A Reliable Event Response Framework for Wireless Sensor and Actor Networks

By

Mehwish Hassan

Submitted to the Department of Computer Engineering in partial fulfillment of the requirements for the degree of

MASTERS OF SCIENCE

IN

COMPUTER SOFTWARE ENGINEERING

Thesis Advisor

Dr. Ghalib A. Shah

College of Electrical and Mechanical Engineering

National University of Science and Technology

2010

A Reliable Event Response Framework for Wireless Sensor and Actor Networks

By

Mehwish Hassan

2007-NUST-MS PhD-CSE(E)-04

Submitted to the Department of Computer Engineering in partial fulfillment of the requirements for the degree of

MASTERS OF SCIENCE

IN

COMPUTER SOFTWARE ENGINEERING

Thesis Advisor

Dr. Ghalib A. Shah

College of Electrical and Mechanical Engineering

National University of Science and Technology

2010

بِسْمِ اللهِ الرَّحْمٰنِ الرَّحِيْمِ

'Read in the name of your Lord, who created, created man from a clot. Read! And your Lord is The Most Bountiful. (He who taught) the use of the pen taught man which he knew not.'

DECLARATION

We hereby declare that no portion of the work referred to in this Project Thesis has been submitted in support of an application for another degree or qualification of this of any other university or other institute of learning. If any act of plagiarism found, we are fully responsible for every disciplinary action taken against us depending upon the seriousness of the proven offence, even the cancellation of our degree.

Signature: ____________

Author's Name: Mehwish Hassan

COPYRIGHT STATEMENT

CERTIFICATE OF APPROVAL

Certificated that work in this thesis entitled:

A Reliable Event Response Framework for Wireless Sensor and Actor Networks

was carried out by Mehwish Hassan under my supervision and that in my opinion, it is adequate, in scope and quality, for the Degree of MS (Computer Software Engineering).

Approved by

Signature: ______________________

Supervisor Name: Ghalib Asadullah Shah

DEDICATION

Dedicated to my loving family, who mean the entire world to me.

ACKNOWLEDGMENTS

All praises are due to Almighty ALLAH, The Most Beneficent and The Most Merciful, Who gave me the courage to complete this project.

I express my profound gratitude to my respected supervisor Dr. Ghalib Asadullah Shah for giving me opportunity to work on this project and steering me through all the details. Moreover, the project task could not have been performed in the finest manner without the apt encouragement of my respected teachers.

I would also like to oblige Mohammad Ahsan for continuously helping us during the implementation phase and many other who have been with us in learning and overcoming the problems we faced in ns-2.

Mehwish Hassan

ABSTRACT

Reliable event delivery at MAC, routing and transport level has been explored to great extent to provide improved throughput and route resilience in wireless sensor networks. The evolution of wireless sensor and actor networks (WSAN) has posed an extra constraint in terms of reliable event response. The reliable response is defined as the in-time event delivery from sensor nodes to actors and timely action of actors against the sensors input. This might be required in an application that includes a diverse mixture of sensors for monitoring multiple events. In this paper, we propose a reliable event response (RER) framework for WSANs that addresses the issue of reliable action. RER implements two functions; reliable event delivery (RED) and prioritized event action scheduling. RED is based on a novel cluster-based multipath routing protocol. While, action reliability is ensured through a prioritized action scheduling based on earliest deadline shortest positioning (EDSP) algorithm. The results reveal that RER achieved its goal of action reliability which is not possible with simple scheduling first-in first-out (FIFO) scheduling algorithm.

TABLE OF CONTENTS

CHAPTER 4 - RELIABILITY IN SENSOR-ACTOR COORDINATION

CHAPTER 5 - EARLIEST DEADLINE NEAREST TARGET (EDNT) ALGORITHM

CHAPTER 6 - PERFORMANCE EVALUATION

CHAPTER 7 - CONCLUSION

LIST OF FIGURES

Chapter 1 - INTRODUCTION

Wireless sensor and actuator networks (WSANs) comprise a significant and stimulating new expertise with immense prospective for improving several recent applications as well as creating new innovative systems in areas for instance worldwide environmental monitoring, accuracy cultivation, home and assisted living medical care, smart buildings and cities, industrial mechanization, and several armed applications.

1.1 Wireless Sensor and Actor Networks

Normally, WSANs comprise of great numbers of negligible capacity sensing, computing, and communicating devices and various types of actuators illustrated in Figure 1. These devices function in intricate and loud real world, synchronized environments. Current and past studies have created many outstanding low level methods and protocols to assemble, transfer, and carry out sensor synthesis of this unprocessed data and respond to organize actions. Nonetheless, a lot of challenges remain.

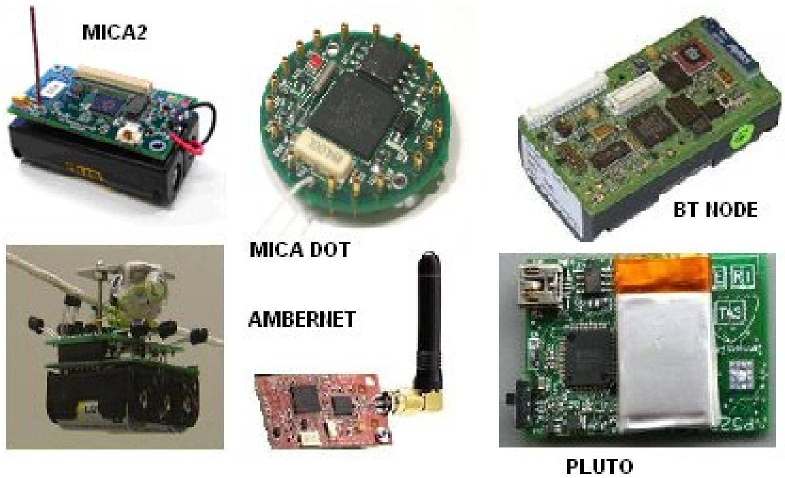

Figure 1: Sensor Nodes

1.2 Wireless Sensor Networks

Wireless Sensor Networks (WSNs) are generally characterized as passive monitor of the environment, which only collect the interested information. However, the realization for intelligent interaction with the environment has led to the development of a new class of network called Wireless Sensor Actor Networks (WSANs), which is capable of not only monitoring the environment but also performing the necessary action on environment [4]. This paradigm of heterogeneous network consists of resource constrained sensing nodes which sense and report the observed events and resource rich actuating nodes which collect the raw event reports, make an intelligent decision and perform action on the environment. Sensors are low-cost, low-power, multi-functional devices that correspond undeterred in short distances. Actors gather and workout sensor data and accordingly execute required actions on the surroundings. Most of the applications have actors which are resource rich devices equipped with processing abilities, greater transmission power, and prolonged battery life.

WSAN Operations

In WSANs, the mutual operation of the sensors helps the distributed sensing of a physical change in the surroundings. Following the sensing of any action requiring change by sensor nodes in the environment, the event data is then communicated to the actors, which collect, process, and ultimately reformed. On the contrary, this research is concerned with fresh applications in which actors are element of the sensor network and carry out actions depending on the information sensed by sensors. Prediction related to the WSANs is that it will soon be an essential element of systems such as battlefield surveillance, nuclear, biological or chemical attack detection, home automation, and environmental monitoring. For instance, applications responsible for detecting fire, sensors can transmit the precise source and strength of the fire to actors for water sprinkling that will extinguish the fire before it causes more destruction.

Furthermore, sensors can perceive detectable or assessable releases of contaminants in water or in the air, and actors can react to the addition and finally take required actions. In the same way, motion, audio, or light sensors in a building can sense the occurrence of interlopers and CCTV cameras to track them. On the other hand, movable actors can be sent to the region where the intruder has been sensed to get good quality images, quick or obstruct the intruder. Earthquake scenarios can be another example where sensors can assist situate survivors and direct actors to react and perform operations. Actors reform the event, based on partial information accessible at different actors, approximate the event uniqueness and recognize an area where event has occurred. Based on this, actors collaboratively decide which actors should be moved to the action area and at which speed. The coordinated mobility of actors is thus triggered by the happening of events. Actors continue receiving event data until the event is active, and numerous successive events activate following relocation of responsibilities between the actors.

1.3 WSN Applications

The actuation functionalities can be performed by a variety of devices as well as humans equipped with event specific action resources. In many real applications, robots are used as actor nodes. Low-flying helicopter platform is also developed that provides ground mapping, and air-to-ground cooperation of autonomous robotic vehicles. This platform may have more several actuation functionalities such as water sprinkling or disposing of a gas, which will make WSANs much more efficient. WSANs has found many applications in different domains like agriculture, industry, health, environment, civil structures, mining, home automation, military etc. This work is influenced by applications which are mission critical and thereby demanding a reliable event-specific response or mission accomplishment. We assume that actors are mobile devices with certain action range and they may need to move or position towards the target area to perform the action. For example fire brigades need to rush towards the place which has caught fire before it destroys everything, armed patrols are to target the suffering area, similarly robots are used to reach the places where humans can't reach in time etc.

1.4 Evolution of WSAN

The evolution of WSANs introduces many new challenges apart from the issues inherited from WSNs. The ultimate goal of deployment of WSANs for all kind of applications is to support reliable response in the environment. The reliable event response comprises of two phases; first, the reliable event report is provided to the actors and second, an actor must execute certain action or respond to the observed event before it becomes uncontrolled. In WSANs, the reliable event response has more comprehensive meaning that encompasses; an event must be sensed by a sensor node i.e. sensing coverage, the data must be delivered to the destination i.e. guaranteed delivery and an event report should be available to the actor/actor before its contents expire i.e. in-time delivery. While the second phase needs event data classification for multiple and simultaneous event reports to an actor along with some actor-actor coordination protocol to provide assurance of action according to event criticality or response time. Multi-path routing protocols [6], [7], [8], [9], [10], [11] have been explored to provide route resilience with improved throughput as compared with single-path routing in wireless sensor networks. The non-adaptive multi-path approach may degrade the network performance drastically due to excessive redundant traffic.

