ISSN Print: 2223-5329
ISSN Online: 2226-809X

International Journal of Engineering and Technology Innovation

Vol. 6 No. 1 January, 2016



TAETI Taiwan Association of Engineering and Technology Innovation

IJETI
International Journal of Engineering and Technology Innovation
Frequency: Quarterly
ISSN Print: 2223-5329
ISSN Online: 2226-809X

Aims and Scope

International Journal of Engineering and Technology Innovation (IJETI), ISSN 2223-5329 (Print), ISSN 2226-809X (Online), is an international, multidiscipline, peer-reviewed scholarly journal, published quarterly for researchers, developers, technical managers, and educators in the field of engineering and technology innovation. Articles of original research, reports, reviews, and commentaries are welcomed by IJETI. The goal of this journal is to provide a platform for scientists and academicians all over the world to promote, share, and discuss various new issues and developments in all areas of engineering and technology innovation.

All manuscripts must be prepared in English, and are subject to a rigorous and fair peer-review process. Decisions will be made as rapidly as possible. Accepted papers will immediately appear online followed by printed hard copy. The journal publishes original papers including but not limited to the following fields:

- Automation Engineering
- Civil Engineering
- Control Engineering
- Electric Engineering
- Electronic Engineering
- Green Technology
- Information Engineering
- Mechanical Engineering
- Material Engineering
- Mechatronics and robotics Engineering
- Nanotechnology
- Optic Engineering
- Sport Science and Technology
- Innovation Management
- Other Engineering and Technology related topics

International Journal of Engineering and Technology Innovation

Instructions to Authors

Submission Policy

The author is responsible for obtaining any necessary copyright releases for material which has been published previously, prior to submitting material for consideration. No paper published in other journals or books can be considered for publication in the *International Journal of Engineering and Technology Innovation*. Concurrent submission to other publications and this Journal is viewed as a serious breach of ethics and if detected will result in immediate rejection of the submission.

Submission and Formats

Papers must be prepared in MS-WORD (doc) formats and submitted to the Editor in Chief of the journal by email to ijeti@nfu.edu.tw. The reference format (including the corresponding Endnote style), and the WORD template which adheres to the style of the IJETI can be found on the Author's Instruction page of the journal website.

In addition to being concise and consistent in style, spelling and the use of abbreviations, new submissions should conform to the following instructions.

1. Language: Papers should be written in English.
2. Units: The SI units should be followed for all dimensional quantities.
3. Typescript: Electronic manuscripts should be prepared in double-spaced format using a font size of 12 points.
4. Title and Author: Title of paper should be concise, informative and in capital letters. Author(s) and affiliation(s) should appear below the title.
5. Abstract and Keywords: Each paper should have an abstract not exceeding 150 words. Several keywords (not more than 4 words) should be given.
6. Headings: Headings for sections and subsections should start from the left-hand margin and numbered with Arabic numerals.
7. Equations and Mathematical Formulas: All equations and mathematical formulas should be typewritten. Equations should be numbered serially on the right-hand side by Arabic numerals in parentheses.
8. Length: The manuscript should be between 4000-8000 words (10-20 printed pages in the Journal), including title, authors and affiliations, abstract, text, figures, tables, and references.
9. Figures and Tables: All figures and tables should be numbered and have descriptive captions. Color figures will be printed in grayscale free of charge. However, if color print is requested by the author, there will be a charge of NT$3,000 or US$100 per color page.
10. References: References should be listed in numerical order according to their order of appearance. The numbered reference citation should be enclosed in brackets. Complete information should be given for each reference. For a journal or conference paper reference, the full title (instead of the abbreviation) of the journal or conference should be provided.

References Format

When referenced in the text, enclose the citation number in square brackets, for example [1]. The template will number citations consecutively within brackets [1]. Refer simply to the reference number, as in [2]—do not use "Ref. [2]" or "reference [2]" except at the beginning of a sentence: "Reference [23] was the first . . ."

Unless there are six authors or more, all authors' names should be given; do not use "et al.". Papers that have not been published, even if they have been submitted for publication, should be cited as “unpublished”. Papers that have been accepted for publication should be cited as “in press”. Capitalize only the first word in a paper title, except for proper nouns and element symbols.

The authors are recommended to use Endnote to generate the references of IJETI. The style of the IJETI references is the same as the Endnote style of IEEE, you can download it from the journal website. Examples of the references are illustrated in what follows:

1. Journal Article

W. H. Hsieh, "An experimental study on cam-controlled planetary gear trains," Mechanism and Machine Theory, vol. 42, pp. 513-525, May 2007.

2. Book

N. P. Chironis and N. Sclater, Mechanisms & mechanical devices sourcebook, 2nd ed. New York: McGraw-Hill, 1996.

3. Conference Proceedings

H. Goto, Y. Hasegawa, and M. Tanaka, “Efficient Scheduling Focusing on the Duality of MPL Representatives,” Proc. IEEE Symp. Computational Intelligence in Scheduling (SCIS 07), IEEE Press, Dec. 2007, pp. 57-64.

4. Patent

C. E. Larsen, R. Trip, and C.R. Johnson, Methods for procedures related to the electrophysiology of the heart, U.S. Patent, 5,529,067, Jun. 25, 1995.

5. Dissertation or thesis

J. O. Williams, “Narrow-band analyzer,” Ph.D. dissertation, Dept. Elect. Eng., Harvard Univ., Cambridge, MA, 1993.

6. In press

R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev., in press.

7. Unpublished work

K. Elissa, “Title of paper if known,” unpublished.

8. World Wide Web

F. Okumura and H. Takagi, “Maglev Guideway On the Yamanashi Test Line,” http://www.rtri.or.jp/rd/maglev2/okumura.html, October 24, 1998.

“AT&T Supplies First CDMA Cellular System in Indonesia,” http://www.att.com/press/1095/951011.nsa.html, Feb 5, 1996.

9. Technical Report

R. Cox and J. S. Turner, “Project Zeus: design of a broadband network and its application on a university campus,” Washington Univ., Dept. of Comp. Sci., Technical Report WUCS-91-45, July 30, 1991.

10. Handbook/date book, no author

Handbook for Writing Operation and Maintenance Manuals. Washington, D.C.: Packaging Machinery Manufacturers Institute, 1973.

Interface Circuits Data Book, Texas Instruments, Austin, Texas, 1993.
User’s Guide: Microsoft Word, Vers. 5.0, Microsoft, 1991.

11. Standards

[1] IEEE Criteria for Class IE Electric Systems, IEEE Standard 308, 1969.

Review

All manuscripts will be assigned to at least two qualified reviewer. Decisions will be made as rapidly as possible, and the journal strives to return reviewers’ comments to authors within 6 weeks. On receiving an article, the Editorial office will Pre-Screen your manuscript in 1 week. The officer will check whether the manuscript is valid and language is fluent and necessary factors are all included. Following is a formal review process accomplished by other official reviewers. It is decided by Editor-in-Chief or one of the Associate Editor whether your manuscript is accepted or not.

International Journal of Engineering and Technology Innovation

Web Site: http://sparc.nfu.edu.tw/~ijeti/
Email: ijeti@nfu.edu.tw

Publisher:

Taiwan Association of Engineering and Technology Innovation (TAETI)
Department of Automatic Engineering, National Formosa University
No. 64, Wenhwa Rd., Huwei, Yunlin, Taiwan
Tel: +886-5-6315368 Fax: +886-5-6314486; E-mail: taeti@nfu.edu.tw

Special Issue on Innovations and Applications in Wireless Ad Hoc and Sensor Networks

Preface

This is the Part 1 of the Special Issue on Innovations and Applications in Wireless Ad Hoc and Sensor Networks. There are 6 papers which are selected from the many submissions to this special issue. They cover many scientific and engineering challenges from both academia and industry, especially for the development of power consumption reduction, efficient and reliable Wireless Sensor Networks based on Internet of Thing (IoT). All of the papers submitted to this special issue subsequently underwent a rigorous peer-review by members of the Editorial Board of International Journal of Engineering and Technology Innovation (IJETI). Accordingly, acceptance for publication in this special issue was based on reviewer recommendations, mandatory revision, and final examination by the guest editor. It is also expected to offer some significant points which related to the innovations and applications in the field of wireless sensor networks.

I also wish to thank the many academicians, researchers and people who contributed their expertise to this important issue and for their efforts to bring this project fruition. In particular, I would deeply acknowledge the efforts of Professor Wen-Hsiang Hsieh, Editor-in-Chief of International Journal of Engineering and Technology Innovation (IJETI), for his kind support to the publication of this special issue.

Guest Editor
Dr. Jia-Shing Sheu
Associate Professor,
Department of Computer Science,
National Taipei University of Education,
Taiwan, ROC

International Journal of Engineering and Technology Innovation

Vol. 6, No. 1 January 2016

Special Issue on “Innovations and Applications in Wireless Ad Hoc and Sensor Networks”

CONTENTS

Intelligent Networks Data Fusion Web-based Services for Ad-hoc Integrated WSNs-RFID

Falah Alshahrany[1,*], Maysam Abbod [2], J. Alshahrani [3], Abdullah Alshahrani [4]

[1,2]Department of Electronic and Computer Engineering, Brunel University, London, UB8 3PH, United Kingdom

[3] School of Engineering and Sustainable Development, De Montfort University, Leicester, LE1 9BH, United Kingdom.

[4] School of Electronic, Electrical and Systems Engineering, Loughborough University, Loughborough, LE11 3TU, United Kingdom.

Received 19 November 2015; received in revised form 20 December 2015; accepted 27 December 2015

Abstract

The use of variety of data fusion tools and techniques for big data processing poses the problem of the data and information integration called data fusion having objectives which can differ from one application to another. The design of network data fusion systems aimed at meeting these objectives; need to take into account of the necessary synergy that can result from distributed data processing within the data networks and data centres, involving increased computation and communication. This papers reports on how this processing distribution is functionally structured as configurable integrated web-based support services, in the context of an ad-hoc wireless sensor network used for sensing and tracking, in the context of distributed detection based on complete observations to support real rime decision making. The interrelated functional and hardware RFID-WSN integration is an essential aspect of the data fusion framework that focuses on multi-sensor collaboration as an innovative approach to extend the heterogeneity of the devices and sensor nodes of ad-hoc networks generating a huge amount of heterogeneous soft and hard raw data. The deployment and configuration of these networks require data fusion processing that includes network and service management and enhances the performance and reliability of networks data fusion support systems providing intelligent capabilities for real-time control access and fire detection.

Keywords: data fusion, RFID-WSN integration, intelligent agents, dynamic multi-agent systems.

1. Introduction

Advances in telecommunications and information technologies, and their integration have enabled distributed sensing and tracking to solve a wide range of context aware environment problems that have common characteristics. Innovative fusion solutions for distributed detection that includes sensing and tracking, and inferring decision making have been designed in the light of new hardware and software developments supported by distributed networks data fusion support systems. These developments are based on the integration of a panoply of technologies which includes WSN, RFID, smart detectors, intelligent agents, web-based services, multi-agents distributed architectures, and hybrid intelligent decision support systems to translate sensing and identification, and tracking activities into web-based services [1]. Their implementation requires the support of multi-sensor data fusion functions for the capture of real time context environment data. These functions use methods and techniques developed in different areas such as artificial intelligence, neural networks, pattern recognition and statistical estimation.

*Corresponding author. E-mail address: falah88@hotmail.com

Problems inherent to the capture of real time context environment data have been addressed as sensor models and multi-sensor integration few decades ago [2]. The potential of sensor fusion has been examined in the context of innovative network architecture [3], and particularly in the context of data fusion in decentralized sensing networks [4], developing the concept of sensor collaboration [5] towards real-time information processing of sensor network data [6] that involves sensor node generic design and configuration [7], network configuration, deployment, planning, and management [8]. The configuration of these sensor networks has received a huge interest from researchers and practitioners to develop:

– integrated configurations of distributed sensing networks for cooperative sensing [8] and developing a flexibility that allows sensing devices to self-configure for a wide range of data fusion applications in dynamic environments [9], and
– frameworks for the modelling of data fusion establishing several fusion decision levels [10] and decentralised mobile sensor coordination based on adaptive sensor nodes clustering [11] required to appropriately support a wide range of data gathering tasks, and various data fusion functions.

