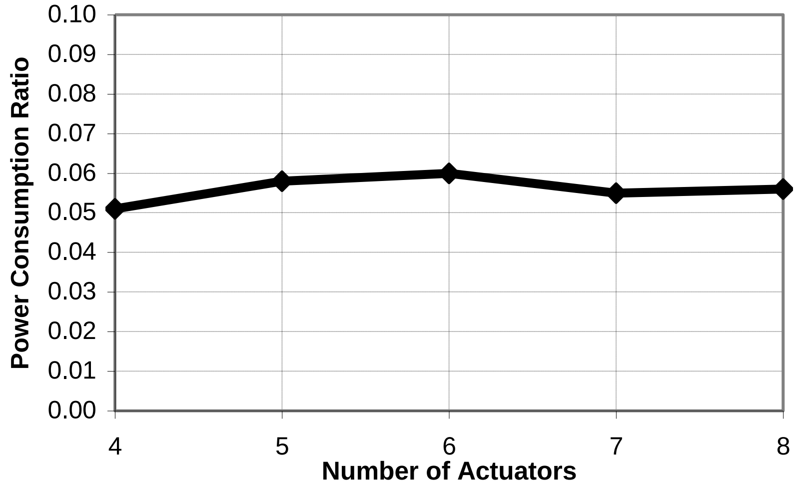


Figure 6.7: Power Consumption Rate for Varying Number of Actuators

In Figure 6.7, the average power consumption per node for varying number of actuators is depicted for PAMR. The power consumption in PAMR is not sensitive for the changes in the number of actuators.

In Figures 6.8, the average end to end delay per node for varying number of actuators is depicted for PAMR. Similar to the power consumption, average end-to-end delay in PAMR is not sensitive for the changes in the number of actuators.

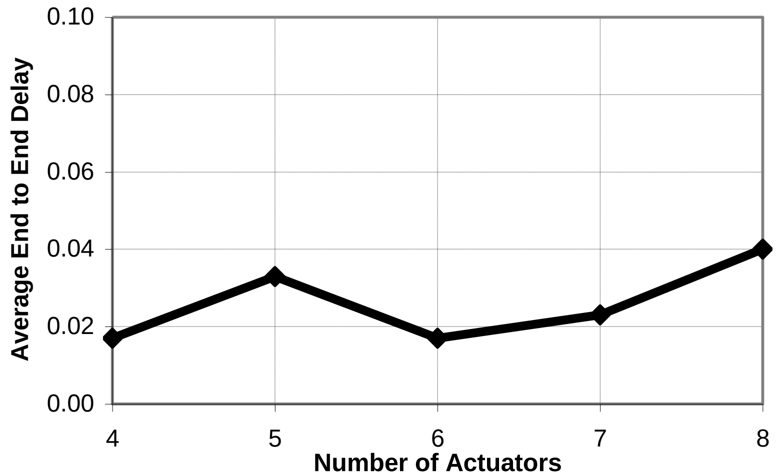


Figure 6.8: Average End to End Delay For Varying Number of Actuators

c) Impact of the changes in the unsuccessful delivery rate

We also evaluate the impact of the changes in the unsuccessful delivery rate (UDR), i.e., the ratio between the number of packets that cannot be delivered to the next hop and total number of packet transmissions per node. In these experiments the number of actuators is always 3 and *w1*, *w2* and *w3* are 0.3, 0.3 and 0.4 respectively.

In Figure 6.9, we examine the number of data packets transmitted in PAMR for increasing UDR. Data traffic decreases as the UDR increases because of the decrease in the number of available uplink nodes to transmit the data packets.