A hierarchical nodes configuration would better allow adopting the multipath redundancy at the intermediate nodes in accordance with networks conditions. Similarly real-time routing and coordination [5], [1] mechanisms are also presented which provides actor-actor coordination to enhance the reliability of event response. However, the event response according to the criticality of event or prioritized event response is not addressed in these protocols. In this paper, we propose a framework that deals with the problem of reliable event response (RER) in WSANs. The RER framework provides action reliability by using a reliable event delivery (RED) protocol and employing prioritized event scheduling. RED is based on cluster-based multipath routing in which the data redundancy is made dynamically to achieve the reliable delivery.

The redundancy factor directly corresponds to the packet losses and also a path is dedicated for shorter deadline traffic to make sure that the data is received in time and reliably. Once the event information is received by the actors, event reports are scheduled by using an earliest deadline shortest positioning (EDSP) algorithm that overrides the default firs-in-first-out (FIFO) scheduling to perform actions efficiently against the requests for which the deadline is shorter.

1.5 Challenges

Despite of the varied applications, sensor networks cause a number of exclusive practical challenges due to the following factors:

1.5.1 Vibrant variations

Sensor network is required to get adapted to the changes visible in connectivity for instance node failures, nodes addition etc, along with the environmental variations. Therefore, different traditional networks, in which the center of attention has been on improving the channel throughput or decreasing the process of node deployment, the key concern in a sensor network is to lengthen the system duration as well as the system strength

1.5.2 Deployment

The majority of sensor nodes are spread in regions which have no infrastructure in any way. A distinctive way of deployment in a forest would be throwing the sensor nodes from an airplane. In such circumstances, it is the nodes which are to recognize its connectivity and allotment.

1.5.3 Unattended process

Generally, one time organized, sensor networks have no human interference. For this reason the nodes are themselves accountable for reconfiguration in case of any modifications.

1.5.4 Energy conservation

The sensor nodes are not connected to any energy source. There is only a finite source of energy, which must be optimally used for processing and communication. An interesting fact is that communication dominates processing in energy consumption. Therefore, in order to make most favorable use of energy, communication between the nodes should be reduced in so far as possible.

The remainder of the paper is organized as follows. We summarize the recent works on QoS in wireless sensor networks in Section II. Section III describes the reliable event delivery (RED) approach based on cluster-based configuration. In Section IV, we present the proposed protocol for WSAN. Performance evaluation and results are considered in Section V. Finally, the paper is concluded in Section VI. Finally, the paper is concluded in Section VI.

Chapter 2 - RELATED WORK

Event reliability is addressed in different ways in literature. Some of these studies solve reliability through the coverage of the field [15] when nodes are densely deployed, while others [6], [7], [8], [9], [10], [11] solve it by information redundancy through multiple paths. The protocol in [6] has proposed a consistent transmission environment with little energy utilization, by proficiently utilizing the energy accessibility and the expected signal power of the nodes to recognize numerous routes to the target location. Simulation outcomes demonstrate that the energy efficient adaptive multipath routing method attains much better and efficient performance than the conventional routing protocols, even in the existence of high node density and overcomes concurrent packet forwarding. Furthermore, this protocol spreads the traffic over the nodes lying on different possible paths between the source and the actor, in proportion to their residual energy and received signal strength. Actor establishes these multiple paths by generating route request message that will pass through the entire network. Although this is done in reactive fashion but it seems an inefficient process to build paths for a highly dynamic nature of networks.

2.1 Hierarchical Multipath Routing

A hierarchical multipath routing protocol [7] can prove to be an efficient routing protocol for reliability in WSNs. Opportunistic Multi-Path Reliable Routing Protocol (OMR) [8] provides reliability through hop to hop acknowledgment and nodes keep the packets in their cache for some time in case of negative acknowledgment transmitted by the actor. The nodes are responsible for guaranteeing hop based reliability by coordinating with each other while blind flooding is avoided. Piggyback acknowledgement distributes the packet and acknowledgement data received between neighbors to moderate the effects of lossy and asymmetric links. An end-to-end reliability issue has also been studied in [2], [10]. In [10], the authors have described a protocol called ReInForM to maintain information consciousness in sensor networks. By means of ReInForM, data can be delivered at desired levels of reliability at proportional cost, in spite of the presence of considerable channel faults. It applies the perception of vibrant packet status in perspective of sensor networks to manage the number of routes necessary for the preferred reliability using only restricted information of channel inaccuracy rates and does not need any preceding calculation or repairs of these multiple paths. For consistent unit disk graphs, numerous edge-disjoint paths summing as many as the standard node degree, exist among any source and sink with very high prospect. These studies consider the need for information awareness and adapt to channel errors along with differentiated allocation of network resources based on the criticality of data. The reliability is still based on end-to-end service. Moreover, these works consider only the packet reliability and do not address the latency issue. In addition to the reliability and real-time requirements of event delivery, WSAN has some its own stringent requirements.

2.2 Reliable Event Response

The most important is the reliable event response in the event region. Event response reliability is addressed as action coverage in the literature [5], [1]. That is, if an actor is unable to execute action in some region then it coordinates with other actors to request response. However, actor action according to the criticality of event in multiple events scenario is not considered in these works. Hence, we require some prioritized event response to enhance the reliability of action in addition to the action coverage. Integrated cluster-based coordination and routing (CCR) framework for

WSAN proposed in [1], has the main intention to make certain reliable action with proficient energy utilization. In CCR, by keeping under consideration the amount of energy of nodes and the network density sensor nodes are first configured to form clusters. As the clusters are created, cluster-heads synchronize with the actors to structure interactive locations. The consistency of action has been addressed in requisites of in time information delivery to actors, in active areas and actor-actor synchronization to make sure that an actor carries out action in the desired region. A new real-time data aggregation (RDA) approach has also been integrated in the framework to preserve energy by reducing communication and achieve fairness.

In [2] this paper the authors have proposed a general reliability-centric framework for event delivery in WSANs. Three modules have been integrated which includes processing of an efficient and fault-tolerant event data aggregation algorithm, a delay-aware data transmission protocol, and an adaptive actuator allocation algorithm for randomly dispersed events. Adaptive replication algorithm has been used to handle node and link failures.

The research in this paper [3] has focused on designing a general energy efficient, fault tolerant, and highly reliable routing protocol extending the network life which is named as SBRR (Score Based Reliable Routing). The best quality path of the network has been selected to reduce packet loss and packet error. The routing choice depends on Path Score which is a heuristic parameter combination of four features. These features are applicable to hop count, energy level of sensors, error rate of links, and free buffer size of sensors for each path. Furthermore this algorithm uses a disjoint backup path for every source. The overall impact of this reduces the threat of data loss and delivery delay. This paper deals with the QoS-based routing in WSAN that achieves the QoS features required by the applications.

A new routing protocol called SEER: Secure and Energy- Efficient multipath Routing protocol has been proposed in [9]. SEER has focused on using multipath alternately for enhancing the lifetime of the network, as the path for communicating between two nodes. Alternatively, SEER is successfully resistive to some unambiguous attacks that have the quality of pulling all traffic using the malicious nodes by promoting a striking route to the target node. The performance of our protocol is compared to the Directed Diffusion protocol. In [15] the authors have formulated the issue as a construction problem to discover a topology that cover up the necessary sensing regions with high reliability. Implementing a high-quality topology is also helpful to organization of network and energy saving. An optimal coverage scheme for wireless sensor networks has been proposed that can keep up adequate sensing region and also provide high reliability and long system lifetime, which is the key design challenge in sensor networks implementation.

2.3 Clustering Protocol

The base routing protocol being used in wireless sensor and actor network is clustering protocol. A brief overview of clustering has been given in this section. The research focus is not clustering protocol but the implementation has purely been done.

2.3.1 Overview

Grouping sensor nodes into clusters has been widely pursued by the research community in order to achieve the network scalability objective. Every cluster has a leader, often referred to as cluster head (CH). Although many clustering algorithms have been proposed in the literature for ad-hoc networks, the objective was mainly to generate stable clusters. This was an even more important criterion in environments with mobile nodes.

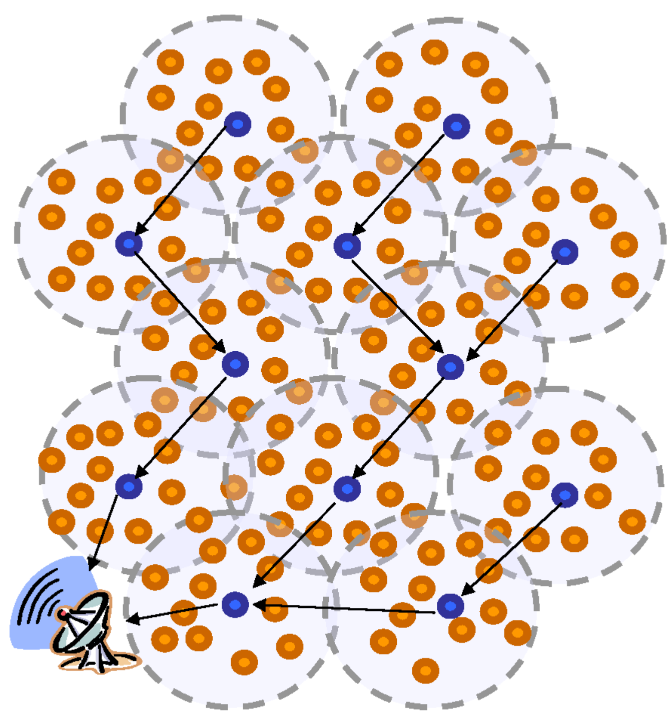

Figure 2: Clustering Protocol Managed by Cluster Heads

Many of such techniques care mostly about node reach ability and route stability, without much concern about critical design goals of WSNs such as network longevity and coverage. Recently, a number of clustering algorithms have been specifically designed for WSNs which are targeting network lifetime [24]. Clustering has numerous advantages. It can localize the route set up within the cluster and thus reduce the size of the routing table stored at the individual nodes. Clustering can also conserve communication bandwidth since it limits the scope of inter-cluster interactions to CHs and avoids redundant exchange of messages among sensor node. Clustering can stabilize the network topology at the level of sensors and thus cuts on topology maintenance overhead [24].