Networks data fusion is presented in Section 2, defining data fusion and the problem, before presenting in Section 3, multi-sensor data fusion requiring a novel integrated fusion approach for big data, and examining data fusion requirements. Section 4 focuses on ad-hoc integrated RFID-WSNs data fusion support, defining generally an ad-hoc WSN architecture and presenting the three integration levels of RFID and WSNs, and a generic sensor node data fusion design implementation. Finally, a functional web-based service framework for data fusion solutions is proposed in Section 5 before presenting the conclusions and future work.

2. Networks data fusion

The main problem in networks data fusion is the shift from conventional detection relying on centralised incomplete observations processed at a central processor, to distributed detection based on complete observations processed by distributed processors created inside and outside the networks. These distributed processors form a distributed architecture that links several fusion centres in a detection network topology that supports in real time, a cooperative fusion processing and integrated decision making activities that extend data fusion to information fusion and knowledge discovery and extraction.

2.1 Data fusion

Data fusion translate raw sensor data into information required by domain context applications to discover the situation context and efficiently disseminate contextual information describing the situation, and support real-time decision making. It is an ongoing process that uses various methods, techniques and algorithms to combine different types of context aware data from distributed sources in networks and elsewhere, to perform inferences and drawing conclusions about the environment, in the same way humans develop their abilities to infer about what they observe and feel. The nature of the data combination depends greatly on:

– the type of application, situation and domain context,
– the different features generated from the analysis of the situation describing the interaction of objects identified by knowledge elements , and
– the level of fusion processing: object, situation and impact assessment, and process and cognitive refinement[12].

Data fusion support plays a key role when designing integrated solutions based on the use of ad-hoc WSNs supported by intelligent sensor-based systems. In these systems, data multi-sensor fusion integrates the collection of different types of data also called signatures, from different sources to observe a dynamic behaviour of a knowledge entity to identify fusion events in the context of distributed detection involving three main iterative functions: surveillance, intelligence and communications. A device may have different types of signature that reflect several situations analysed to determine the associated behaviour.

2.2 Problem definition

Solutions for secure real-time distributed detection, access control applications and real-time inferring decision making are supported by a generic hardware RFID-WSN integration that is enhanced by a flexible and adaptive functional integration extending a hardware integration and enabling new elaborated functionalities in the domain of application. This study examines the complexity of network data fusion in the domain of indoor sensing and tracking with a particular focus on the sensors and tags heterogeneity.

The sensors and tags heterogeneity is a major focus in this study and is concerned by the presence of heterogeneous nodes that have enhanced capabilities in terms of energy and communication capability, which both are required in a sensor network to increase the network reliability and lifetime. Heterogeneous nodes when adequately defined in the ad-hoc WSN can "triple the average delivery rate and provide a 5-fold increase in the lifetime (respectively) of a large battery-powered network of simple sensors" [13].

The context considered in this paper is the domain of sensing and tracking activities that are supported by real-time detection systems, based on the gathering from different sources of a huge amount of heterogeneous data made up of different types of soft and hard data provided by integrated devices containing sensors and RFID tags. These devices which may be of the same or different types, and have different strengths and weaknesses, are deployed simultaneously in distributed environments, and wirelessly connected in an ad-hoc network topology. Their concurrent operating enhances their combined performance. This study examines the data fusion support for the deployment and configuration of these devices integrated into an ad-hoc WSN.

2.3 Indoor sensing and tracking

Surveillance is an activity that includes outdoor and indoor sensing and tracking, and consists of monitoring environmental changes and a panoply of processes showing the behaviour of entities (people, goods, species and elements of the natural environment) interacting over a period of time in a defined space. Rapid advances in digital and communications technologies have made a wide range of theoretical capabilities practical with sensing, tracking and data processing, using hybrid communications based on network domains and mobile IP protocols.

The surveillance knowledge domain supports the sensing and tracking design in smart, safe, sustainable, and energy efficient buildings attended by people, where the determination of environmental conditions and presence in physical locations is a central problem in location-aware computing. This knowledge domain is analysed in the light of advanced technological hardware and software developments integrating WSNs and RFID, and their configuration and deployment. The case study developed in this research uses a symbolic location related to the virtual layout chosen to represent the indoor sensing premises. The symbolic location is based on absolute location systems which use coordinating system for locating, meeting the precision and accuracy of localisation requirements.

3. Multi-sensor data fusion

Multi-sensor systems generate data fusion that requires a novel integrated fusion approach for big data to solve multi-sensor data fusion problems that include raw data sensitivity, data fusion modelling, and fusion requirements.

3.1 Novel integrated fusion approach for big data

The data fusion approach takes into account the necessary decoupling of the service-based data applications processing from the raw data collection. The generic nature of data processing in the domain context applications consists of inferring decision making. This requires the elaboration of predictive distributions on data marts extracted in the context of data warehousing by selecting and estimating distinct collections of multidimensional discrete or continuous variables of interest in the domain context. This approach is also based on the integration of the real-time context aware data collected, in a forward modelling approach to evaluate, simulate and validate a wide range of plausible models related to the representation of the domain context entities, their behaviour and interactions.

3.2 Multi-sensor data fusion problems

The domain context examined in this research includes sensors and tags status and measurement accuracy, sensor nodes connectivity, location model, fire model, evacuation model, and other models. Data integration from multiple sources by the means of sensors and tags requires integrating new data expressed in different forms into historical, temporal and spatial contexts. The technical design of sensors, tags and sensor nodes, and the enhancement of their capabilities have a huge data impact on the specifications for their accuracy and the quality of their metrics.

3.2.1 Raw data sensitivity

The multi-sensor data fusion approach above detailed, includes the evaluation of the raw data sensitivity required to:

- Constraint the several models above mentioned,
- Reduce the data measurement errors,
- Increase the location accuracy, and
- Solve multi-sensor data fusion inherent problems.

3.2.2 Data fusion models

Multi-sensor data fusion problems are solved by context-aware computing large scale systems which rely on the use of location (symbolic or geometric) and logic-based models to:

- Discover and take advantage of contextual information and sensing context changes,
- Support both passive and active context awareness, and context triggered actions, and
- Enable an automatic contextual reconfiguration.

The nature of these large scale systems is hybrid in the sense that the fusion decision levels involve for the elaboration of intelligence information, different types of fusion, varying from a smart device to a software agent, all based on one or several models, interacting within a distributed framework.

3.2.3 Multi-sensor data fusion requirements:

The main multi-sensor data fusion requirements of for the design of these systems are:

- Group shared multi-level fusion activity by different fusion centres, considering the limited communication between sensor nodes, and the difficult balance ensuring a rational power supply and consumption while maintaining both an adequate desired coverage and connectivity in distributed ad-hoc wireless sensor networks.
- Different sensors and tags observing one or several common problems considering the context sensed in which the results of the different sensing and tracking tasks can conflict with each other, creating ambiguity in the data aggregation processes to perform and the data to store.
- Ad-hoc integrated WSN-RFID networks integrate three levels of sensor intelligence (fixed actions, actions adapted and data integration with work adjustment) which may be used in three different environments (designed world, real world and hostile world), facing variations of the detection network (depending of the different dynamic sensor node

configurations that might impose new fusion processing constraints) requiring the integration of the generated heterogeneous raw data representing different levels of data fusion.

– Distributed fusion poses the complexity of supporting a detection network topology due to the variations of the parallel fusion network when global inference in the form of feedback from distributed fusion centres is needed, and combined decision making involving individual sensor nodes that can use specific decision criteria depending on the tasks they perform, is needed.

Multi-sensor data fusion processing is very domain context dependant, and its design is constructed around the data network technical spectrum that includes in the context of real-time detection, false alarms from sensor nodes, dead sensors or tags, disconnected nodes, and also hostile actions resulting from intrusion events and other network treats. These events can be problematic for the network, persistent, and preponderant in nature and importance.

4. Ad-Hoc Integrated RFID-WSNs data fusion support

The research work presented in this paper takes into account the fact that integrating WSN and RFID technologies can be a complex design process, mainly when supporting real-time applications, which involves both hardware and software integration. The study has examined severe constraints imposed on the sensing, storage, processing, and communication features of the sensor nodes, in the context of designing a flexible and adaptive ad-hoc WSNs configuration solution based on the use of hybrid intelligent web-based support systems. The reconfiguration of WSNs which aims at enhance the network deployment and increase its performance, is needed when the sensor nodes may become faulty due to improper hardware functioning and/or lack energy supply (dead or low battery power).

4.1 Ad-hoc WSN architecture

The proposed Ad-hoc WSN architecture includes homogeneous and heterogeneous devices wirelessly connected. Due to the devices differences, their integration requires a high-level of data modularity and adaptability in the distributed multi-agent system architecture. Homogeneous devices are smart fire detectors composed of a several sensor and nodes as developed in the hardware integration description of the next section. Heterogeneous devices are IP smart devices which are deployed to enhance the deployment of homogeneous devices. They include RFID reader, IP Camera, Sprinkler, Message and Sign Displayer, Opening and Closing controller, People Counter, RFID reader, and other smart devices.

4.2 RFID-WSN integration

An extensive literature has been devoted to the hardware integration of RFID and WSN, and our research interest includes:

– the mode of functional and hardware integrations and the resulting data fusion[14],
– the allocation of specific tasks to RFID and WSN devices [15],
– the classification of both RFID and WSN devices to create similarity classes of their deployment attributes[16], and
– the use of low-level programming knowledge to adapt successfully RFID devices to ad-hoc WSN.

4.2.1 Functional integration

The functional RFID-WSN integration adopted in the proposed study, can be configured in six different ways [17], as summarized in Table 1. Although a hardware solution have been proposed in the next section, to show the adaptation of the integrated WSN-RFID in a context aware mode for new extended sensing capabilities by integrating RFID tags and readers in WSN sensor nodes, the conceptual design framework developed in this study has addressed the devices interoperability.

Table 1 Functional RFID-WSN integration

	Object and/or Person				Environment Sensing
	Attached	Sensing	Identification	Localised	
WSN	1	2	4,6	5	3
RFID	1,2,3		4,5	6	

Data fusion processing integrates the enabling of dynamic smart nodes with automatic reconfiguration information generated by dynamic control requirements imposed by self turning distributed multi-agent systems. Of great interest in the study of the RFID-WSNs integration problems, is the needed knowledge-based validation of integrated configurations when supporting multi-sensor heterogeneous data fusion. Heterogeneous data includes soft and hard data processed by data fusion techniques, prior to information and knowledge fusion.

4.2.2 Hardware integration:

The Hardware configuration of the integrated RFID-WSN sensor node is based on the functional integration resulting from the tasks requirements of the domain applications, as illustrated in Fig. 1.