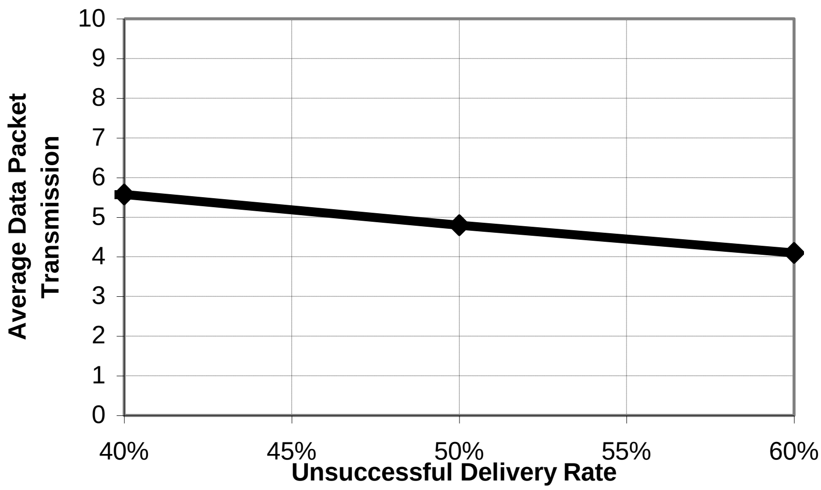


Figure 6.9: Average Data Packet Transmission Rate for Varying Unsuccessful Delivery Rate

In Figure 6.10, the number of control packets transmitted in PAMR is shown for increasing UDR. The average number of control packets per node increases as the UDR increases until it becomes 40%. After that point, it starts decreasing because control packets cannot reach some sensor nodes when UDR is higher.

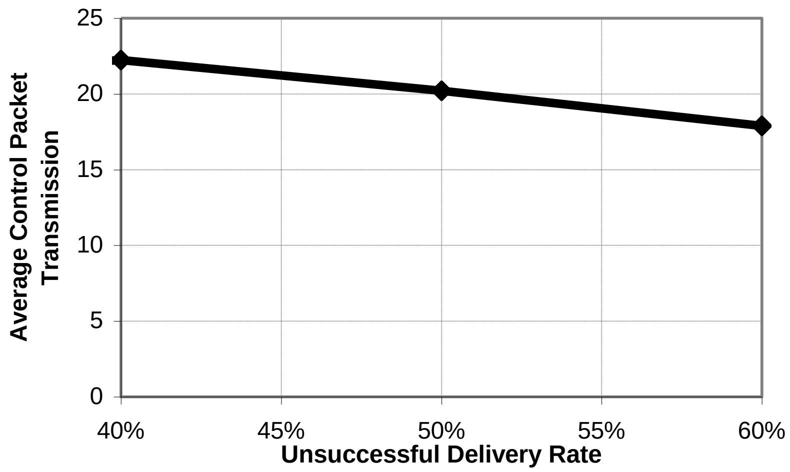


Figure 6.10: Average Control Packet Transmission Rate for Varying Unsuccessful Delivery Rate

In Figure 6.11, the power consumption performance of PAMR is given for increasing UDR. The average power consumption in PAMR is not sensitive for the changes in UDR.