2.3.2 Clustering Objectives

Clustering aims differ from one WSN application to another. On the other hand, some most common aims of clustering are described below.

Load balancing: Constant allocation of sensors amongst the clusters is typically an objective for scenarios where CHs carry out data processing or important intra-cluster administration jobs. Given the duties of CHs, it is perceptive to stable the load among them so that they can meet up the expected performance goals. When CHs execute data aggregation, it is essential to have comparable number of node in the clusters so that the mutual data report becomes prepared almost at the same time for additional processing at the base-station or at the subsequent tier in the network [24].

Fault tolerance: In a lot of applications, WSNs will be prepared in insensitive environments and thus nodes are frequently uncovered to increased danger of break down and physical damage. Tolerating the failure of CHs is usually necessary in such applications in order to avoid the loss of important sensors' data. The most intuitive way to recover from a CH breakdown is to re-cluster the network. However, re-clustering is not only a resource weight on the nodes, it is often very disruptive to the on-going process. Consequently, modern fault-tolerance techniques would be more appropriate for that sake. Conveying backup CHs is the most notable system followed in the literature for recovery from a CH failure. The assortment of a support and the function such spare CH will play during standard network operation varies. When CHs have long radio range, neighboring CHs can adapt the sensors in the failing cluster. Rotating the function of CHs among nodes in the cluster can also be a means for fault-tolerance in totaling to their load balancing benefit [24].

Connectivity: When a number of of the sensors suppose the CH function, the connectivity purpose makes network clustering one of the many alternative of the connected dominating set difficulty. Conversely, when data latency is an apprehension, intra-cluster connectivity becomes a design objective or constraint [24].

Network lifetime: Given that sensor nodes are energy-constrained, the networks lifetime is a main apprehension; particularly for applications of WSNs in insensitive environments. When CHs are richer in resources than sensors, it is imperative to decrease the energy for intra-cluster communication [24].

2.4 Multipath Routing Protocols

Multipath routing implemented in any of the above discussed papers does not look into the on demand multipath clustering protocol. In this paper on demand multipath routing is implemented which is handled at cluster heads only and nodes are not responsible for maintaining path information which only increases the overhead. Furthermore, in time action reliability has not been discussed in previously implemented algorithms which in this paper have been focused to deal with the real time application. The framework achieves its goal by providing in-network services such as data redundancy according to the needs of reliability, real-time guarantees through some dedicated delay-constrained paths and request scheduling to achieve action fairness as well as action reliability.

The multi-hop and many-to-one character of data present in WSN indicates an evaluation of reliable multicast techniques proposed in other wired/wireless networks. There are numerous techniques present that have worked on the reliable transport and congestion control for the case of single correspondent and multiple receivers. Although the announcement structure of the overturn course, i.e., from actor to sources in WSAN, is an instance of multicast, it is not valid for the advance channel where several associated reports are sent to a single target. Comparable transport challenges with multiple sources and a single actor to respond to the event in other wired/wireless networks basically match to a multiple unicast. However, the WSN example requires the concept of cooperative reliability. For this reason, neither the reliable multicast nor unicast transport solutions can be applied in our case.

Chapter 3 - RELIABLE EVENT DELIVERY (RED)

The reliable event delivery (RED) approach is based on cluster-based configuration [1] of sensor nodes. Multiple connecting nodes of clusters are exploited to build multiple paths as shown in Figure 2. RED preserves low-delay paths for real-time traffic rather than implementing multiple queues for a single path [13]. These paths are managed only by the cluster-heads to ensure efficient path building process. In addition to delay information, congestion indicator I is also collected by the cluster-heads to approximate the reliability of the paths. Packets might be replicated through multiple paths to achieve the desired level of reliability.

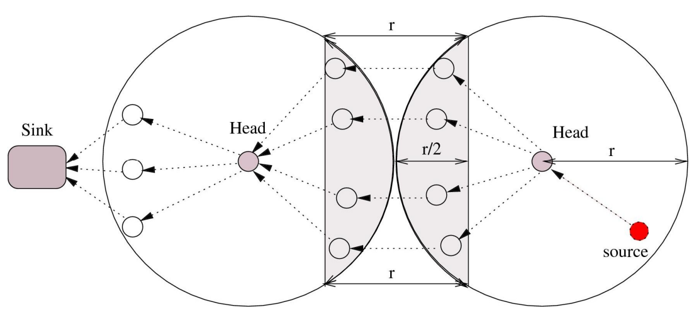

Figure 3: Multiple paths establishment from source to actor

3.1 How RED Works

The reporting sensor node to the destination actor connectivity semantic is decomposed into source-to-head, head-to-head and head-to-destination connectivity paradigm, which is more adaptive to wireless communication and, in particular, to sensor networks where the nodes failure rate is high. Packets always follow a single path from source node to its head. However, multiple paths are established from head-to-head and head-to-actor. We compute the number of possible head-to-head paths $P\begin{bmatrix} i \\ j \end{bmatrix}$ between head i and head j rather than computing end-to-end path from source to destination. The value of $P\begin{bmatrix} i \\ j \end{bmatrix}$ depends on the density of nodes. The shaded area of transmission range of cluster-head which is shown in Figure 2 reflects the number of such paths. Let A be the shaded area of the circle then we can evaluate the possible paths between two heads as follows.

$$P\begin{bmatrix} i \\ j \end{bmatrix} = \frac{\pi r^2}{R^2} n \times \frac{A}{\pi r^2} = \frac{A}{R^2} \times n$$

The value of A can be obtained by applying the following steps to compute the area between any two vertical chords at axial positions a and b as:

$$A = r^2 \left[\frac{\sin(2\omega_2) - \sin(2\omega_1)}{2} + (\omega_1 - \omega_2) \right]$$

where ω_1 and ω_2 can be found as follows:

$$\omega_1 = \cos^{-1}\left(\frac{a}{r}\right)$$

$$\omega_2 = \cos^{-1}\left(\frac{b}{r}\right)$$

We take $a = -r$ and $b = -r/2$ in our case as shown the shaded area of a circle in Figure 2. Initially the best effort traffic is relayed through single path but packets are replicated according to the achieved reliability through multiple paths. Hence, the normalized value of reliability η directly corresponds to the number of duplicate packets β through different paths. In case of network congestion, we gradually decrease the number of replicas until the congestion is eliminated regardless the value of η.

Suppose that P_l, α and σ are respectively the average length of a route from source nodes to actor, reporting rate of nodes, and the processing rate. We evaluate the total traffic X within the network. It is easy to find out that $n\alpha$ is the traffic in network for a single hop sources i.e. sensor nodes spend enough power to transmit data directly to actor/actors. However, with minimum transmission, packets are relayed through multi-hops. With an average route length from sensor nodes to actor of P_l and $P\begin{bmatrix} i \\ j \end{bmatrix}$ as the transmission of duplicate packets from cluster-heads to actor, the total traffic within the network is:

$$X = n\alpha(1 + P\begin{bmatrix} i \\ j \end{bmatrix}(P_l - 1))$$

The value of P_l can be computed as derived in [1]:

$$P_l = \frac{R}{r}(1 - p_{hops})$$

Where P_{hops} is the probability indicating the distance from source to the actor and is computed as.

$$p_{hops} = \frac{r}{R}\sum_{i=1}^{R/r} \frac{R^2 - (ir)^2}{M^2}$$

Hence, the load X_i on each node i can be obtained as X/n and we can evaluate the number of packets queued per node by using the Little Theorem [17] that is used as congestion indicator I

$$I = \frac{X_i}{\sigma - X_i}$$

3.2 States of Sensor Network

For the reliability measure, paths are selected according to the value of I, where $I > 1$ indicates congestion. Suppose Є represents the tolerance value in reliability then the value of β converges to $P\begin{bmatrix} i \\ j \end{bmatrix}$ when $\eta < 1$-Є. However, for $\eta >$ 1+Є, the value of β where β is the replication factor, is reduced until it approaches to either one (single packet) or desired reliability level (1-Є ≥ η ≤ 1+Є). On detection of congestion in the network, the value of β is also reduced until congestion is removed or a single copy of packet is relayed regardless the value of η.

We can identify the states of the network in the Figure 3.