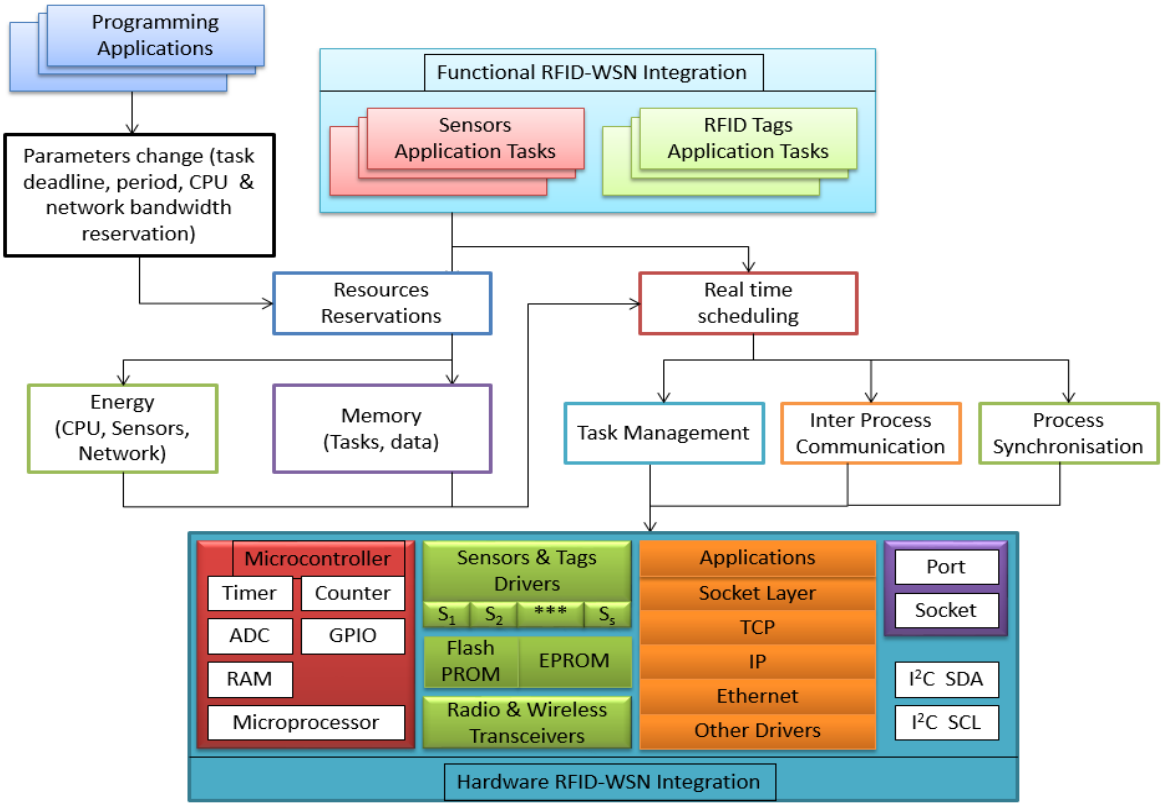


Fig. 1 Hardware RFID-WSN integration

The hardware development of the generic sensor node is not the focus of the paper. However, the proposed principles are to be considered in real world implementation:

– An RFID node is integrated in the sensor node that reads data from tags attached to people or goods.
– RFID readers placed in the read data from tags contained in fix sensors nodes.
– RFID nodes and readers contain the required components and functions to support a communication interface (I^2C interface for example) to send and receive parallel data through a serial data line (SDA) and another bus line for the serial clock line (SCL). These functions which include elementary commands (Initialisation, Read and Write) are pre-programmed, can be re-programmed in the context of their configuration and deployment.

4.2.3 Software integration

The communication interface I^2C assists the embedded software supporting the generic sensor node, or the intelligent fusion system supporting the network to detect in real-time tag IDs, and verify their existence and status in the network database. The proposed integration configuration is supported by distributed generic sensor node control and database control functions which are performed simultaneously and integrated in the network data fusion processing. The data fusion process includes the monitoring and reduction of RFID reading and/or identification errors, to provide a robust way to effectively enhance the RFID and WSN integration. Although it has been suggested that the hardware integration needs to be completed before configuring and programming the generic sensor node, the network data fusion process explores at the maximum the sensor node hardware configurability that procures a multitude of different hardware configurations needed by a wide variety of distributed detection and tracking applications. Based on distributed services composing the service layout which is decoupled from the network data fusion processing supported by Intelligent Fusion Support Systems (IFSS), the system architecture shown in Fig. 2, integrates the exchange support to the middleware, coordination and communication software layers in the data fusion system.

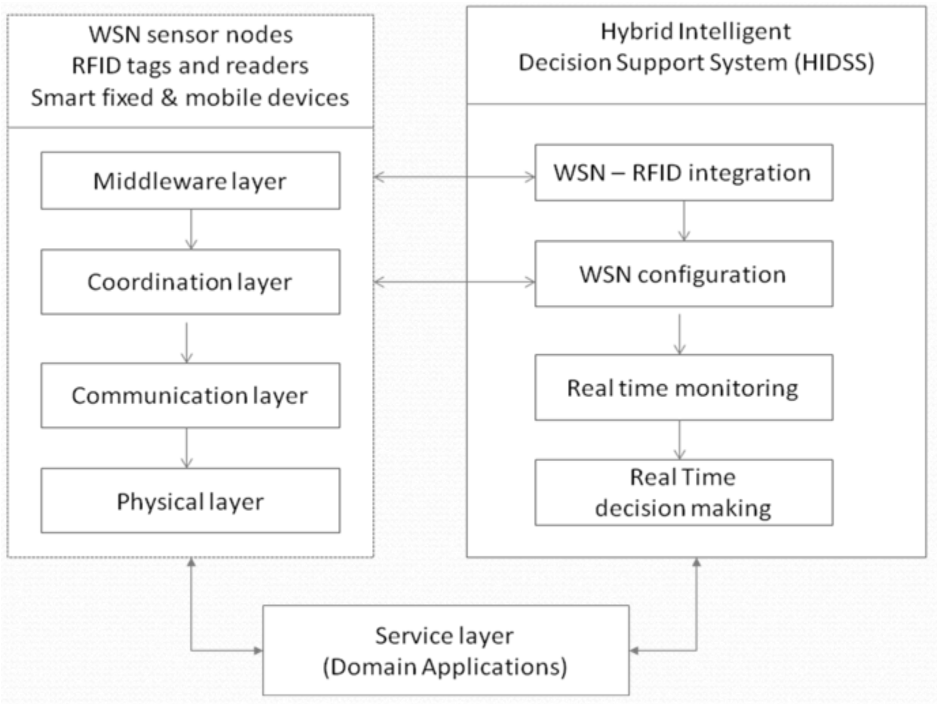


Fig. 2 Software integration for intelligent fusion support systems

4.3 Generic sensor node design implementation

Homogeneous devices which are generically designed using a knowledge-based design support are smart sensor nodes that are formed by grouping a variety of sensors and RFID tags, wirelessly connected in an ad-hoc network topology. An example of implementation is illustrated in Fig. 3. The modelling and publication of sensor data and their contexts of use consist of using of a data representation which is based on the sensor data being annotated with semantic metadata with the aim of increasing interoperability for sensors and sensing systems, and providing contextual information essential for situational knowledge[18] . This representation is supported by the sensor semantic network ontology (SSN), defining data encodings and web services to store and access sensor-related data [19].

4.3.1 Sensor semantic network ontology

The SSN ontology is a solution elaborated to describe the WSN sensors, their data and their contexts of use. This description allows autonomous or semi-autonomous knowledge agents associated to the deployment of these sensors to assist in deploying, configuring, collecting, processing, reasoning about, and acting on sensors and their observations. It describes sensors in their main characteristics which include their capabilities, measurement processes, observations, and deployments. The sensing mission matching is a sensor domain feature which consists of breaking a monitoring mission down into a collection of sensing and monitoring operations**,** each of which is broken down further into a collection of distinct elementary measurement, control and coordination tasks. Each task has specific capability requirements that enable the accurate measurement of the feature of interest in the task.

4.3.2 Sensor and RFID tags mission matching

The sensor mission matching is a requirement engineering process that supports the measurement requirements and capabilities association, validating in a second step the selection of existing sensor nodes during the individual tasks composition, or suggesting the design of specific sensor nodes. The individual tasks composition results in the identification of a group of capabilities already present in existing sensor nodes, or these grouped capabilities are a candidate for the design of new sensor nodes, as illustrated in Fig. 3 and implemented in Fig. 4.

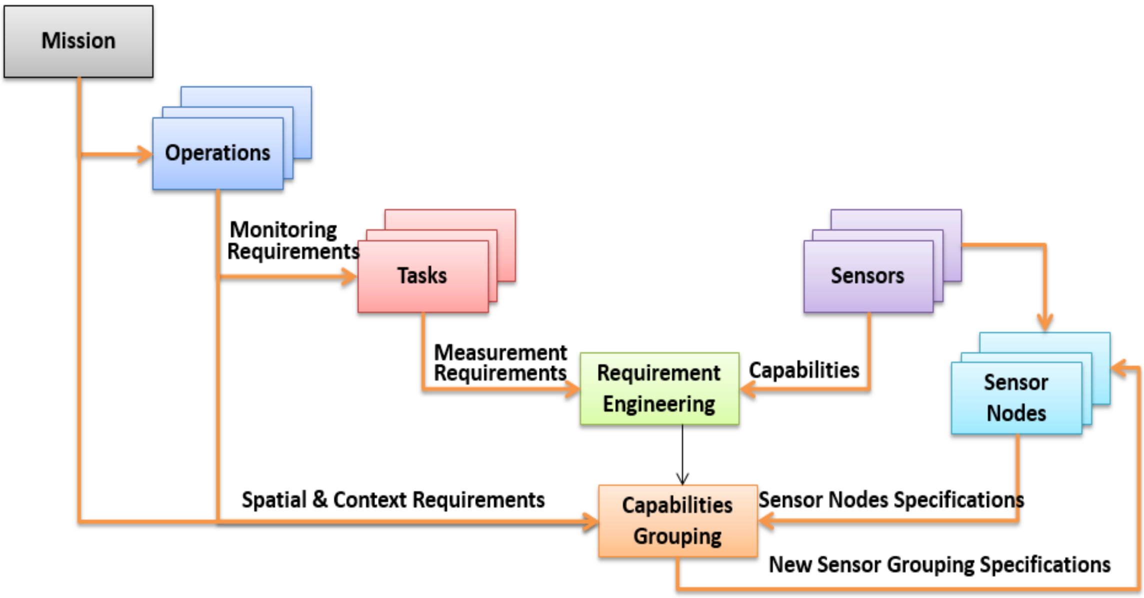


Fig. 3 Sensor capability task matching

Sensor nodes can be fixed or mobile. They are self deployed, and their sensors can be:

– Duplicated to enable the inclusion of a safety factor,
– Of different:
  - types, depending of the tasks they perform
  - configurations & sensing distances depending of the sensing requirements and their conditions of use, and
– Active or Passive.

Sensor nodes act independently, processing and/or transmitting the sensed data to a base station or another sensor node (multi-hop communication), for further processing and aggregation. Multi-hop communication is of great advantage for in-network distributed processing. At the implementation level, a software consisted of a program module controls the microprocessors of the nodes, and a RFID table is created in the WSN database to record the tag IDs of devices attached to, or of people wearing them.