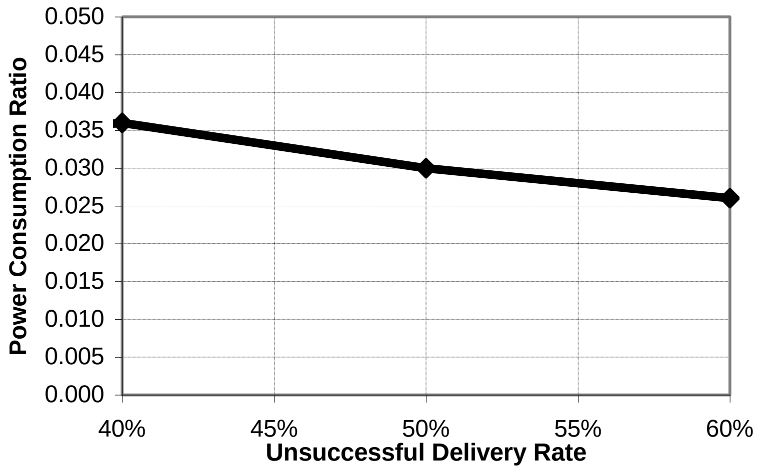


Figure 6.11: Power Consumption Ratio for Varying Unsuccessful Delivery Rate

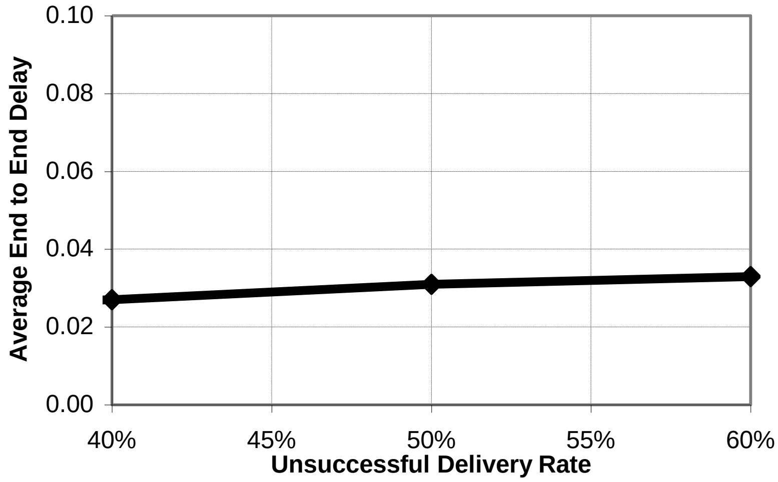


Figure 6.12: Average End to End Delay for Varying Unsuccessful Delivery Rate

In Figure 6.12, the average end to end delay in PAMR is given for increasing UDR. As the UDR increases, end to end delay also gets higher.

d) Impact of the changes in weighting parameters

We also evaluate the impact of the changes in weighting parameters for the varying network size with 3 actuators. The behavior of PAMR is investigated when weighting parameter *w1* (for echelon) is high and the others (*w2* and *w3*) are low, *w2* (for minPA) is high and the others (*w1* and *w3*) are low and *w3* (for totPA) is high and the others (*w1* and *w2*) are low in route selection formula as given in Equation 1. Here low is 0,1 and high is 0,8.

In Figure 6.13, the average number of data packet transmissions for varying weighting parameters is depicted. When *w1* is high, network size has almost no effect on the performance of PAMR. That is also intuitively clear because when *w1* is high, PAMR always selects the minimum hop routes. However, when *w2* or *w3* is high, number of the data packet transmissions gets higher as the network size increases.

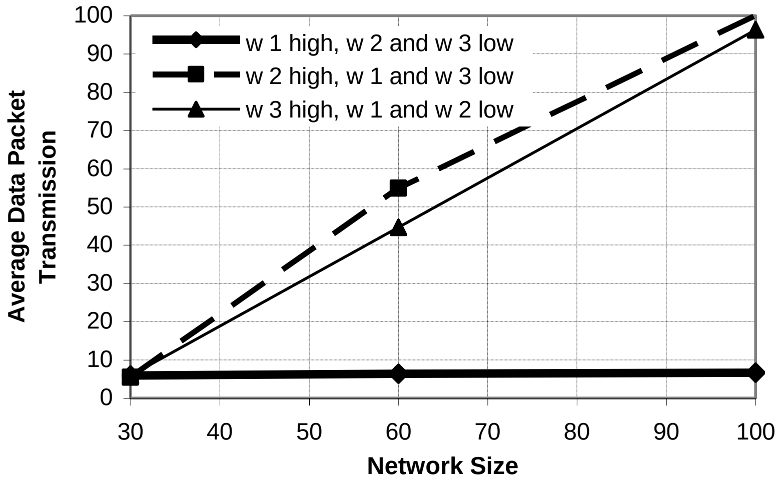


Figure 6.13: Average Data Packet Transmission Rate for Varying Network Size and Different Weighting Parameter Values