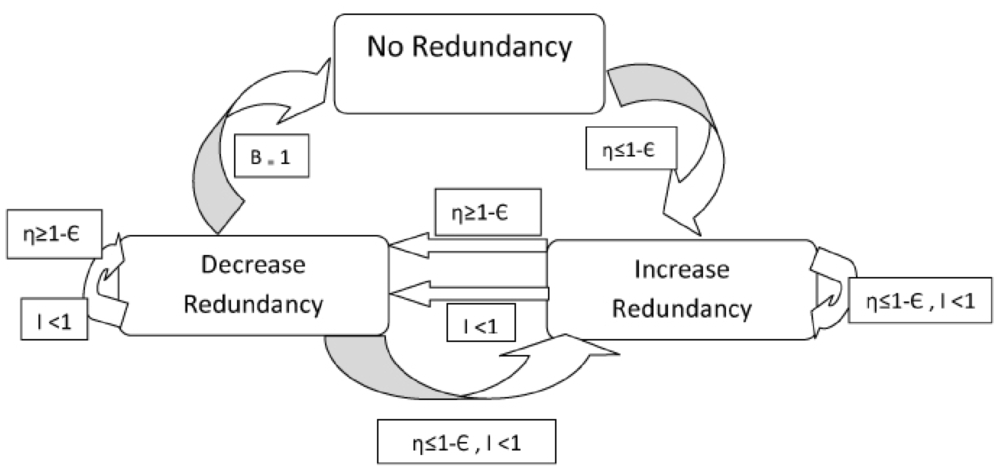

Figure 4: States of Sensor Network

3.3 Actor-driven Action Reliability

In WSANs, the roles of sensor and actor nodes are to collect data from the environment and perform appropriate actions based on this collected data, respectively. Sensors are energy-scarce devices and use minimum transmission energy to relay the event information to actors. As a result, sensors far from the actors require some routing protocol to convey the event readings. However, the routing protocol must be able to deliver the data packets by considering the expiration of event readings i.e. certain delay limit so that sensor data must still be valid when received by the actors.

On the other hand, if the nodes are close to the actors and can transmit their data directly to the actors then the routing protocol is unobligatory.

For example, if sensors detect a malicious person in a room and transmits this information to the disposer of tranquilizing gas actors, in order to avoid unnecessary actions on the environment that person should still be in the room when actors receive the sensor information.

The issue of real-time communication is very important in WSANs since actions are performed on the environment right after the sensing occurs. This implies that one of the main goals in WSANs is to decrease the communication delay between sensing and acting. A number of real-time routing protocols are proposed [2], [21] to ensure end-to-end packet deadlines. However, it is not yet certain that an actor performs some action against a request received within the time constraint from the sensor nodes. It might be due to the following reasons.

- Action rate is limited and actor is unable to perform certain action within the given response time for all the requests.
- Action range is limited and actor receives a request from the area that it cannot target.

In either of the above case, applications demand some assurance in order to take measures against the critical events. Although the communication delay is restricted to some extent but actors also take certain in making the decision and executing the relevant actions after receiving the requests. We focus on the first scenario that comes in the way of action reliability.

Chapter 4 - RELIABILITY IN SENSOR-ACTOR COORDINATION

Generally, the action completion time is ignored in addressing the response time. If an actor is busy in attending some event area then the subsequent requests during that time are queued. At higher event frequency, more data is generated and thereby more number of requests arrives at the actors. If the requests increase beyond the action rate of the actors then the waiting time in the queue also increases, delaying the response of actors. In such cases, sensors close to the actors are able to send their requests more quickly and may keep the actors busy if first-come-first-serve policy is adopted. Hence, we need to employ some order on the requests to avoid starvation of the farther nodes.

In some applications, actor needs to position itself in the direction and location of event area before initiating action that extends the action completion time. The actors' efficiency can be enhanced if we feed the sensors input to the actors such that the target positioning time is minimized. That is regardless of the order of the received requests; we order the requests according to the location of sources and their remaining time to achieve the response time as well as fairness.

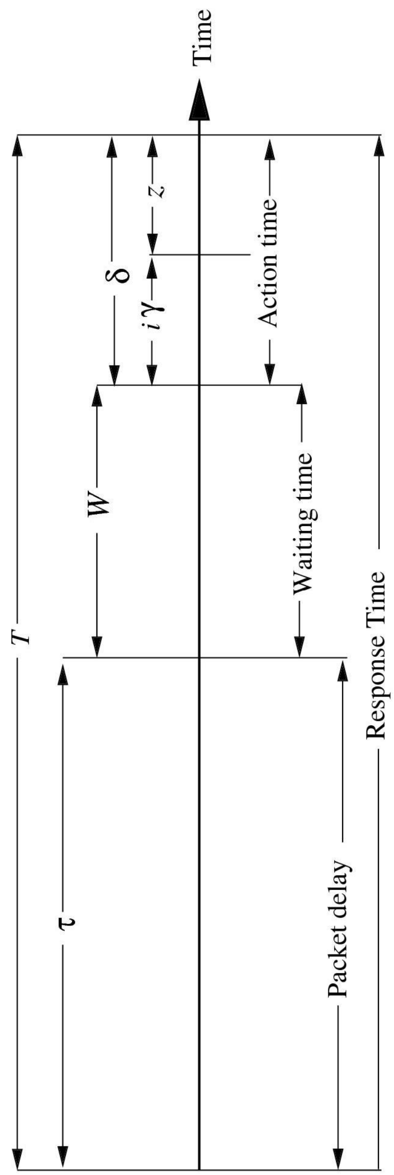

Figure 5: Delay factors from sensing event to executing action.

4.1 Delay factors

Let us denote the event response time, packet delay, request queue waiting time and action completion time as T, τ, W and δ respectively. We consider the response time as the time since an event occurs to the time when an action is completed as illustrated in Figure 4, the various factors during this process. The value of τ is measured from the occurrence of event to the time data packet received by one of the actor. The request waiting time is taken as the request arrival to the start of request processing. Similarly, the action completion time is the time since a request or input is fed to the actor to the time when the requesting source is targeted that includes the target settings (positioning according to the target area) as well as the action execution time. In order for in-time response of actor, the following condition should be met.

$$\tau + W + \delta \leq T$$

The value of τ depends on the underlying real-time routing protocol. However, we try to keep the values of W and δ such that the above constraint is not violated to provide action reliability. Actors make sure that one of the actors has responded to the request within application defined response time.

The action reliability is instrumented with NACK. If an actor receiving the source request is not able to react, it broadcasts NACK that triggers the actor-actor coordination to respond to that request. It needs certain assurance in both of the above cases to react to the sensors input.

4.2 Action Reliability

We deal with the action reliability by considering the actors as a group of servers that fulfil the requests from the sensors collectively. If an actor is unable to respond to some sources then it coordinates with the neighbouring actors to attend that source. The requests may arise from a number of sensors detecting some event and demanding certain allocation of resources from the actors. The resource allocation in WSAN is in fact the actor resources that execute some action corresponding to the detected phenomenon. One of the most important aspects of designing WSAN is allocating the actor resources in an efficient manner in order to overcome the conflicting demands. Moreover the requests from close nodes arrive earlier than the sensor nodes multi-hops farther. Intuitively, the requests from neighbouring nodes may unfairly capture the actor resources and makes the system unreliable in terms of action execution. Hence, the allocation must be fair to avoid starving the requests of far nodes.

Let C denote the capacity or action rate (average actions/second) of an actor. We consider an orderly situation in which waiting requests form a queue.

Furthermore, we let μ represent the average number of requests received per sensor node. Thus we see that the average number of seconds require per source is simply μ/C. We let λ denote the average number of sources reporting events to an actor.

If we assume first-come-first-serve queuing discipline, then we know that the utilization of such a resource is given by:

$$\rho = \lambda \frac{\upsilon}{C}$$

The average number of requests in the system is:

$$\acute{\mathbf{N}} = \frac{\rho}{1-\rho}$$

Similarly, the request waiting time is determined as:

$$W = \frac{\rho\upsilon}{1-\rho}$$

We employ an algorithm to control the values of W and δ in order to increase the responsiveness of the actors.

The objective is to minimize the number of unattended sources under the constraint of response time subject to minimum energy consumption of the actors. We assume that the energy consumed in executing action is much higher than communicating. Therefore by limiting the number of actions, the energy of the actors can be saved. Moreover to increase the number of attended sources with action fairness, we are interested in allocating the capacity C_a of actor ***a*** among all the sources equally. That is each sensor gets response from the actor for at least 1/ λ seconds. As a result the nearby nodes pumping the event readings quickly get the response from the actors only for the time fraction assigned to all the sources equally.

Now if $\frac{C_a}{\lambda\upsilon} < 1$ or $C_a < \acute{N}$

Then the actor is over-utilized and is unable to meet all the requests in a given time. We can overcome this scenario in two ways.

4.2.1 Sources Reporting Events at High Rate

First, if $C_a \geq \lambda$ then the actor can handle the high request rate locally. This situation arises when the request rate of individual sources is high but the number of sources is smaller than the action rate. It means that the sources are reporting events at rate greater than one. It is obvious that if two event readings from the same source are reported to the actors at different time then the most recent reading is valid while the previous is obsolete. Intuitively, it is unwise to take the action according to the information recorded in obsolete request. As a result, the actor ***a*** compacts the successive requests of individual source to single request if there exist already in the queue and triggers an action only once per source. In particular, as a request arrives to an actor, it checks for existing request of the same source. If it finds an entry of the source in request waiting queue then the event readings of the source are only updated but the queue entry time remains unchanged. By compacting the requests, we avoid the unnecessary actions to save the action resources and in turn enhance the efficiency of the actors. Hence, an actor can attend a large number of sources and the probability of responding to far nodes increases.

4.2.2 Actor-actor Coordination

Second, if $C_a < \lambda$ then the actor is unable to attend all the requesting sources and it triggers the actor-actor coordination procedure given in to handover some of the sources to appropriate underutilized actors. By appropriate actors we mean the actors which can handle the requests within given response time *T*.

The procedure of setting up data routes between sensors and actors is known as sensor- actor coordination. The events generated by the source are then sent to actor following this route. Once the event has been sensed, the actors synchronize to restructure it, to approximate its characteristics, and to formulate a mutual decision on how to execute the action. This procedure is referred to as actor-actor coordination.