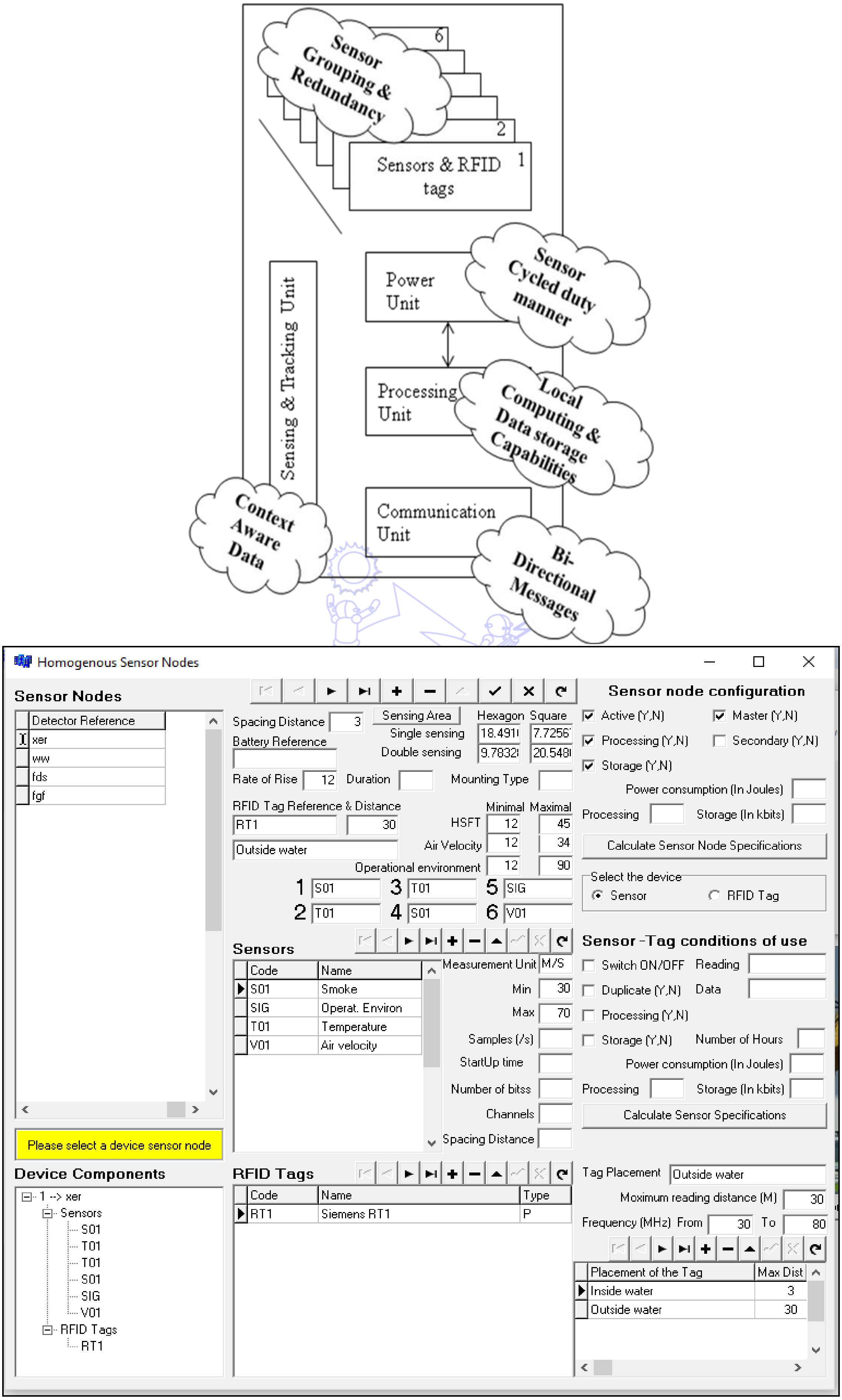


Fig. 4 Knowledge based support for generic sensor node design

4.4 Homogeneous & Heterogeneous devices location

Homogeneous and heterogeneous devices location requires a priori planning to ensure an optimal spatial observation and detection coverage. In this priori planning, the node location is a unique point calculated using a localized algorithm based on the use of a specific knowledge procedure depending on the type of sensor node needed to perform the required task, such as for example: sensing for fire detection, human presence and tracking, recording a scene, and sprinkling. The exact node location as accurately determined by the appropriate node localization algorithm detailed in the following sections, enables the improvement of the data fusion performance [20]. Fig. 5 illustrates for a "fire detection and people and goods tracking" application, the location of the physical network devices composed of homogeneous & heterogeneous sensor nodes wirelessly connected. Homogeneous devices have been distributed using a hexagon geometric patterns, whereas heterogeneous are placed using rules of thumbs.

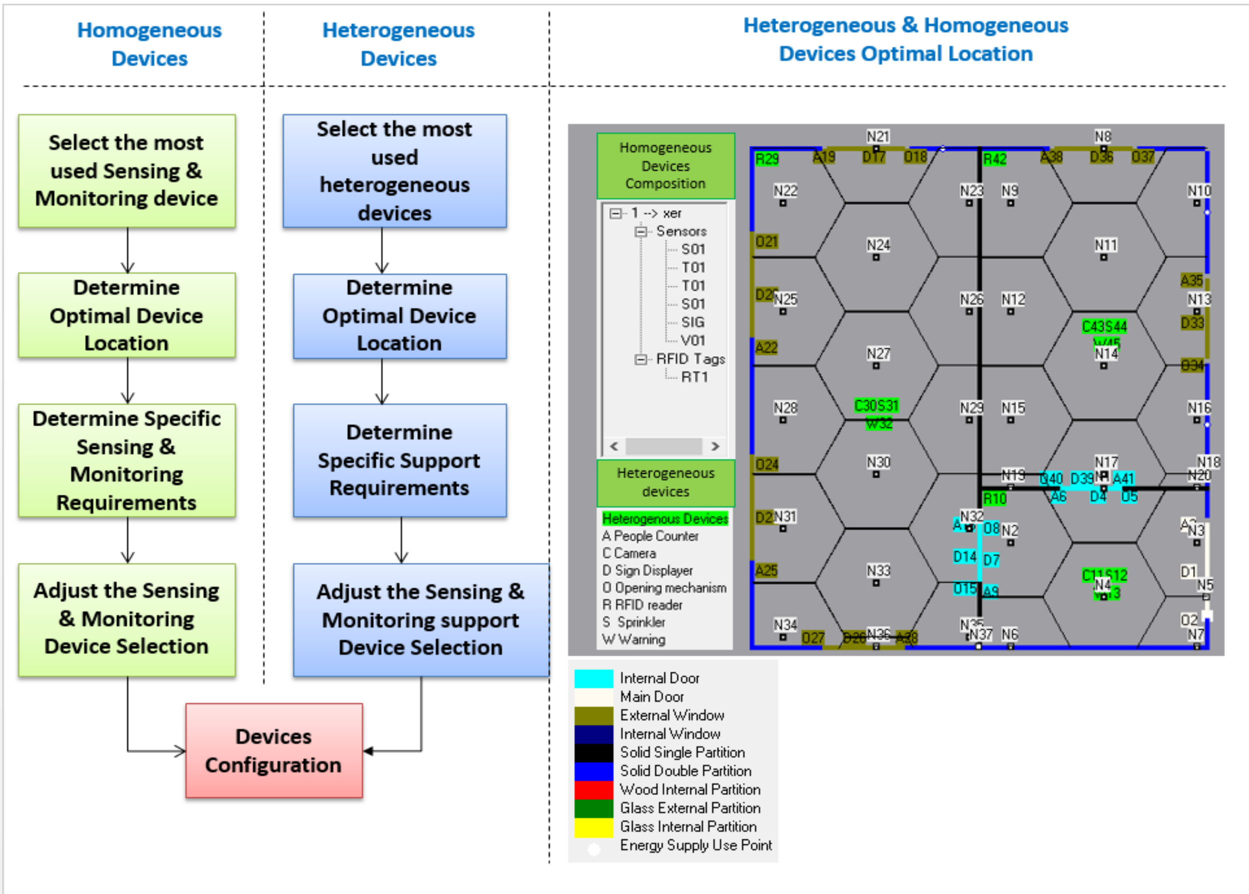
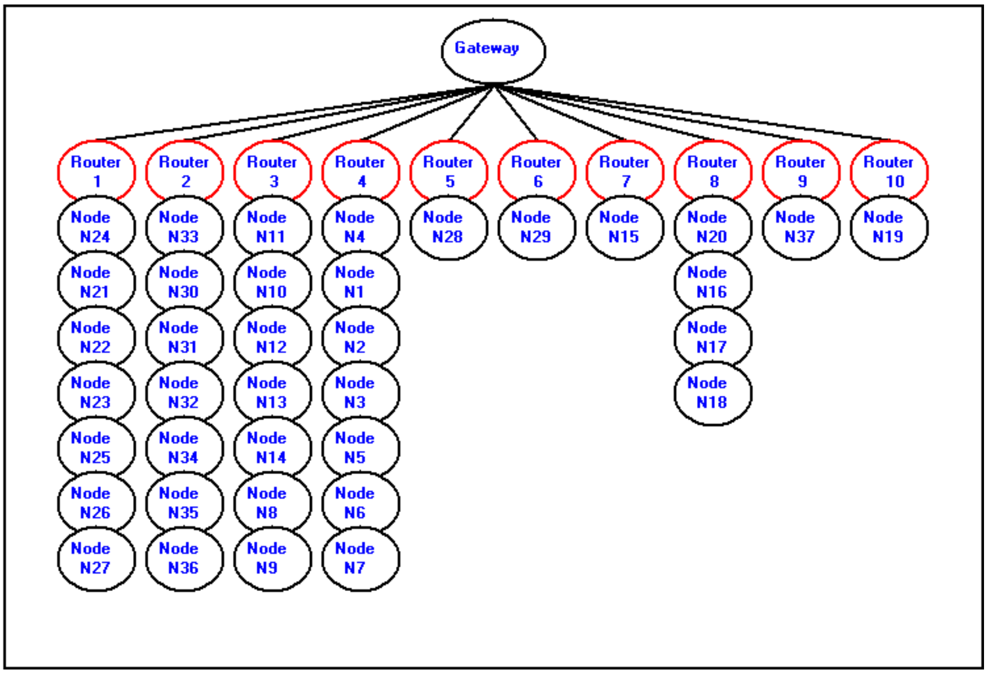


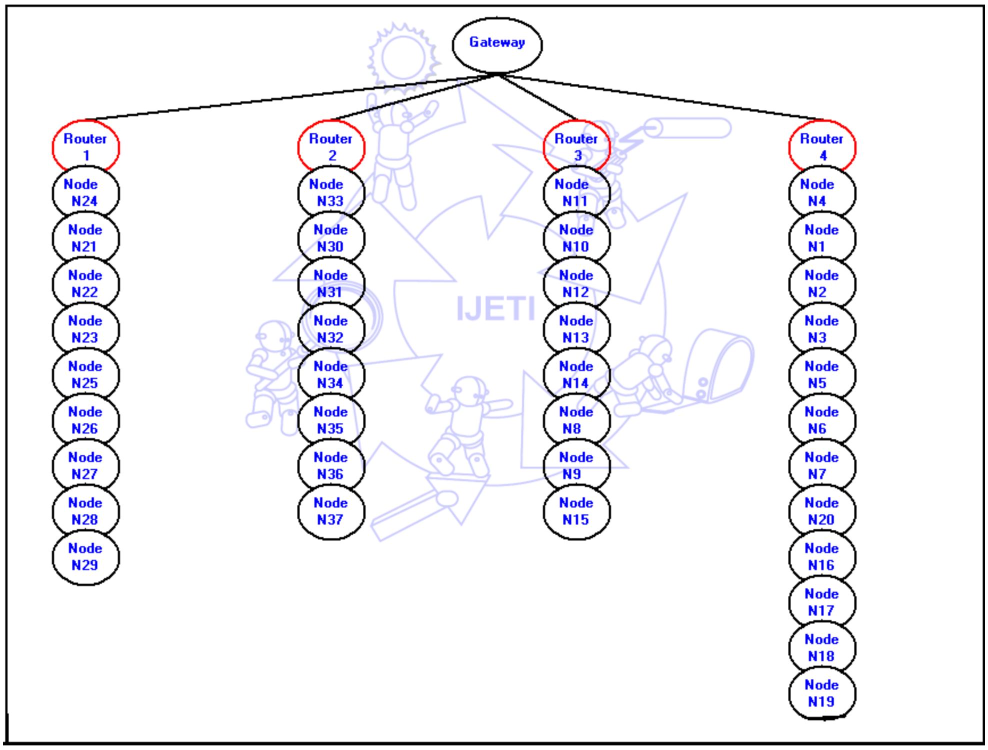
Fig. 5 Location model for homogeneous and heterogeneous devices allocation

4.5 Ad-hoc WSN configuration and clusterisation

Distributed topology control is a mechanism that enables adjustment at the different level of the network in terms of prolonging network lifetime via power conservation and increasing network capacity using spatial bandwidth monitoring and reuse, ensuring reach ability between two sensor nodes within the same or different sensor node clusters. The incorporation of different sensors in sensor nodes forms a network of heterogeneous wireless devices and generates a data heterogenisation with different maximum transmission ranges subject to asymmetric wireless links. A distributed topology control algorithm is required to calculate the nodal reading and transmission power required for the data reading, its in-networking and transmission to the front-end server, in the context of creating sensor nodes clusters. Virtual clusters defined as a logical level of grouping sensor nodes into clusters shown in Fig. 6, enable flexibility in the WSN deployment to support more effectively a variety of context applications.



(a) Before Clusters Aggregation



(b) After Clusters Aggregation

Fig. 6 Sensor nodes configuration and clustering

Examples of cluster logical levels above illustrated are:

– All the sensor nodes involved in detection event response are configured as one sensor node cluster with more reliable connection specifications (active connection, high performance routers, high band signal), whereas the other sensor nodes of the WSN will be configured separately in other clusters.

– All the non-switchable sensor nodes are configured as one sensor node cluster, whereas the other sensor nodes of the WSN will be configured separately in other clusters.

Sensor nodes clustering policies are an essential tool for the WSN configuration, elaborated from process model supporting the information fusion resulting from the context knowledge domain configuration.