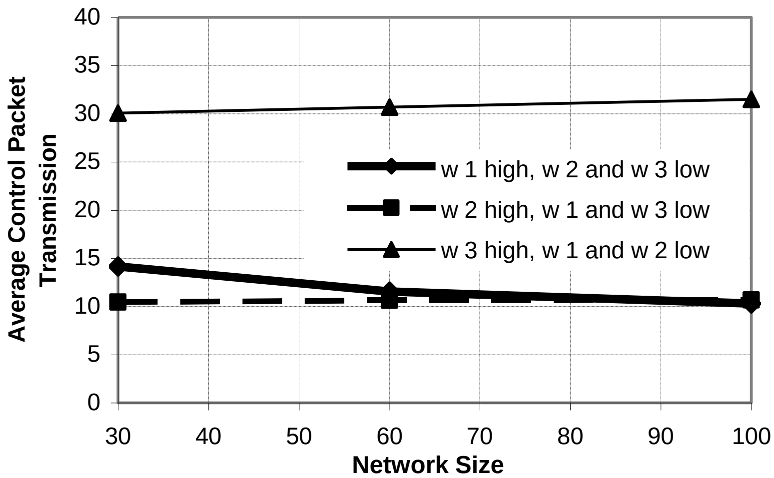


Figure 6.14: Average Control Packet Transmission Rate for Varying Network Size and Different Weighting Parameter Values

In Figure 6.14, the average number of control packet transmission for varying weighting parameters is depicted. The number of control packet transmissions is not sensitive to the network size in any values assigned to the weighting parameters.

In Figure 6.15, the power consumption ratio per node for varying weighting parameters is depicted. Here, our observations are almost the same as in the Figure 6.13.

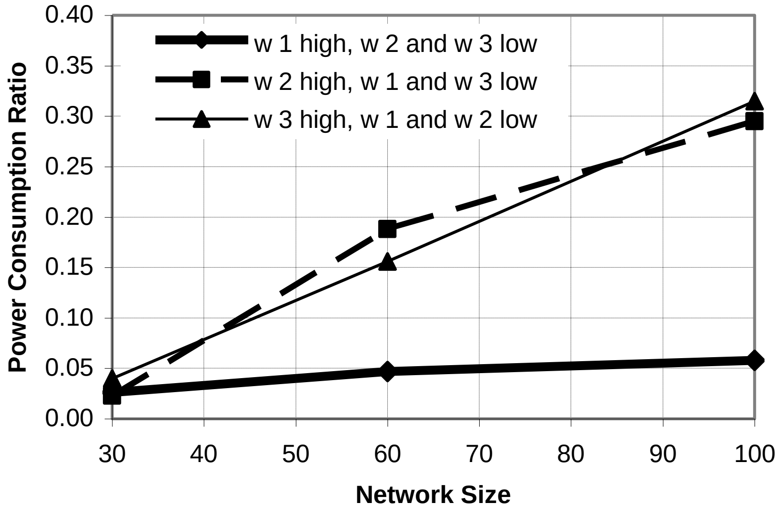


Figure 6.15: Power Consumption Ratio for Varying Network Size and Different Weighting Parameter Values

In Figure 6.16, we show the average end to end delay per node for varying weighting parameters. Here, our observations are again the same as in the Figure 6.13.

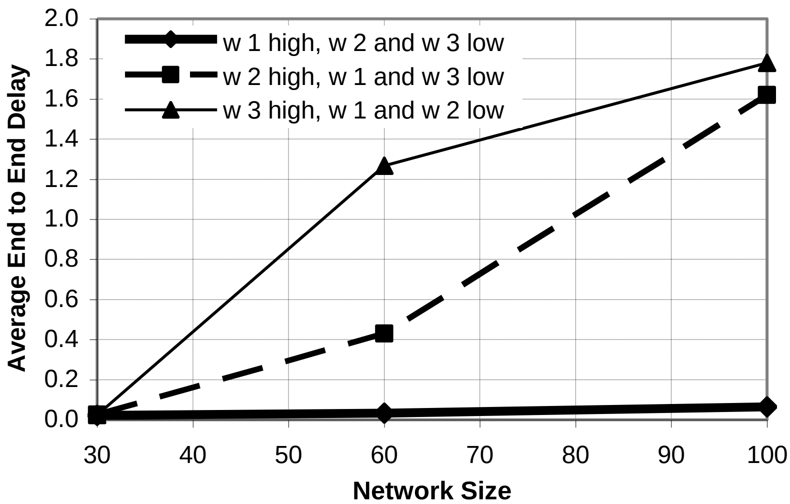


Figure 6.16: Average End to End Delay for Varying Network Size and Different Weighting Parameter Values

We summarize the outputs of our experiments for the sensitivity of PAMR for the changes in weighting parameters in Table 6.2.