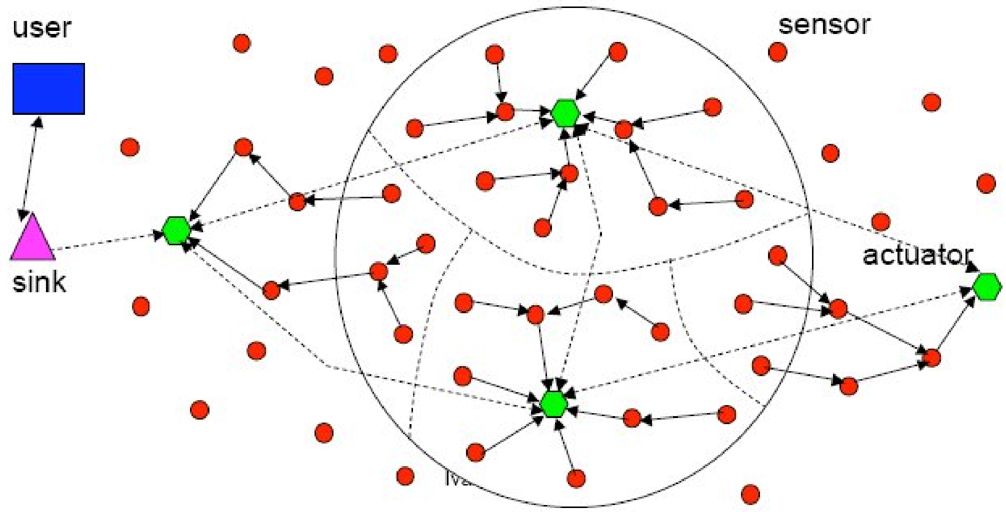

Figure 6: Actor Coordination

Consequently, the operation of a WSAN can be considered as an event-sensing, communication, resolution, and acting paradigm. Hence the actor-actor coordination predicament is defined as a commission obligation problem and a class of coordination problems in which the region to be acted upon requires to be optimally divided among diverse actors. The action workload is therefore divided among diverse potentially assorted actors, depending on the individuality of the event. In this case the cluster heads receiving information from sources recognizes a region of the event and the responsibility of the corresponding cluster head would now be to carry out required processing. However, the actor may not be proficient to act on the whole area that it is accountable for, given that the area may not be absolutely within the actor's action range. The action range classifies the spherical area wherever an actor is competent to act. Furthermore, the actor node may not be the finest actor for that job in conditions of action achievement time or energy utilization, where the previous is the time to carry out the action and the later is the necessary energy for the action. For these causes, actor-actor coordination is requested before instigating the action.

Chapter 5 - EARLIEST DEADLINE NEAREST TARGET (EDNT) ALGORITHM

Although we have reduced the action rate by eliminating the obsolete requests of sources but there are still no assurance that the timely arrived requests get in-time response. That is, a request arrives from source ***s*** such that:

$$T - \tau > \delta$$

Responding to ***s*** meets the timing constraint of the event reported by ***s***. However, if there are already queued ***i*** requests waiting for the response then the newly arrived requests has to wait for its turn if first-come-first-serve policy is followed. The source ***s*** has to wait for $\boldsymbol{i} \times \boldsymbol{\delta}$ seconds. Therefore, in case of:

$$\delta < T - \tau < i \times \delta$$

The timely arrived requests of ***s*** source remain unfulfilled. It is due to the fact that the far nodes get place at the end of queue while the requests of nearby nodes occupy the initial queue entries and thereby delay the response of far nodes even if for some nearby node placing request in the queue after ***g*** sources such that

$$\delta < T - \tau < g \times \delta$$

It is therefore indispensable to employ an order according to the requests deadlines on executing actions so that the response ratio is improved in addition to the fairness.

5.1 Action Completion Time

The action completion time has been defined as the sum of actor positioning time and action execution time. In this definition, the action execution time is assumed to be constant for all the requests of identical events. However, the positioning time varies for different sources according to their locations. The positioning time consists of adjusting the aim towards the target region. Additionally, it may also include the relocation time of actor if the source is not within the action range because the actor needs to move towards the event region sufficiently to take it under its coverage. Hence, the positioning time depends on the relative distance between the currently attended sources to the next source to be attended. If we somehow reduce the positioning time then the response time can be improved. This can be achieved if we order the requests according to shortest distance between the current source and next source.

Let us assume that the actor is currently attending source s_i at location S_i (x_i, y_i). After targeting the current location, it has to responds to source s_j at S_j (x_j, y_j). Let γ be the positioning delay per unit distance and z be the action execution time. Therefore the time δ required attending the source s_j after the source s_i:

$$\delta = \left| S_i - S_j \right| \times \gamma + z$$

5.2 Earliest Deadline Shortest Positioning (EDSP)

Based on the above considerations, we propose a request scheduling algorithm *earliest deadline shortest positioning (EDSP)* to enhance the action reliability in WSANs. Let r_i be the remaining response time of the action request of source s_i then we compute it as:

$$r_i = T - \tau_i - W_i$$

The algorithm maintains a queue sorted according to the deadlines. A window of size *m* is defined as the first *m* requests in the queue i.e. the window consists of *m* requests of smallest deadlines. While serving the requests picked in the window, the actor does not consider the arrivals of new requests during the service time of requests in windows. Moreover, we employ the relative distance of the sources in scheduling the requests in order to minimize the positioning time. The advantage of defining the window is to minimize the computation cost and it runs the scheduling algorithm after processing *m* requests rather than on arrival of each new request.

At a scheduling instance, it assigns each request a weight, say:

$$w_i = r_i d_i$$

This is the weight allocated to the requests in the window and selects one of the requests for service. Where d_i is the distance between the current source position and the source s_i. We refer to this quantity w_i as the priority of value associated with the request. However, if there is more than one request with the same priority then it selects the one with earliest deadline. It should be clear that for any specific request, its priority value varies at each scheduling instance, since d_i varies with respect to the positioning of actor for serving sources at different locations. Similarly r_i decreases with increase in queue waiting time W_i of request from source s_i. The requests are fed to the actor in the ascending order of their weights i.e. request of lower weight is scheduled among the requests in window.

Figure 5 shows the steps from request arrival to submitting it to the *action processing* component. Let d_1, d_2,...d_n be the distance of sources s_1, s_2,...s_n respectively from the actor responsible of actions in their region. We assume that $d_1 < d_2 < d_3 ... < d_{n-1} < d_n$ i.e. s_1 is closer to the actor than s_2 and therefore the delay of packet from these sources is relative to their distance from actor. That is packet delay τ_1 of source s_1 is smaller than the packet delay τ_2 of source s_2.

5.2.1 Compacting and Ordering

We order the requests in the ascending order of the remaining time (descending order of communication delay) and eliminate the multiple requests of the same source in the *compacting and ordering* component.

5.2.2 Scheduling

Once the requests are prioritized, the *scheduling* module selects the first *m* requests for $m < n$ and calculates their weight by considering their relative positions in addition to their deadlines and builds an action window.

5.2.3 Action Processing

The requests from the action queue are fed to the *action processing* component one by one until the window is free. The *scheduling* process prepares another window of requests by picking subset of *m* requests from the priority queue ordered according to their deadlines. Hence, the requests are processed by grouping the sources in the form of window such that their deadlines are meet as well as the actor positioning time is minimized.

The idea behind the above algorithm is that we want to give requests with smaller deadlines higher priorities so that they can receive service earlier. This is accomplished by assigning smaller values to their weights. On the other hand, when a request of large deadline is very close to the current actor positioning, it should get higher priority. Since there is positioning time γ in this case and we are assuming the value of γ dominates the execution time z, therefore these requests should be given higher priority.

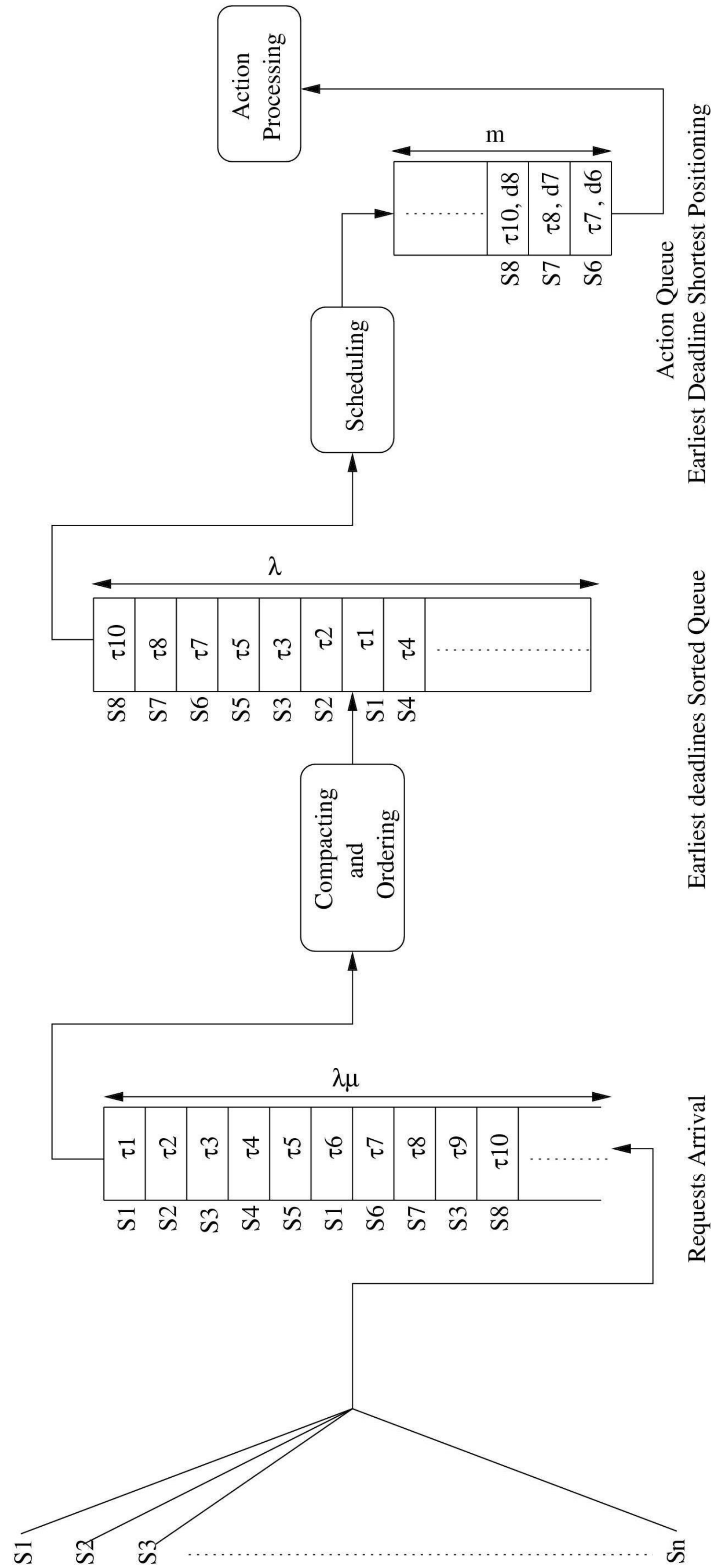

Figure 7: Scheduling actions on the basis of earliest deadlines shortest positioning