4.6 *Predictive data for building navigation*

The predictive data for building navigation shown in Fig. 7 is based on calculating the rooms capacity and using average evacuation times to determine the minimal evacuation time required.

Rooms nodes

Node	From	To	Type	Dir.
1	1	0	3	1
2	1	2	2	7
3	1	3	2	5
4	2	0	1	1
5	2	0	1	7
6	3	0	1	7

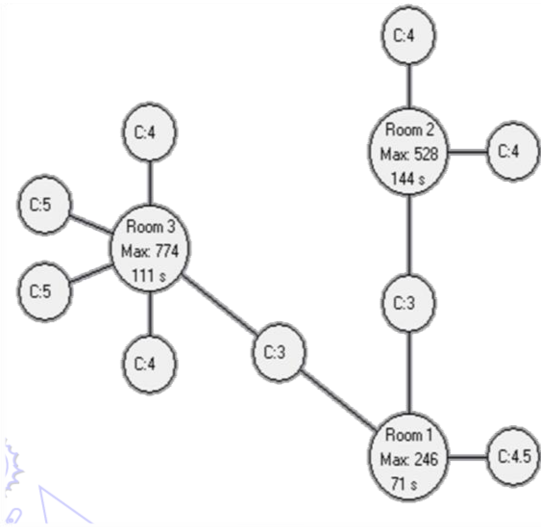
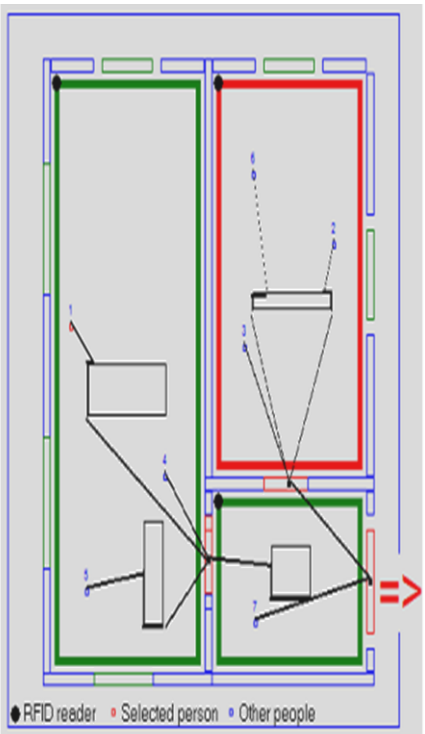


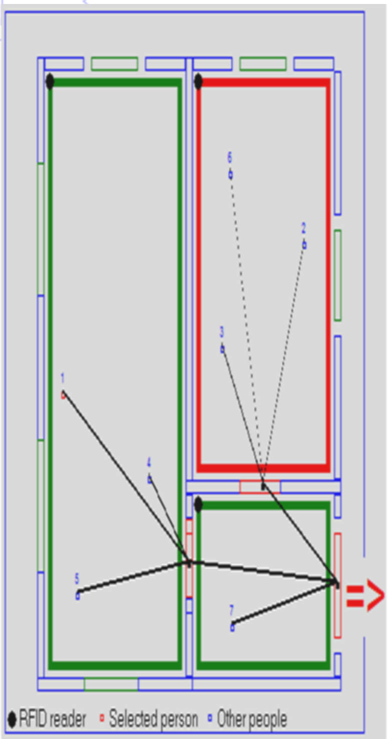
Fig. 7 Predictive data for building navigation

4.7 *Predictive data for building evacuation*

The predictive data for building evacuation shown in Fig. 8 is based on identifying the evacuation path set to be the direct line between the initial localisation of every person attending the building and the middle of the room door, and using average an algorithm for collision detection.



N.	Time	x	y
1	20:42:04	632	150
2	20:42:04	633	152
3	20:42:05	634	153
4	20:42:05	635	155
5	20:42:05	636	156
6	20:42:05	637	157
7	20:42:05	638	159
8	20:42:05	639	160
9	20:42:05	640	162
10	20:42:05	641	163
11	20:42:05	642	164
12	20:42:05	643	166
13	20:42:05	644	167
14	20:42:05	645	169
15	20:42:06	646	170
16	20:42:06	647	171



ID	Start	End	Duration	Room	N.
41767	20:42:04	20:42:22	00:00:18	3	1
41784	20:42:04	20:42:15	00:00:11	2	2
141767	20:42:04	20:42:13	00:00:09	2	3
241767	20:42:04	20:42:17	00:00:13	3	4
341767	20:42:04	20:42:20	00:00:16	3	5
341784	20:42:04	20:42:16	00:00:12	2	6
741767	20:42:04	20:42:12	00:00:08	1	7

Fig. 8 Predictive data for building evacuation

4.8 Predictive data for fire alarm simulation

The predictive data for fire alarm simulation model shown in Fig. 9 is based on calculating the fire proximity to the detector location which sets its alarm on when this distance becomes equal or smaller than the sensing distance.

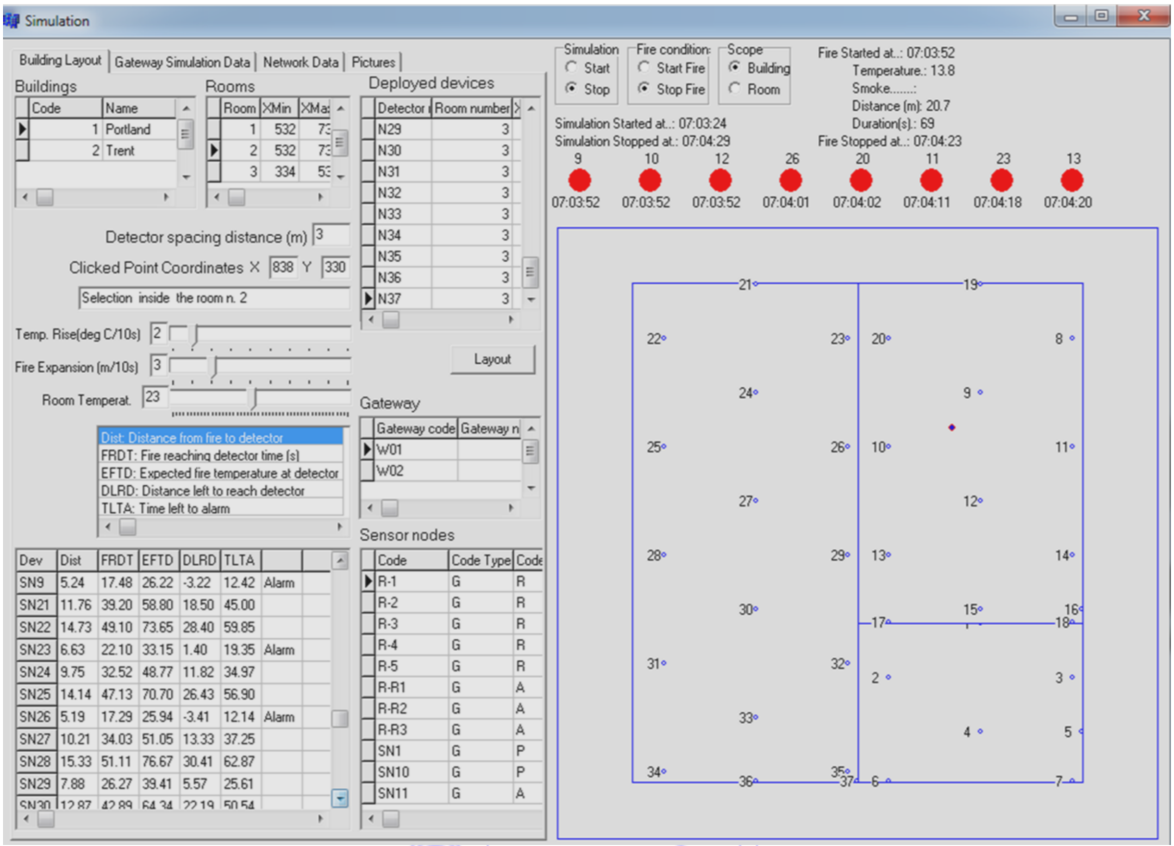


Fig. 9 Predictive data for fire alarm simulation

As the fire propagates, the individual distance to each detector in the cluster and to other surrounding detectors increases by integrating the effect of the two following parameters: Fire expansion or propagation, and Temperature rise. The data resulting from this alarm simulation is twofold: the predictive data given by the fore simulation model, and the data captured by the WSN from the different detectors. The predictive data shown in Fig. 9 is composed of: Distance from fire to detector (DIST), Fire reaching detector time (FRDT), Expected fire temperature at detector (EFTD), Distance left to reach detector (DLRD), and Time left to alarm (TLTA).

5. Functional framework for data fusion solutions

The generic fusion system design framework developed in this work is a methodological support that enables the mapping of a knowledge domain model and architecture to a generic design for distributed detection applications, covering several domain contexts that include sensing and tracking. These applications are designed in the form of web-based services for fast and flexible access to real-time data captured by sensor networks and processed to support real-time inferring decision making in the context of data, information and knowledge fusion. They integrate concurrently third party fusion data, models and knowledge, and share fusion tools and processes jointly supported by web-based GDSS linking end-users. They support generic data fusion solutions based on data and information fusion management to procure fusion capabilities structured in fusion process models services decoupled from other services representing knowledge processes, models, data & support system capabilities as illustrated in Fig. 10. The proposed generic design conceptual fusion framework focuses on service interoperability which is based on generic and recursive fusion in inferring decision making using the different management components (Knowledge, model, data and support capabilities) to deploy configure, and operate the network homogeneous and heterogeneous devices, capture, integrate, process, and aggregate their data.

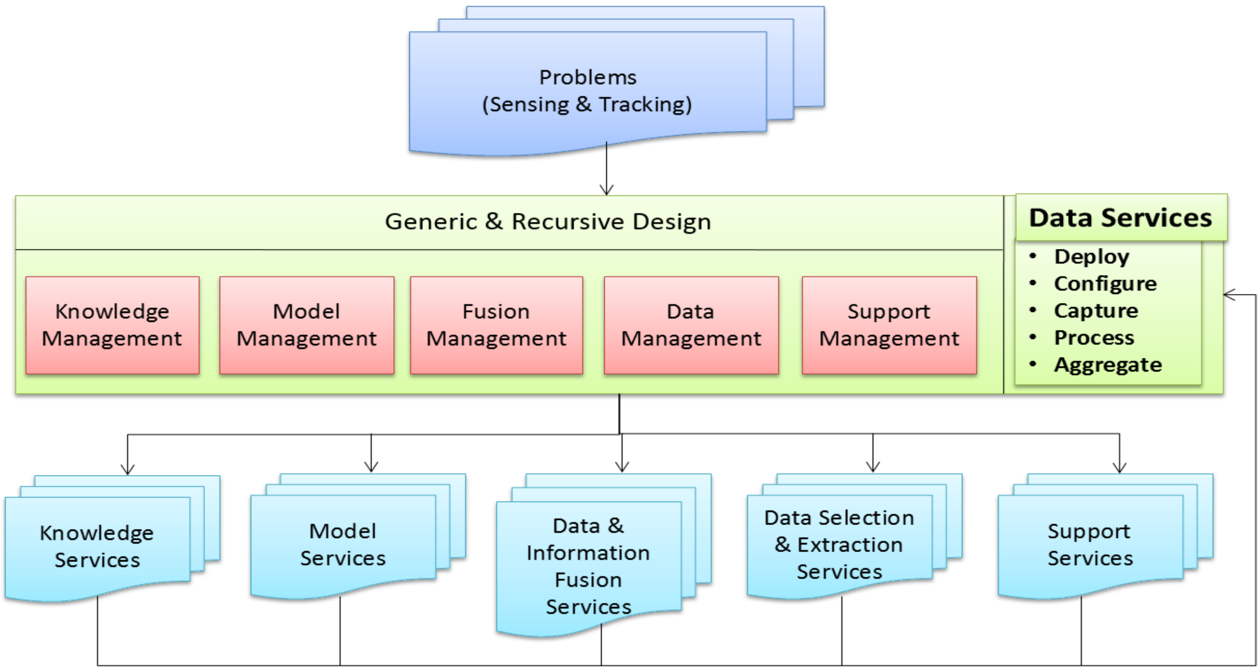


Fig. 10 Distributed Web-based service support for networks data fusion

6. Conclusions

A novel data fusion approach for big data has been proposed in this paper to solve problems inherent to networks multi-source data fusion for distributed detection based on the use of ad-hoc integrated WSNs-RFID combining generic homogeneity and heterogeneity. Although this approach suggests the decoupling of data fusion from information fusion, the case study used in this work has shown the inevitable interaction between these two hierarchical processes that enable the synergy of knowledge discovery and extraction. This interaction has been shown in the generic sensor node data fusion design implementation through the different techniques and models used in the proposed fusion framework to generate predictive data for fire alarm simulation, and also in the knowledge-based sensor node design and its integration of sensors and RFID tags to procure adaptive configurable smart devices for distributed detection.