Table 6.2: Sensitivity of PAMR for the Changes in Weighting Parameters

Sensitivity	***w1* high, *w2* & *w3* low**	***w2* high, *w1* & *w3* low**	***w3* high, *w1* & *w2* low**
Ave. data packet trans. rate	nearly constant	%900 increase	%900 increase
Ave. control packet trans. rate	nearly constant	nearly constant	nearly constant
Power consumption ratio	nearly constant	%900 increase	%900 increase
Ave. end to end delay	nearly constant	%1500 increase	%1700 increase

In Figure 6.17 and Figure 6.18, the distribution of sensor and actuators nodes and packet transmission between nodes in the sensor field are shown using Network Animator in ns-2. There are 30 nodes in this sensor field in Figure 6.17 and 100 nodes in Figure 6.18. The black nodes are actuators and the others are sensor nodes.

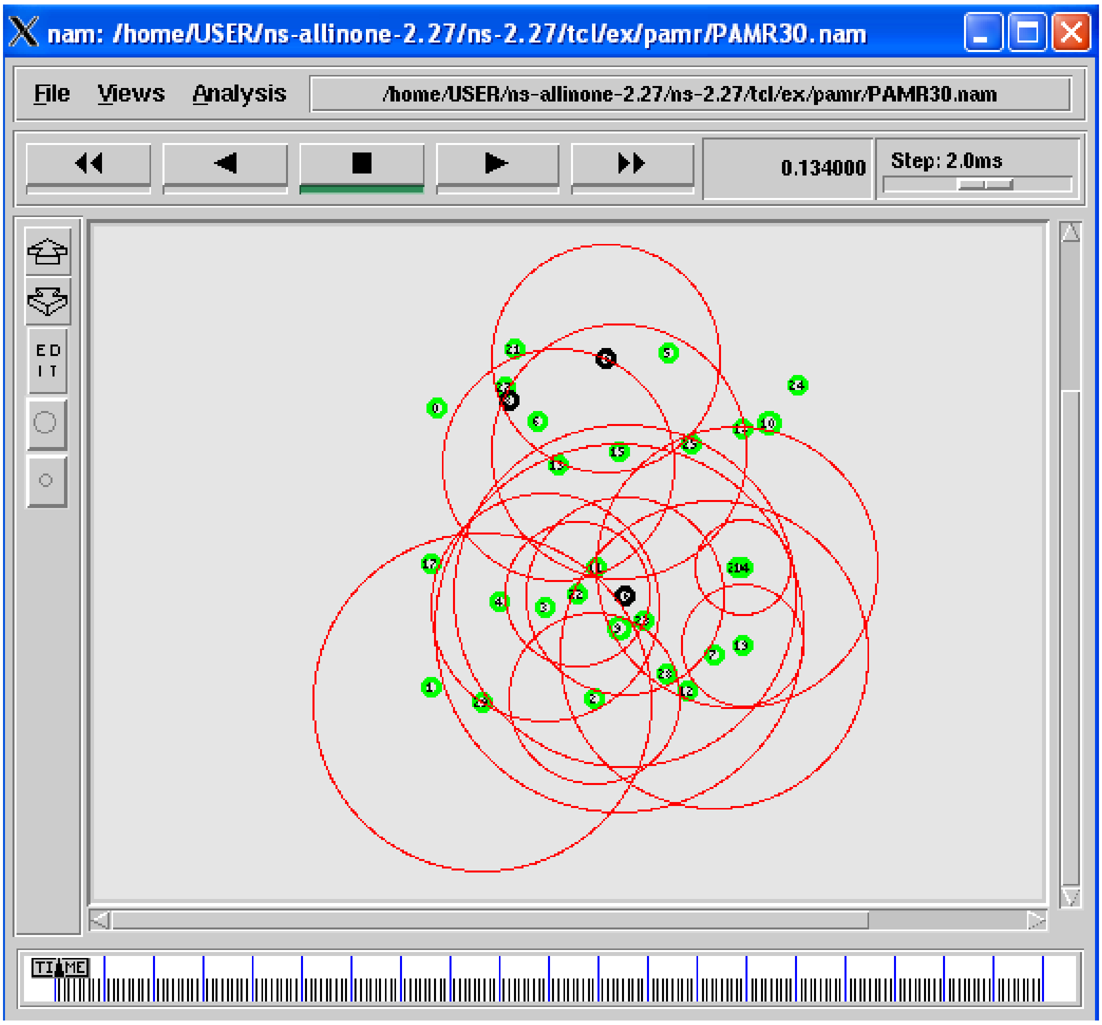


Figure 6.17: Display of 30 Nodes in Network Animator

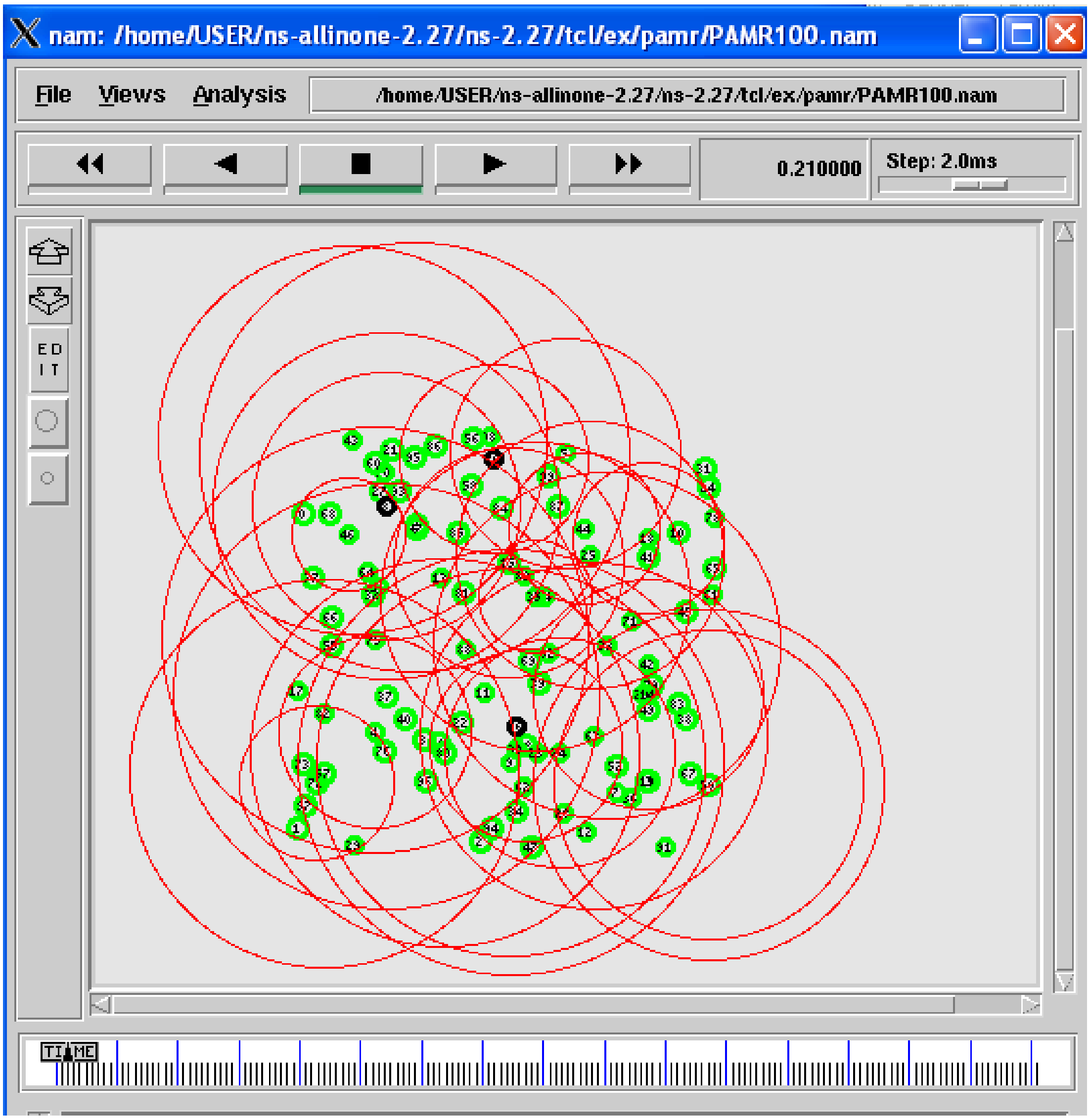


Figure 6.18: Display of 100 Nodes in Network Animator

7. CONCLUSIONS AND FUTURE WORK

In this thesis we introduce a new power aware many to many routing scheme for wireless sensor and actuator networks. In our scheme actuators register the types of the sensed data that they need to the network by broadcasting a task registration message. During the dissemination of the task registration message, the multicast tree for the registered task is updated such that the most power efficient route to the actuator