Chapter 6 - PERFORMANCE EVALUATION

To deal with the reliable and real time communication challenges existing in sensors and actors in WSANs, on demand multipath routing and EDSP have been presented in this paper. In this section, performance has been evaluated in comparison with single path routing and first in first out (FIFO) approach. To evaluate the performance of the presented techniques, we have developed an evaluation environment using ns-2. Different simulation scenarios have been considered including 400 × 600 field with different number of sources and actors. The main performance metrics for multipath routing is packet loss and end to end packet delay, for EDSP the execution time for the event generated and the distance covered by the actor to reach the source for carrying out the task. We have run a single simulation configuration several times in order to get an averaged point for each considered value on graph.

6.1 Throughput

The result for the multi path scenario implemented for the networks having higher error rate shows a better performance (approx. 10% to 20% higher) in case of large number of packets. Congestion in single path routing would cause frequent packet drop which causes a drop in throughput. As is demonstrated in Figure 8, with the increase in number of packets sent the throughput of multipath routing is increasing at a higher rate while throughput in single path routing increases at low rate.

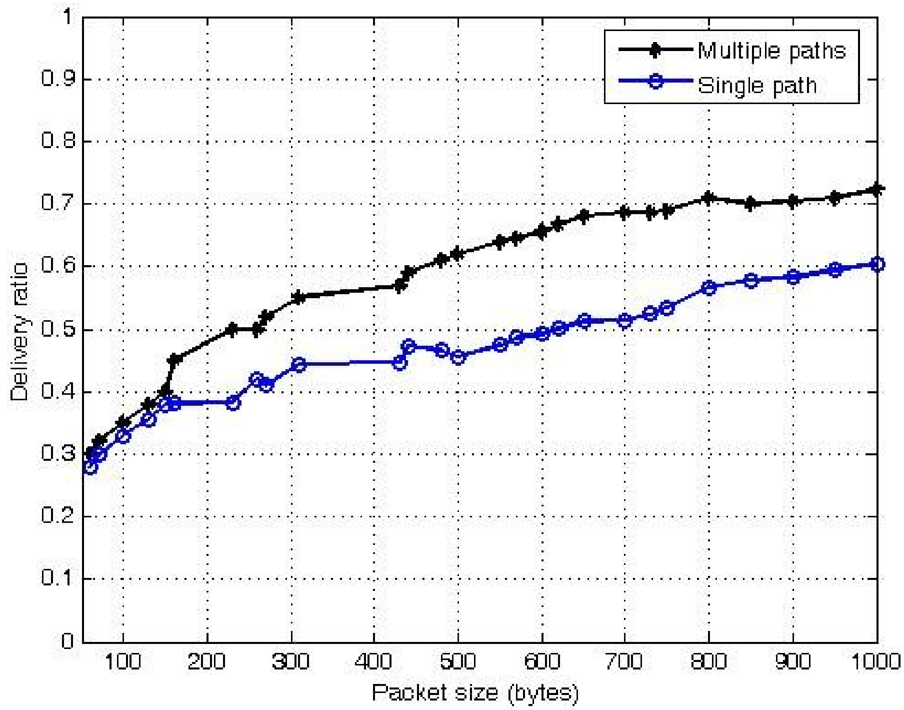

Figure 8: Throughput (Multipath vs Singlepath)

6.2 Average Packet Loss

As can be seen in Figure 8 the throughput of multipath routing is higher than single path routing, explaining that more packets would reach actor and more events would get served. In Figure 9 packet loss is graphed to see if number of packets generated per second is increased what is the effect on reliable and in time delivery of packets in singlepath and multipath routing. Average number of packets lost during multipath routing is 35% lower than singlepath routing as the packet if lost during its transmission, would reach the actor from the other route. Although this would increase the end to end delay but the probability for reliable packet delivery in multipath would remain higher.

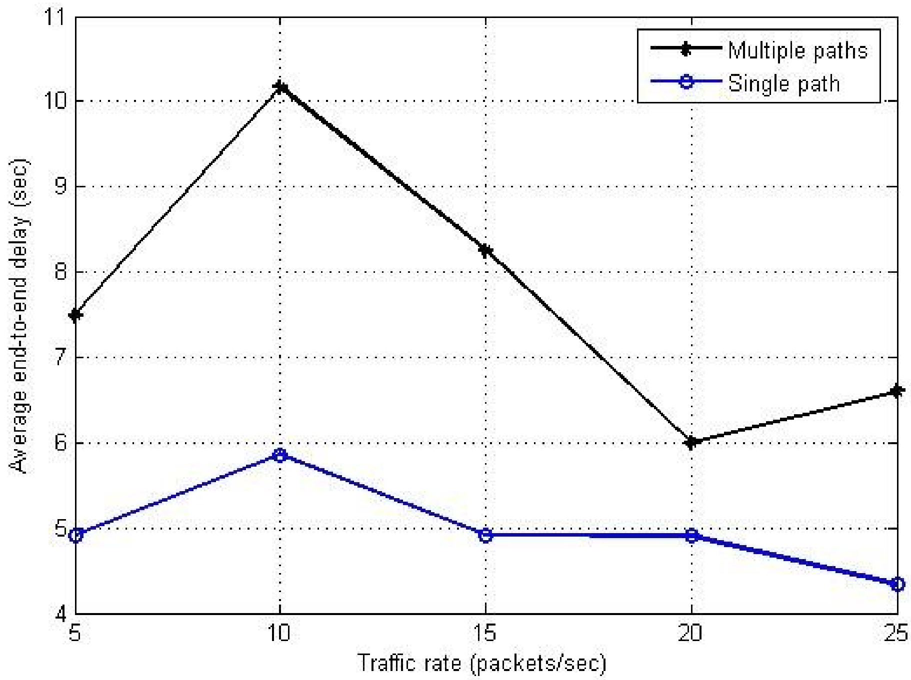

Figure 9: Averaged Packet Loss/simulation

6.3 Average End to End Packet Delay

End to end delay of each packet depends on the route it takes to reach the actor. At the same time BER (bit error rate) will also affect the reliable delivery of packets. In case of single path routing if the BER is increased, it will not guarantee the delivery of packets, hence the events whose packet is lost will not get served. In Figure 10 although the average end to end delay of multipath is 20% to 30% more than average end to end delays of single path routing, but it is guaranteeing the reliable event delivery to actor. So in multipath routing end to end delay is unnoticeably more than single path making no difference in overall network efficiency, making the network more reliable?

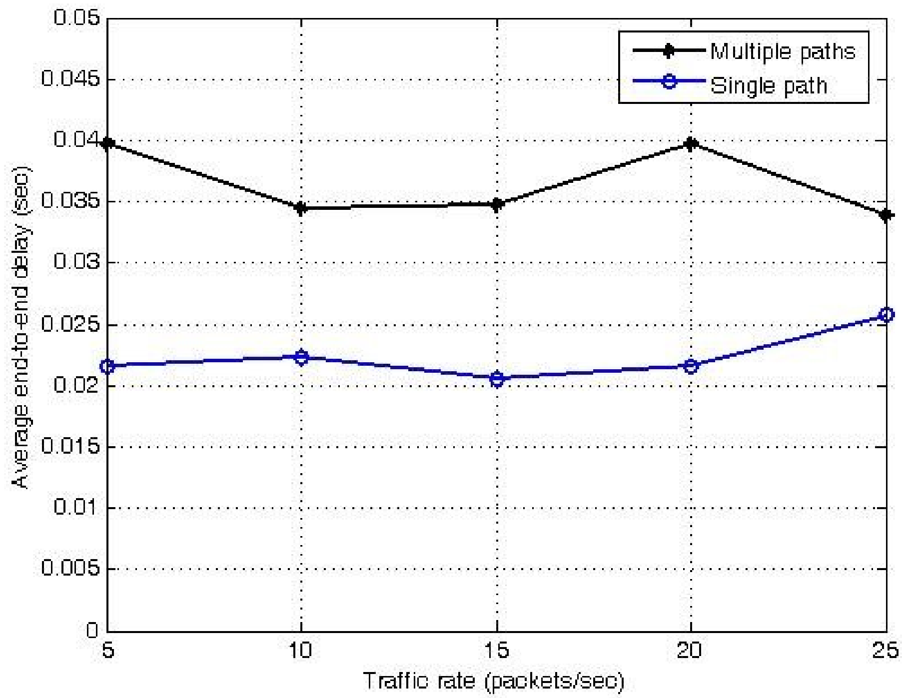

Figure 10: Average End to End Delay

6.4 Execution time with five Actors

The execution time has been evaluated with respect to the number of sources and the number of actors. The execution time for any request depends on the total number of events waiting in queue and also on the total number of actors present in the network. In Figure 11 we see that average execution time for EDSP is lower than the average execution time of FIFO where requests may end up in starvation. Considering different source nodes with total number of 5 actors present we can see that the same node is taking longer to be served in FIFO as compare to EDSP which clearly shows a better performance. There is a clear 40% to 50% decrease in execution time. We also see that more than 75% of the requests have been served in EDSP before the deadline and for FIFO 50% of the requests crossed the deadline which shows that EDSP is more reliable that the actor would take action before the event expires.