7. Future work

The multi-level interaction between data, information and knowledge is under explored due to the importance of data uncertainty and complexity associated with both the knowledge domain context and the use of non-interoperable heterogeneous devices. The improvement of the elicitation methods, tools and models of fusion requirements in the context of requirement engineering to define elaborate fusion functions, and the development of more interoperable generic smart devices are new directions for future research and applications.

References

[1] F. Alshahrany, H. Zedan, and I. Moualek, "A conceptual framework for small WSN configuration using intelligent decision support system," The third International Conference On Innovative Computing Technology (INTECH '13), IEEE press, 2013, pp. 349-355.

[2] H. F. Durrant-Whyte, "Sensor models and multisensor integration," Autonomous Robot VehiclesAnonymous Springer, 1990, pp. 73-89.

[3] B. V. Dasarathy, "Sensor fusion potential exploitation-innovative architectures and illustrative applications," Proc IEEE, vol. 85, pp. 24-38, 1997.

[4] J. N. Tsitsiklis, "Decentralized detection by a large number of sensors," Mathematics of Control, Signals and Systems, vol. 1, pp. 167-182, 1988.

[5] F. Zhao, J. Shin, and J. Reich, "Information-driven dynamic sensor collaboration," Signal Processing Magazine, vol. 19, pp. 61-72, 2002.
[6] M. A. Osborne, S. J. Roberts, A. Rogers, S. D. Ramchurn, and N. R. Jennings, "Towards real-time information processing of sensor network data using computationally efficient multi-output gaussian processes," The 7th International Conference on Information Processing in Sensor Networks, 2008, pp. 109-120.
[7] F. Alshahrany, H. Zedanb, and I. Moualekc, "WSN configuration using agent modeling and hybrid intelligent decision support system," International Journal of Sciences: Basic and Applied Research, vol. 14, pp. 67-90, 2014.
[8] F. Alshahrany, M. Abbod, and I. Moualek, "WSN and RFID integration to support intelligent monitoring in smart buildings using hybrid intelligent decision support systems," International Conference on Computational and Experimental Science and Engineering, Acta Physica Polonica, 2015, pp. 152 - 159.
[9] J. Lu, F. Valois, D. Barthel, and M. Dohler, "Fisco: A fully integrated scheme of self-configuration and self-organization for WSN," Wireless Communications and Networking Conference (WCNC ’07), IEEE press, 2007, pp. 3370-3375.
[10] D. L. Hall and J. Llinas, "An introduction to multisensor data fusion," IEEE Proc., vol. 85, pp. 6-23, 1997.
[11] R. Stranders, F. M. Delle Fave, A. Rogers, and N. Jennings, "A decentralised coordination algorithm for mobile sensors," Twenty-Fourth AAAI Conference on Artificial Intelligence, 2010, pp. 874-880.
[12] S. M. Smith and J. M. Brady, "SUSAN—A new approach to low level image processing," International Journal of Computer Vision, vol. 23, pp. 45-78, 1997.
[13] M. Yarvis, N. Kushalnagar, H. Singh, A. Rangarajan, Y. Liu, and S. Singh, "Exploiting heterogeneity in sensor networks," The 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM ‘05), IEEE press, 2005, pp. 878-890.
[14] L. Zhang and Z. Wang, "Integration of RFID into wireless sensor networks: Architectures, opportunities and challenging problems," Fifth International Conference On Grid and Cooperative Computing Workshops (GCCW '06), 2006, pp. 463-469.
[15] J. Mitsugi, T. Inaba, B. Pátkai, L. Theodorou, J. Sung, T. S. López, D. Kim, D. McFarlane, H. Hada, and Y. Kawakita, "Architecture development for sensor integration in the EPCglobal network," Auto-ID Labs White Paper Series, pp. 8-13, 2007.
[16] S. Cheekiralla and D. W. Engels, "A functional taxonomy of wireless sensor network devices," The 2nd International Conference On Broadband Networks (BroadNets ’05), 2005, pp. 949-956.
[17] H. Liu, M. Bolic, A. Nayak, and I. Stojmenovi, "Integration of RFID and wireless sensor networks," Proc. of Sense ID Worksop at ACN SenSys, 2007, pp. 6-9.
[18] T. Hegazy and G. Vachtsevanos, "Sensor placement for isotropic source localization," Information Processing in Sensor Networks, 2003, pp. 432-441.
[19] A. Sheth, C. Henson, and S. S. Sahoo, "Semantic sensor web," IEEE Internet Computing, vol. 12, pp. 78-83, 2008.
[20] M. Compton, P. Barnaghi, L. Bermudez, R. GarcíA-Castro, O. Corcho, S. Cox, J. Graybeal, M. Hauswirth, C. Henson, and A. Herzog, "The SSN ontology of the W3C semantic sensor network incubator group," Web Semantics: Science, Services and Agents on the World Wide Web, vol. 17, pp. 25-32, 2012.

International Journal of Engineering and Technology Innovation, vol. 6, no. 1, 2016, pp. 16 - 29

Improved-Coverage Preserving Clustering Protocol in Wireless Sensor Networks

Manju*, Satish Chand, Bijender Kumar

Netaji Subhash Institute of Technology, Sector-3, Dwarka, New Delhi, 110078, India

Received 13 November 2015; received in revised form 15 December 2015; accepted 19 January 2016

Abstract

Coverage maintenance for longer period is crucial problem in wireless sensor network (WSNs) due to limited inbuilt battery in sensors. Coverage maintenance can be prolonged by using the network energy efficiently, which can be done by keeping sufficient number of sensors in sensor covers. There has been discussed a Coverage-Preserving Clustering Protocol (CPCP) to increase the network lifetime in clustered WSNs. It selects sensors for various roles such as cluster heads and sensor cover members by considering various coverage aware cost metrics. In this paper, we propose a new heuristic called Improved-Coverage-Preserving Clustering Protocol (I-CPCP) to maximize the total network lifetime. In our proposed method, minimal numbers of sensor are selected to construct a sensor covers based on various coverage aware cost metrics. These cost metrics are evaluated by using residual energy of a sensor and their coverage. The simulation results show that our method has longer network lifetime as compared to generic CPCP.

Keywords: sensor networks, energy-efficiency, clustering, network lifetime, coverage

1. Introduction

Wireless sensor network (WSNs) aims to provide monitoring (called coverage) services in many areas [1]. In coverage problems, the objective is to keep all the points of interest within the sensing range of at least one sensor node for maximum possible time. There are many works [2-5] in this area which aims to achieve full coverage through various energy based heuristic approaches. In all above works, it has been shown that instead of activating all the sensors at a time, it is better to activate a subset of sensors (called sensor cover) which covers all the points of interest. After generating maximum possible cover sets, the next phase is to communicate the data collected by these sensor covers to the base station (BS) where further processing is to be done. In large sensor network, the maintenance of centralize processing is not performing well specially, when it comes to the quality and throughput requirement. Therefore, it is better to design distributed network where each node is part of processing (instead of central processor), thus, they can spread the workload out more evenly than the centralized algorithm [6]. To apply distributed approach, the whole networks to be divided into small groups based on density, neighbourhood, residual energy or many other parameters of interest. Recently, clustering is one of the best ways so far for designing scalable and energy-efficient distributed sensor networks [7]. Efficient utilization of clusters reduces the communication overhead and interference among the sensor nodes, thereby decreasing the energy consumption for overall network architecture. Research community has widely pursue clustering over the last few years which leads to the appearance of a great number of application-oriented customized clustering protocols [8–9]. When applying clustering, all the nodes deployed

* Corresponding author. E-mail address: manju.nunia@gmail.com

Tel.: +919560686262

in the network organize themselves into local groups called clusters, where only one node among each group is acting as the cluster head. All non-cluster-head nodes used to transmit their information to their cluster heads. Due to this, some neighbouring sensor nodes may send redundant data at the cluster head node which lead to unavoidable energy consumption in energy scared sensor networks. To avoid this problem, the cluster head nodes perform local aggregation on the received information and forward it to the remote base station where final processing is to be performed. Local data aggregation at the cluster heads can significantly reduce the total amount of data to be sent to the base station thus reducing energy consumption in transmitting data. This in turn results in longer network lifetime.

Lots of work has been done in the area of coverage problems as well as on clustering in WSNs. But, there are very few works [10-11, 23] that dealt with both the problems (coverage problems and clustering techniques) in the wireless sensor networks at the same time. Therefore, in this paper, our main focus is to provide coverage for longer period in clustered WSNs. The rest of the paper is organized as follows: related work on the coverage and clustered network is given in section 2. Section 3 discusses the various coverage aware cost metrics which are further used to decide various roles of sensors. Section 4 describes the proposed heuristic called Improved-Coverage-Preserving Clustering Protocol (I-CPCP) to maximize the total network lifetime. Section 5 gives a series of simulation results to claim the quality of proposed method. Conclusion is given in section 6 of the paper.

2. Related Work

Several studies on energy-efficient coverage have been conducted in the field of wireless sensor networks. Here our main discussion is on two basic problems: coverage problems and efficient use of total network energy. Coverage problems deal with prolonging the total coverage for maximum possible time. Since the sensors in WSNs have limited available batteries which is not rechargeable or cannot be replaced [1], it is very essential to utilize the network energy very efficiently. The efficient utilization of the network energy is generally done through clustering. Following sections discuss related work of the coverage and clustering in wireless sensor network.

2.1 Coverage

The coverage problem in wireless sensor networks is an important area due to several applications including health care, national security, environmental monitoring and surveillances [1]. There has been some works [2-5] that address the coverage problem in WSNs. In these works, the basic objective is to activate a subset of sensors (called sensor cover) based on various criteria like, residual energy of nodes, number of 1-hop neighbour nodes etc. One of the known variant of coverage problem i.e. target coverage problem which has been solved by Cardei and Du [2] where disjoint sensor covers are generated such that every target point is completely covered by each cover. They proved that the disjoint sensor set problem is NP-Complete problem. This problem is further addressed by Cardei,Thai, Li, and Wu [3] in which the sensor covers need not to be disjoint, that is, a sensor can be a member of more than one sensor cover (non-disjoint set). They used greedy heuristic to generate sensor cover by giving priority to those sensors covering the critical target (covered by least number of sensors) and also those covering maximum number of uncovered targets. Mini, Siba and Samrat [4] propose another algorithm for target coverage problem where a weighted coverage is calculated for each sensor. To select sensor in a cover, sensor with more weights are given priority. In many *QoS* based surveillance applications, it is all time necessary that particular area should be covered by at least *Q*-number of sensors. It is generally used when there is a very high probability of a node failure or the data collected by the sensors with large amount of noise. Manju and Arun [5] proposed an energy based heuristic to solve target Coverage problem where sensor covers are generated by choosing high energy sensors first. Being a special case of coverage problems, *Q*-Coverage problem [21, 22] is also reported as NP-Complete problem. To address the impact of standby mode and energy consumption in peak hours, Collotta and Giovanni [26] address a completely new approach called information and communication technologies (ICT). The ICT ensures energy management in smart homes which combines a wireless network based on Bluetooth Low Energy (BLE) for communication among home appliances, with the help of home energy management (HEM) scheme.

2.2 Clustering

As discussed in section 1, centralized communication does not perform well when deployed networks size grows exponentially. Distributed network design can resolve this problem in energy-efficient way. As far as distributed network design is considered, clustering is the best way to design distributed network architecture.