that registers the task is included. This multicast tree has a many-to-many relation among the sensor nodes and actuators such that the sensed data generated by any sensor node is forwarded to every actuator that is interested in that type of data. Our experiments prove that our scheme is scalable and has high performance gains comparing to directed diffusion.

REFERENCES

[1] **Cayirci, E., Coplu, T., Emiroglu,** O., 2005. Power aware many to many routing in wireless sensor and actuator networks, *2nd European Workshop on Wireless Sensor Networks (EWSN)*, Istanbul, Turkey, February.

[2] **Alberts, D.S., Garska, J.J., and Stein F.P.**, 1999. Network Centric Warfare: Developing and Leveraging Information Superiority, Online Available: http://www.dodccrp.org/NCW/ncw.html.

[3] The cooperative engagement capability, 1995. Online available: http://techdigest.jhuapl.edu/td1604/APLteam.pdf.

[4] **Kumar, S. and Shepherd, D.**, 2001. SensIT: Sensor information technology for the war fighter, *Proceedings of the 4th International Conference on Information Fusion*, 1.

[5] **Corella J.**, 2003. A tactical automates security system (TASS): Air force expeditionary security, *SPIE Conference on Unattended Ground Sensor Technologies and Applications*, Orlondo, FL.

[6] **Kahn, J. M., Katz, R. H. and Pister, K. S. J.**, 1999. Mobile networking for smart dust, *Proceeding of ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, 271-278.