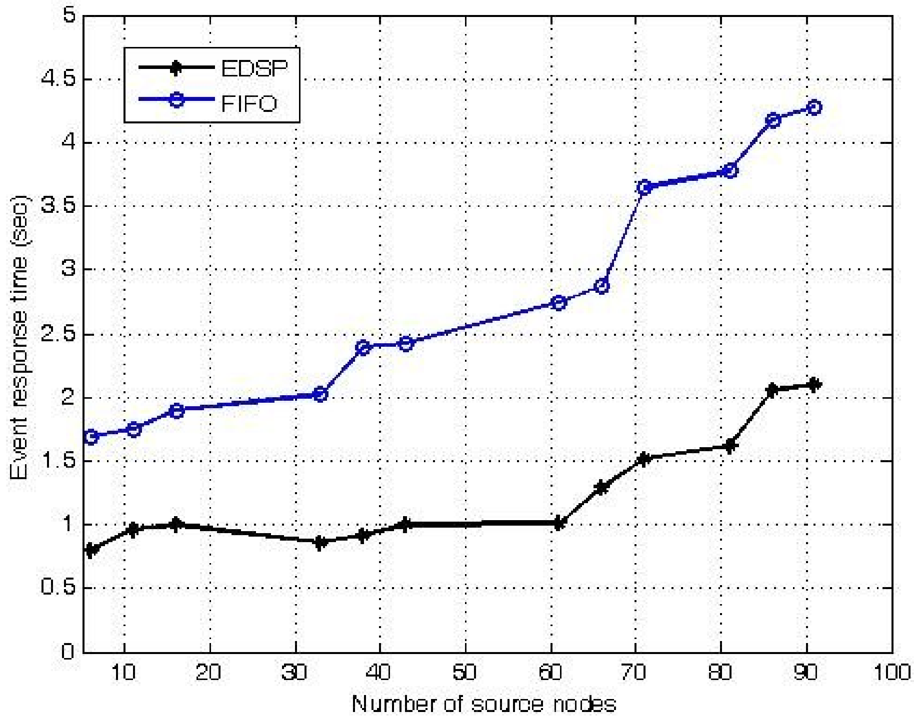

Figure 11: Execution Time w.r.t. five Actors

6.5 Execution time with ten Actors

Increasing the number of actors would provide opportunity to the sources to be served in time with lesser execution time as has been graphed in Figure 12. Analyzing the percentage decrease in execution time of request with respect to FIFO, there are still 20% to 30% sources which do not get served in time. If EDSP trend is showing 50% improvement and execution time is also dropped to 1.5 seconds at the maximum. While if we see execution time of requests with 5 actors the properly queued requests in EDSP are taking 2.2 seconds, still an increase of 30%.

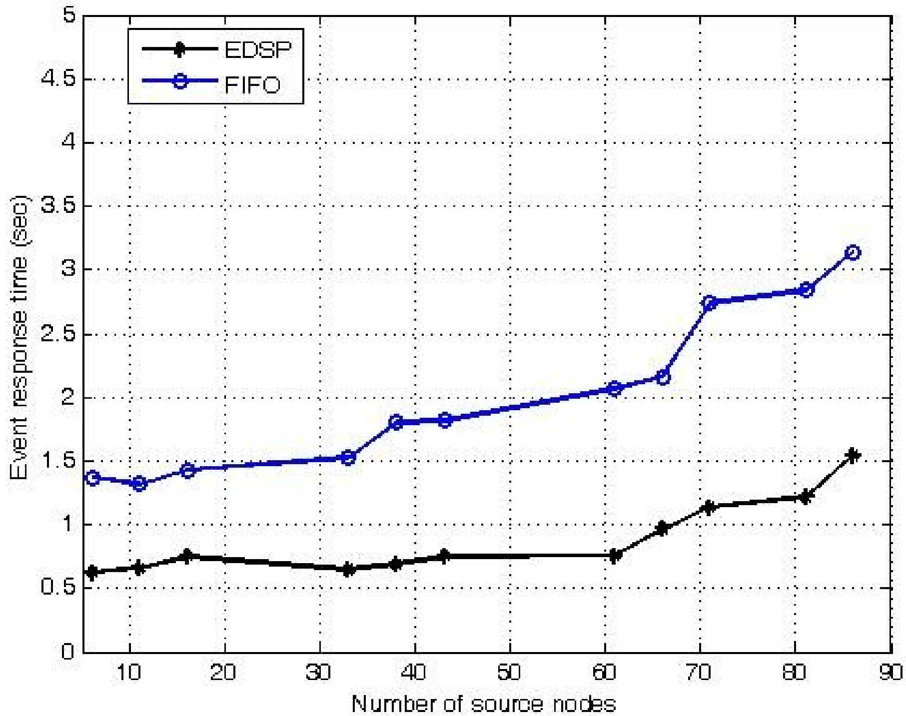

Figure 12: Execution Time w.r.t. ten Actors

6.6 Execution time with respect to Sources

If we look at the graph in Figure 13 for execution time with respect to sources and considering 2 to 3 actors present to serve all network requests, we see more than 50% decline in trend of EDSP as compared to the FIFO trend. The source which is taking more than 3 seconds to get served is now taking less than 1 second to be executed. Since the EDSP is following a proper procedure to order the requests. The deadline has been assigned so as to execute maximum of FIFO requests arriving but we can see that still the requests arriving at last of the queue exceeding the deadline.

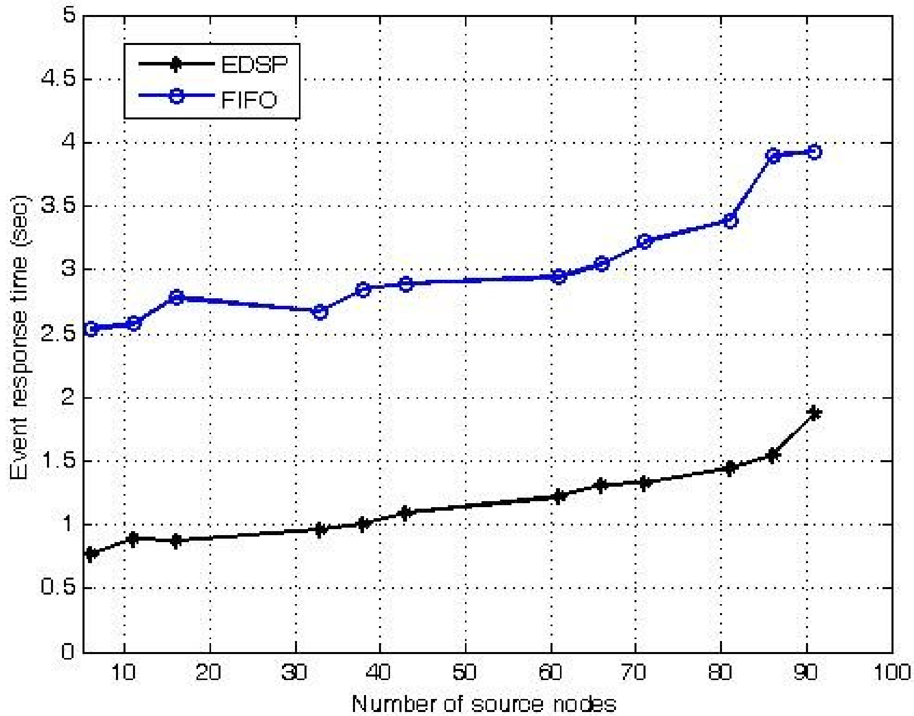

Figure 13: Execution Time w.r.t. ten Sources

6.7 Distance covered with respect to Actors

Actor has to move around to serve the events occurred. This distance covered is greatly affected by the order requests are arranged. If earliest deadline shortest positioning is considered, the total distance covered by the actor is in some cases more than 25% lesser than the distance covered by the actor in FIFO as can be seen in Figure 14. But as the number of actors is increasing the FIFO is approaching EDSP trend since the actors would be nearly placed and the distance to be covered by actors would be less.