LEACH [12] is one of the first classical clustering protocols in WSNs which selects cluster heads dynamically and frequently by a round mechanism and it allows different nodes to become cluster heads at each round. However, this protocol cannot guarantee good cluster distribution and require strict time synchronization. PEGASIS [13] is an extension of LEACH, in which all nodes are organized into a chain and each node only transmits data to its nearest neighbour. Compared to LEACH, PEGASIS reduces the energy consumption of the nodes, but the data delay is significantly high and is proved to be unsuitable for large-sized networks. The HEED clustering protocol [14] uses a hybrid criterion for cluster head selection, which considers the residual energy of each node and a secondary parameter, such as the node's proximity to its neighbours or the number of its neighbours. HEED plays a great role in reducing the energy consumption of the nodes and enhancing the network lifetime. HEED and LEACH require re-clustering after a period of time (called round), which causes extra energy consumption. Therefore, to resolve this problem, the generic weight-based clustering algorithm (called Weighted Clustering Algorithm (WCA)) [8] is proposed where each sensor associated with some weight is good examples of genetic clustering. In WCA approach, weight is calculated from a sensor's local properties, such as speed, node degree, coverage and residual energy. Cluster heads are selected from those nodes that have the highest weights among their neighbours. Quin and Roger [16] discuss a Voting based Clustering Algorithm (VCA) which performs some voting criteria to select cluster head. VCA can generate fewer clusters and a longer lifetime than clustering algorithms discussed in [12-14]. To elect a cluster head, VCA basically perform voting scheme where each sensor can get vote only from its neighbour sensors. So, nodes with more neighbours tend to receive more votes. Thus, cluster heads are likely to be those high-degree nodes. Most of the clustering algorithms like LEACH, PEGASIS and HEED [12-14] have been designed for homogenous network and they perform poorly in heterogeneous environment. There are few works which discuss heterogeneous network. In [17], Stable Election Protocol (SEP) is discussed for a two level heterogeneous network, which includes two types of nodes according to the initial energy, i.e. the advanced nodes and normal nodes. It selects cluster head based on the residual energy of node and weighted probability. Kumar, Aseri and Patel [18] discuss an energy-efficient heterogeneous cluster (EEHC) protocol for three-level heterogeneous WSNs whose nodes are divided into three types, i.e., super, advanced and normal nodes. It selects cluster heads using the node's energy after dividing by the average networks residual energy and weighted probability. In [19] Hong, Li and Zhang discuss a heuristic for multilevel heterogeneous wireless sensor networks which is known as EDCS (Efficient and Dynamic Clustering Scheme). It calculates some weighted probability (based on remaining energy) of node to decide cluster heads.

In order to save energy consumption, Jihwan Hyang-Won and Song [24] proposed a Traffic-aware energy-saving (TAES) base station sleeping and clustering approach where network energy is saved by putting BSs off when network capacity is excessive compared to traffic demand. Apart from that, they also addressed a new optimal clustering method that has polynomial complexity. To provide secure data and communication in clustered wireless sensor networks, Seung-Hyun, Jongho, Salmin and Elisa [25] provide a secure encryption key protocol named certificate less-effective key management

(CL-EKM) for dynamic WSN. It ensures forward and backward key secrecy at the moment a node leave or joins a cluster. Due to the severe energy constraints, hardware crash, software bugs and environmental risks, the network reliability should be on priority as nodes. Therefore, in order to provide fault tolerant network, Venkatesh and Mehata [23] discussed a different approach for clustering and optimal coverage of tracking multiple objects in WSNs. They proposed an improved K-means clustering algorithm with simulated annealing technique. The discussed method finds optimal set of sensor nodes for covering moving targets and recover from fault as soon as possible so that target detection won't be disrupted. Another security based method is discussed in [27] where a highly secure authentication mechanism is proposed in order to provide secure communication between deployed nodes with gateway nodes.

As discussed in this section, there is lots of work done on coverage [4, 22, 28] and clustering [19, 29] separately, but there are very few works which deals with both the issues collectively [10, 11]. Authors in [4] consider energy as the prime criteria to find cover set while in [22], both coverage and energy are considered to generate sensor covers. In [28], a hybrid approach based on genetic algorithm is used to find cover set for partial coverage. In order to provide clustering in WSN, Authors in [19] discussed a new approach which also considers heterogeneity in terms of energy in the sensor networks. Similarly, in [29] a new self-configurable clustering mechanism is presented which periodically check for cluster heads failure and replace these failed CHs with newly elected nodes so that the network activity won't interrupt. As we said earlier too, these methods do not take coverage and clustering into account collectively. Therefore, in this paper, we consider only those works which provide coverage and clustering both in the sensor network [10, 11]. Liu, Zheng, Xue and Guan [10] proposed Distributed Energy-Efficient Clustering with Improved Coverage (DEECIC) techniques while maximizing coverage lifetime. The main objective of DEECIC is to generate least number of cluster heads by assigning a unique ID to each node based on local information. Due to least number of CHs, the communication overhead is bit high in this approach. The work proposed in [11] considers four weighted cost metrics to elect cluster heads as well as sensor covers. The proposed scheme out performs the well-known HEED method of clustering [14], but none of the cost applied for coverage followed by minimalization of generated sensor cover. The generated cover sets are not minimal in [11] which results in less total network lifetime. Therefore, in this paper, our proposed method minimizes the generated sensor cover also which results in extended network lifetime as minimal sensor cover are having least required sensors for full coverage. Our proposed approach also generates sufficient number of CHs in order to reduce communication overhead unlike [10]

3. Coverage Aware Cost Metrics

Every node has its own importance in the deployed wireless sensor network. Therefore, based on that importance, sensors are selected for various roles (like cluster head, cluster member, routers, or sensor covers for coverage). To decide the roles of a sensor, there are various parameters like its residual energy, number of 1-hop neighbouring sensor nodes, number of n-hop neighbouring sensor nodes and its distance from the base station. By keeping these criteria in view, Soro and Heinzelman [11] discuss various coverage aware cost metrics to decide their roles for various activities and propose a Coverage Preserving Clustering Protocol (CPCP).

Before discussing the coverage aware cost metrics, we first estimate the total energy available to cover each point in the network area. We consider a set S consisting of N sensors, $s_i \in S$, i=1,…, N, randomly deployed in the square area A. Each sensor node has its sensing area $C(s_i)$ (circular area around the node) with sensing range R_{sense}. For each sensor node s_c, calculate, its 1-hop neighbouring sensors node denoted by N_c as follows:

$$N(c)=\{s_k \mid d(s_c, s_k) <= R_{sense}\} \tag{1}$$

Where d (sc, s_k) is the Euclidean distance between nodes sc and sk. The whole area is covered by the sensors of S. Some area might be covered by multiple sensors. Let A_i denote the area covered by the sensor s_i, A_{ij} denote the area covered by the sensors s_i and s_j both and so on (Fig.1). The area A may be considered as consisting of the points (x, y), which would lie in one of the A_i or A_{ij} and so on. We calculate the total amount of energy denoted by *Etotal(x, y)* which is available to cover the point (x, y) that is given as below.

$$Etotal(x,y) = \sum_{S_j:(x,y)\in C(S_j)} E(s_j) \tag{2}$$

where $E(s_j)$ is the remaining energy of sensor s_j. The total coverage available for a point (x, y) is defined as the number of sensors covering it, and it is given as below.

$$Ototal(x,y) = \sum_{S_j:(x,y)\in C(S_j)} 1 \tag{3}$$

The followings are the coverage aware cost matrices discussed in [11] based on the total remaining energy or coverage of a sensor node.

3.1 *Minimum-Weight Coverage Cost [11]*

The minimum-weight coverage cost is defined as below:

$$Cmw(s_i) = \max \frac{1}{Etotal(x,y)} \quad \text{where } (x, y) \in C(s_i) \tag{4}$$

3.2 *Weighted Sum Coverage Cost [11]*

The weighted-sum coverage cost is defined as below:

$$Cws(s_i) = \int_{C(S_i)} \frac{dxdy}{Etotal(x,y)} \tag{5}$$

This weighted sum coverage cost metric measures the weighted average of the total energies of all points that are covered by the sensing area of node s_i.

3.3 *Coverage Redundancy Cost [11]*

This is the cost which does not depend on the nodes remaining energy as well as its neighbouring nodes remaining energy. Apart from the energy view, this cost only looks at the coverage (*Ototal*) of a particular sensor and its coverage redundancy with its neighbouring sensors. The coverage redundancy cost of sensor s_i is

$$Ccc(s_i) = \int_{C(S_i)} \frac{dxdy}{Ototal(x,y)} \tag{6}$$

3.4 *Energy-Aware Cost [11]*

The energy-aware cost function simply evaluates the sensor's priority to take part in the sensing task based solely on its remaining energy $E(s_i)$. Therefore, this cost can be expressed as:

$$Cea(s_i) = \frac{1}{E(s_i)} \tag{7}$$

To understand all the discussed coverage aware cost matrices in detail, we illustrate all the costs defined above using the following example (Fig.1) where four sensors (S_1, S_2, S_3, S_4) with radius R_{sence} are part of the network with initial residual energy given as below.

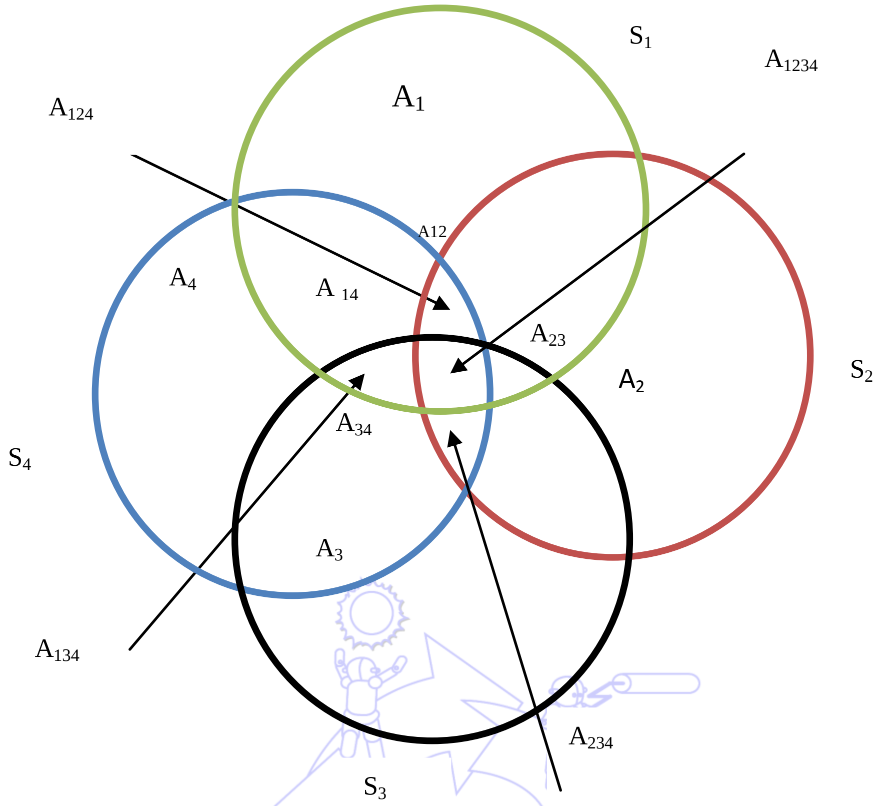


Fig.1 Four circles denoting sensors S_1, S_2, S_3, S_4 and their sensing range with their intersection areas.