[7] **Chong, C., Kumar, S.P.**, 2003. Sensor networks: evolution, opportunities and challenges, *Proceedings of IEEE*, August, 91, 8,

[8] **Akyildiz, I. F., Su, W., Sankarasubrabramaniam, Y., Cayirci, E.**, 2002. A survey on sensor networks, *IEEE Communications Magazine*, August, 40, 102-114.

[9] **Sahasrabuddhe, L. Mukherjee, H., B.**, 2000. Multicast routing algorithms and protocols: a tutorial, *IEEE Network*, January/February, 14, 90-102.

[10] **Romer, K. and Mattern, F.**, 2004. The design space of wireless sensor networks, *IEEE Wireless Communications*, December.

[11] **Al-Karaki, J. N., Kamal, A. E.**, 2004. Routing techniques in wireless sensor networks: a survey, *IEEE Wireless Communications*, December.

[12] **Akyildiz, I. F. and Kasımoglu, I. H.**, 2004. Wireless sensor and actor networks: research challenges, *Elsevier Ad Hoc Networks*, March.

[13] **Akkaya, K. and Younis, M.**, 2004. A survey on Routing Protocols for Wireless Sensor Networks, *Elsevier Ad Hoc Networks*.

[14] **Jiang, Q. and Manivannan, D.**, 2004. Routing Protocols for Sensor Networks, *Proceedings of the IEEE Consumer Communications and Networking Conference (CCNC 2004)*, Las Vegas, Nevada, USA, January.

[15] **Heinzelman, W., Kulik, J., and Balakrishnan, H.**, 1999. Adaptive protocols for information dissemination in wireless sensor networks, *Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'99)*, Seattle, WA, August.