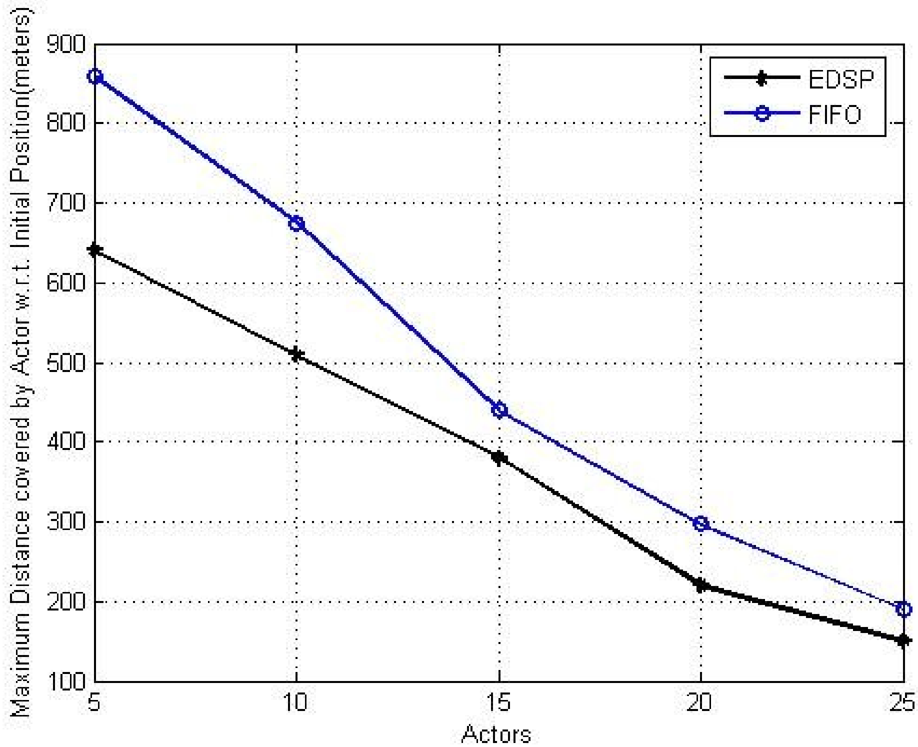

Figure 14: Total Distance Covered by the Actors w.r.t. Initial Position.

6.8 Distance covered with respect to Sources

The trends in Figure 15 show that if the sources increase, the actor has to cover more cumulative distance in FIFO as compare to EDSP. This would not only increase the execution time of events generated but also would consume more energy resource.

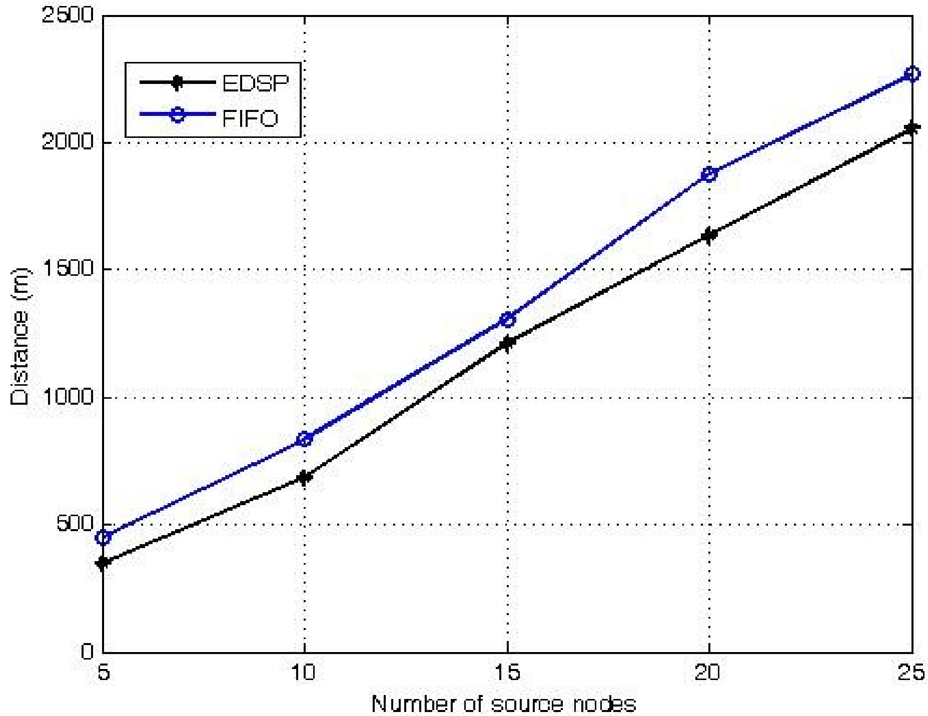

Figure 15: Distance Covered by the Actors w.r.t. Number of Source Nodes

Chapter 7 - CONCLUSION

This paper demonstrated the real time and reliable event delivery and response in WSAN. For the large scale sensor and actor networks where error rate may increase due to the number of requests, multipath routing would guarantee the request delivery to the actor making the presented framework reliable for request delivery. The ordering and scheduling algorithm manages the events received by the actor catering all the sensor network restrictions for example energy constraint, real time response of events etc. Performance evaluation computed using NS-2 simulation experiments shows that EDSP efficiently utilizes the energy at the actor end and gives the high performance in terms of reliable event response in WSANs.

REFERENCES

[1]. G. A. Shah, M. Bozyigit and F. B. Hussain, “Cluster-based Coordination and Routing in Wireless Sensor and Actor Networks, Wiley Journal of Wireless Communications & Mobile Computing, 2009.

[2]. Ngai,Yangfan Zhoub, Michael R. Lyub and Jiangchuan Liuc, “A delayaware Reliable Event Reporting framework for wireless sensor actuator networks ,” To appear in Ad Hoc Networks,2010.

[3]. Hamed Yousefi,Ali Dabirmoghaddam,Kambiz Mizanian, Amir Hossein Jahangir, “Score based reliable routing in wireless sensor networks,” Proceeding ICOIN’09 Proceedings of the 23rd international conference on Information Networking, 2009.

[4]. F. Akyildiz and I. H. Kasimoglu, “Wireless Sensor and Actor Networks: Research Challenges,” Ad Hoc Networks, Vol. 2, Issue 4, pp. 351-367, October 2004.

[5]. T. Melodia, D. Pompili, V. C. Gungor, I. F. Akyildiz, “Communication and Coordination in Wireless Sensor and Actor Networks,” IEEE Transactions on Mobile Computing, vol. 6, no. 10, 2008.

[6]. R. Vidhyapriya , P. T. Vanathi, “Energy Efficient Adaptive Multipath Routing for Wireless Sensor Networks”, International Journal of Computer Science, August 2007.

[7]. Y. Wang, C. Tsai and H. Mao, “HMRP: Hierarchy-Based Multipath Routing Protocol for Wireless Sensor Networks”, Journal of Science and Engineering, Vol. 9, No 3, pp. 255-264, 2006.

[8]. J. Du, H. Liu, P. Chen, “OMR: An Opportunistic Multi-Path Reliable Routing Protocol in Wireless Sensor Networks”, in Proc. of International Conference on Parallel Processing Workshops (ICPPW 2007).

[9]. N. Nasser, Y. Chen, “Secure Multipath Routing Protocol for Wireless Sensor Networks”, in Proc. of 27th International Conference on Distributed Computing Systems Workshops (ICDCSW’07).

[10]. B. Deb, S. Bhatnagar and B. Nath, “ReInForM: Reliable Information Forwarding using Multiple Paths in Sensor Networks,” In 28th Annual

IEEE Conference on Local Computer Networks (LCN 2003), Bonn, germany, October 2003.

[11]. Y. L. Jiming Chen, R. L. Wang, “A Reliable Routing Protocol Design for Wireless Sensor Networks”, in Proc. of IEEE MASS 2005.

[12]. E. Felemban, C.G. Lee, E. Ekici, R. Boder, and S. Vural, “Probabilistic QoS Guarantee in Reliability and Timeliness Domains in Wireless Sensor Networks,” in Proc. of IEEE INFOCOM 2005, Miami, FL, USA, March 2005.

[13]. K. Akkaya and M. Younis, “An Energy-aware QoS Routing Protocols for Wireless Sensor Networks,” In the Proc. of IEEE Workshop on Mobile and Wireless Networks (MWN2003), Providence, RI, May 2003.

[14]. T. He, J.A. Stankovic, C. Lu and T. Abdelzaher, “A Real-Time Routing Protocol for Sensor Networks,” Proc. IEEE International Conference on Distributed Computing Systems, May 2003.

[15]. H. Tian and H. Shen, “An Optimal Coverage Scheme for Wireless Sensor Network,”, in Proc. of ICN 2005.

[16]. J. Kay and J. Frolik, “Quality of Service Analysis and Control for Wireless Sensor Networks,” 1st IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS), October 2004.

[17]. P. P. Pham and S. Perreau, “Performance Analysis of Reactive Shortest Path and Multi-ath Routing Mechanism with Load Balance,” IEEE INFOCOM 2003.

[18]. Y. Sankarasubramaniam, O. B. Akan, and I. F. Akyildiz, “ESRT: Eventto-Sink Reliable Transport in Wireless Sensor Networks” MobiHoc’03, Annapolis, Maryland, USA. June 1-3, 2003.

[19]. F. Stann, and J. Heidemann, “RMST: Reliable Data Transport in Sensor Networks,” Proceedings of the First IEEE International Workshop on Sensor Network Protocols and Applications (SNPA’03), 11 May 2003.

[20]. S. Thrun, M. Diel, D. Hahnel, “Scan alignment and 3-D surface modeling with a helicopter platform,” in Proc. of the International Conference on Field and Service Robotic (FSR’03), July 2003.

[21]. V. C. Gungor, O. B. Akan, I. F. Akyildiz, “A Real-Time and Reliable Transport Protocol for Wireless Sensor and Actor Networks,” IEEE/ACM Transactions on Networking, Vol. 16,No. 2, April 2008.

[22]. Kashyap K. R. Kambhatla, S. Kumar, John D. Matyjas, "Reliable Multihop Remote Sensing and Control using Wireless Sensor and Actor Networks," IEEE, 2008.

[23]. UC Berkeley, LBL, USC/ISI and Xerox PARC, "The Network Simulator ns-2," The VINT Project, http://www.isi.edu/nsnam/ns/.

[24]. A. A. Abbasi and M. Younis, "A survey on clustering algorithms for wireless sensor networks," Comput. Commun., vol. 30, no. 14-15, pp. 2826–2841, 2007.