[16] **Intanagonwiwat, C., Govindan, R., and Estrin, D.,** 2000. Directed Diffusion: A scalable and robust communication paradigm for sensor networks, *ACM MobiCom'00*, 56-67.

[17] **Estrin, D.**, 1999. Next century challenges: scalable coordination in sensor networks, Proceedings of the 5th annual ACM/IEEE international conference on Mobile Computing and Networking (MobiCom'99), Seattle, WA, August.

[18] **Braginsky, D. and Estrin, D.,** 2002. Rumor routing algorithm for sensor networks, *ACM WSNA'02*, Atlanta.

[19] **Ye, F., Chen, A., Liu, S., and Zhang, L.**, 2001. A scalable solution to minimum cost forwarding in large sensor networks, *Proceeding of the 10th International Conference on Computer Communications and Networks*, 304-309.

[20] **Schurgers, C. and Srivastava, M.B.**, 2001. Energy efficient routing in wireless sensor networks, *MILCOM Proceedings on Communications for Network-Centric Operations: Creating the Information Force*, McLean, VA.

[21] **Chu, M., Haussecker, H., and Zhao, F.**, 2002. Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks, *The International Journal of High Performance Computing Applications*, August, 16, 3.

[22] **Yao, Y. and Gehrke, J.**, 2002. The cougar approach to in-network query processing in sensor networks, *SIGMOD Record*, September.

[23] **Sadagopan, N.**, 2003. The ACQUIRE mechanism for efficient querying in sensor networks, *Proceedings of the First International Workshop on Sensor Network Protocol and Applications*, Anchorage, Alaska, May.

[24] **Shah, R. and Rabaey, J.**, 2002. Energy aware routing for low energy ad hoc sensor networks, *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC),* Orlando, FL, March.

[25] **Servetto, S. and Barenechea, G.**, 2002. Constrained random walks on random graphs: routing algorithms for large scale wireless sensor networks, *Proceeding of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, Atlanta, Georgia, USA.

[26] **Heinzelman, W.R., Chandrakasan, A., Balakrishnan, H.,** 1999. Energy-efficient communication protocol for wireless microsensor networks, *IEEE JSAC*, 17, 8, August.

[27] **Lindsey, S. and Raghavendra, C.S.**, 2002. PEGASIS: Power efficient gathering in sensor information systems, *IEEE Aerospace Conference*, Montana.

[28] **Li, L. and Halpern, J.Y.**, 2001. Minimum energy mobile wireless networks, *Proceedings of IEEE International Conference on Communications (ICC'01)*.

[29] **Novakovic, D. and Dukic, M.**, 2000. Evolution of the power control techniques for DS-CDMA toward 3G wireless communication systems, *IEEE Communications Surveys & Tutorials*, 3, 4.

[30] **Park, S.-J. and Sivakumar, R.**, 2002. Load-sensitive transmission power control in wireless ad-hoc networks, *Proceeding of IEEE Global Communications Conference (GLOBECOM)*, November, 42- 46.

[31] **Kubisch, M., Karl, H., Wolisz, A., Zhong, L.C., Rabaey, J.**, 2003. Distributed algorithms for transmission power control in wireless sensor networks, *IEEE WCNC*, March, 1, 558 - 563.

[32] **Kawadia, V., Narayanaswamy, S., Rozovsky, R., Sreenivas, R. S.**, **Kumar, P. R.,** 2001. Protocols for media access control and power control in wireless networks, *Proceedings of the 40th IEEE Conference on Decision and Control*, December, 2, 1935 - 1940.

[33] **Schurgers, C. and Srivastava, M. B.,** 2004. Energy efficient routing in wireless sensor networks, *Proceedings of MILCOM*.

BIOGRAPHY

Özlem EMİROĞLU was born in İzmir in 1981. She graduated from İzmir Ege High School in 1999 and Dokuz Eylül University Department of Computer Engineering in 2003. She has been working as a software design engineer about telecommunication in research and development department in Nortel Networks NETAŞ since June, 2004.