The area covered by *k* sensors (among these four) are denoted by $A_{i...k}$. Fig.1 take residual energy of all sensors as: $E(S_1) = 3$, $E(S_2) = 2$, $E(S_3) = 4$, $E(S_4) = 1$ and then calculate various coverage aware cost matrices in detail.

$$Etotal\ (x, y) = \left\{ \begin{array}{c} 3, (x,y) \in A1 \\ 2, (x,y) \in A2 \\ 4, (x,y) \in A3 \\ 1, (x,y) \in A4 \\ 5, (x,y) \in A12 \\ 6, (x,y) \in A23 \\ 5, (x,y) \in A34 \\ 4, (x,y) \in A14 \\ 9, (x,y) \in A123 \\ 6, (x,y) \in A124 \\ 8, (x,y) \in A134 \\ 7, (x,y) \in A234 \\ 10, (x,y) \in A1234 \end{array} \right\} \quad (8)$$

and

$$Ototal\ (x, y) = \left\{ \begin{array}{c} 1, (x,y) \in A1 \\ 1, (x,y) \in A2 \\ 1, (x,y) \in A3 \\ 1, (x,y) \in A4 \\ 2, (x,y) \in A12 \\ 2, (x,y) \in A23 \\ 2, (x,y) \in A34 \\ 2, (x,y) \in A14 \\ 3, (x,y) \in A123 \\ 3, (x,y) \in A124 \\ 3, (x,y) \in A134 \\ 3, (x,y) \in A234 \\ 4, (x,y) \in A1234 \end{array} \right\} \quad (9)$$

3.5 Minimum Weighted Cost

The minimum weighted coverage cost defined in (4) only considers the energy of the most critically covered target. Thus, for the above example, it can be calculated as follows:

$Cmw\ (S_1)$ =1/*min* (3, 4, 5, 6, 8, 9, 10) =1/3;

$Cmw\ (S_2)$ =1/*min* (2, 5, 6, 6, 7, 9, 10) =1/2;

$Cmw\ (S_3)$ =1/*min* (4, 5, 6, 7, 8, 9, 10) =1/4;

$Cmw\ (S_4)$ =1/*min* (1, 4, 5, 7, 8, 10) =1/1;

3.6 Weighted Sum Cost

As defined in (5), weighted sum coverage cost metric measures the weighted average of the total energies of all points that are covered by the sensing area of a sensor. Here, for the given example, it can be calculated as follows:

$Cws\ (S_1) = A_1/3+A_{12}/5+A_{14}/4+A_{123}/9+A_{124}/6+A_{134}/8+A_{1234}/10;$

$Cws\ (S_2) = A_2/2+A_{12}/5+A_{23}/6+A_{123}/9+A_{124}/6+A_{234}/7+A_{1234}/10;$

$Cws\ (S_3) = A_3/4+A_{23}/6+A_{34}/5+A_{123}/9+A_{134}/8+A_{234}/7+A_{1234}/10;$

$Cws\ (S_4) = A_4/1+A_{34}/5+A_{14}/4+A_{124}/6+A_{134}/8+A_{234}/7+A_{1234}/10;$

3.7 Minimum Weighted Cost

As per the definition (6), this cost only looks at the coverage of a particular sensor and its coverage redundancy with its neighbouring sensors. Thus, for the above example, it can be calculated as follows:

$Ccc\ (S_1) = A_1/1+ (A_{12}+A_{14})/2+ (A_{123}+A_{124}+A_{134})/3+A_{1234}/4;$

$Ccc\ (S_2) = A_2/1+ (A_{12}+A_{23})/2+ (A_{123}+A_{124}+A_{234})/3+A_{1234}/4;$

$Ccc\ (S_3) = A_3/1+ (A_{23}+A_{34})/2+ (A_{123}+A_{134}+A_{234})/3+A_{1234}/4;$

$Ccc\ (S_4) = A_4/1+ (A_{34}+A_{14})/2+ (A_{124}+A_{134}+A_{234})/3+A_{1234}/4;$

3.8 Energy Aware Cost

Energy aware cost is the inverse function of sensors remaining energy as defined in (7). Thus, for the given example, it can be calculated for each sensor as given below:

$Cea\ (S_1)$ =1/3; $Cea\ (S_2)$ =1/2; $Cea\ (S_3)$ =1/4; $Cea\ (S_4)$ =1/1;

4. Improved-Coverage Preserving Clustering Protocol (I-CPCP)

This section describes our proposed heuristic to maximize the coverage lifetime for monitoring the objects in a given environment in WSNs. There are three main phases in proposed heuristic (I-CPCP): a) Cluster head selection and cluster formation, b) Sensor covers selection and c) Routing phase. Our proposed work initially consider the various coverage aware costs matrices, i.e., *Cmw*, *Cws*, *Ccc* or *Cea* discussed in section 3 for cluster heads selection, sensor cover generation, and for routing the data from cluster heads to the sink node.

4.1 Phase 1: Cluster Head Selection and Cluster Formation

This phase decides the cluster head nodes in the deployed network followed by their clusters. First, we fix one coverage aware costs among *Cmw*, *Cws*, *Ccc* or *Cea*. For deciding the cluster heads, our methods choose that sensor having smallest coverage aware cost. All the sensors lying in its sensing range constitute its cluster members. So far, we have decided a cluster head and its cluster members. We again consider a sensor node having minimum coverage aware cost among the sensor nodes after excluding the first cluster head and its cluster members. This will be the second cluster head and its members are decided by its sensing range as was done in case of first cluster. This process of cluster heads and their cluster formation continues till all the sensors have been clustered. Algorithm in Fig. 2 shows the complete process of cluster heads selection and cluster formation.

1: $S_{initial}$ = Set of all sensor nodes
2: *E(s)*= remaining energy of sensor *s*
3: S_{CH} = {} // it contains cluster heads, which is initially empty
4: *while* $S_{initial}$! = $\acute{\emptyset}$ *do*
5: $Nc = \acute{\emptyset}$ // contain all neighbouring node of a cluster head s_c
6: *Select* sensor $s_c \epsilon S_{initial}$ that has minimum coverage aware cost
7: $S_{CH} = S_{CH} \cup s_c$
8: *find* neighbouring sensors *s* of s_c such that dist(s,s_c)$\leq R_{cluster}$
9: $N_c = N_c \cup s$
10: $S_{initial} = S_{initial} - N_c$
11: ***s*** send cluster JOIN message to cluster head $\boldsymbol{s_c}$
12: end while
13: *exit*

Fig. 2 Algorithm to find Cluster Head Selection and Cluster Formation

The detailed outline of Algorithm shows in Fig. 2 is as follows: Line 1 and 2 contain set of sensors initially deployed and their residual energy. In line 3, S_{CH} contains all cluster heads generated in the network which is initially empty. Cluster heads are formed and respective clusters are formed through line 4-12. Sensor with minimum cost is selected as cluster head (line 6) and included in $\mathbf{S_{CH}}$ (line 7). Once cluster head is selected, the respective cluster is formed which has all neighbouring sensors of selected cluster head. (line 8-11). Thus, at the end of Fig. 2, all cluster heads are generated and clusters are formed. In CPCP [11] some of the sensors left non-clustered at the end of this phase while forming the clusters. These sensor nodes which do not belong to any cluster head send their data to their nearest neighbour node. Due to this, the inter cluster communication over cluster heads increases.

4.2 Phase II: Sensor Cover Selection

Once cluster heads have been decided and their respective clusters have been formed. The next step is to find coverage of the objects in the given area by generating sensor covers. In this phase, only those sensors participate which have not been part of Phase 1. While choosing sensors for this phase, our method always gives preference to those sensors which carry smallest coverage aware cost among (*Cmw*, *Cws*, *Cea* and *Ccc*) in the network. For each sensor **s**, check whether it’s neighboring sensors with cost lesser than its cost are collectively covering all the objects which are covered by sensor **s**. If so, then the sensor **s** will not be part of current sensor cover and will go to off state. The neighboring sensors (with lower cost) will now take part in this process of deciding their states (to be on or off). To decide the state of neighbor sensors, our method keeps selecting lowest cost neighboring sensors till all the objects earlier covered by sensor **s** are not covered by these lowest cost neighbors. Once the coverage of sensor **s** is same as the set of such neighbor sensor nodes coverage (with lower cost) selected so far, then we stop selecting more neighbors even with cost lower than the cost of sensor **s (**where in CPCP [11], all the neighbors with cost lesser then cost of sensor **s** are selected as part of active sensor cover). The selected neighboring sensors will go into **on** state. This process will be carried out for all the remaining sensors to find out the complete sensor cover for the current round. We have completed the current round and started the next round in the same way as done in the current round. The process of rounds will be continued till no object is left uncovered. Our propped heuristic selects only minimum number of lower cost neighbor for each sensor **s.** Therefore, the number of sensors selected in sensor cover is much lesser than those selected by CPCP [11]. Due to this, less network energy is consumed for covering all the objects. The detailed process of coverage phase is given in Fig. 3 as follows:

1: $S_{initial}=\{s|E(s)>0, s\notin SCH\}$, $E(s)$=residual energy of sensor s
2:Assign Cost to each sensor s where Cost(s)∈{*Cmw*,*Cws*,*Cea*, *Ccc*}
3: $S_{cover}=\{\}$ // S_{cover} is set of active sensor
4: *for* each sensor *s* calculate
5: A_s=set of all the objects covered by sensor s
6: *for* each neighbour sensor s_k of sensor s
7: *if Cost* (s_k) <*Cost (s)*
8: $S_{nei}=S_{nei}\cup(s_k)$
9: $A_s=A_s - As_k$ // As_k is set of all the objects covered by sensor s_k
10: *end if*
11: *if* A_s= ={}
12: *break*
13: *end if*
14: end for
15: *if* S_{nei} is completely covering all the points covered by sensor *s*
16: $S_{cover}=S_{cover}\cup(S_{nei})$ // *s* decides to be *turn off* in this round
17: *else*
18: S_{cover} = $S_{cover}\cup(s)$ // *s* decided to be *turn on* in this round
19: *end if*
20: end for
21: *return* S_{cover}

Fig. 3 Algorithm to find sensor cover in I-CPCP

In Fig. 3, line 1 contains all the remaining sensors after deciding cluster heads in Fig. 2. The respective weighted costs are assigned to each sensor (line 2). The set of active sensors (sensor cover) is generated through line 4-20. To do so, for each sensor **s**, the set of all neighboring sensors with lower weighted cost are selected (line 6-14). If these neighbors' collectively covering all the objects covered by sensor **s** (line 15), then sensor **s** can go to sleep state (line 16), otherwise, it is included in current sensor cover (line -18). Thus, at the end of Algorithm in Fig. 3, a cover set is generated which completely covers all the objects in the given field.

4.3 Phase III: Routing

This phase describes how the collected data is to be transferred to the base station. The cluster members send their data to their respective cluster heads which in turn send to the base station. In case, the cluster heads are not able to send the data directly to base station, then they send via those cluster heads having least associated coverage aware cost with them. In this phase, the cluster members do not take part in data communication.

5. Simulation

The main objective of the sensor cover generation is to save the network energy so that the network lifetime can be prolonged. One of the most commonly used network lifetime is defined [15] as the time of start-up of the network till some object in the given area becomes uncovered.

For simulation, we have considered the square area of size 200M*200M in which there are 500 objects placed randomly. We have evaluated the results by deploying the sensors ranging from 50 to 250 with fix sensing range R_{sen}=70M. We assume that the base station is fixed and located outside the network. Sensors are generated in terms of their coordinates which are generated by Pseudo-random number generation routine. All simulations were performed in MATLAB (R2009) on core i3 with 2.10GHz processor and 4GB RAM system

5.1 Energy Model

In the simulation, we compute the transmission and receiving energy consumption based on the free-space energy model defined in [20].

The energy required to transmit (E_{tx}) and receive (E_{rx}) a P-bit packet is given below.

$$E_{tx}=P\times(E_{amp}+\boldsymbol{\varepsilon}_{fs}\times d^{n}) \tag{10}$$

$$E_{rx}=P\times E_{amp} \tag{11}$$

E_{amp} and $\boldsymbol{\varepsilon}_{fs}$ are the parameters of the transmission/reception circuitry, and n is the path-loss exponent, listed in Table 1. By putting all the values specified in Table 1 for simulation in equations (10) and (11), we obtain the transmission and receiving energy consumed as follows:

$$\begin{aligned} E_{tx}&=30\times8(50\times10^{-9}+10\times10^{-12}\times(70^{2})\ J \\ &=240(50\times10^{-9}+49\times10^{-9})J \\ &=0.24\times10^{-4}J \end{aligned} \tag{12}$$

Similarly,

$$E_{rx}=0.12\times10^{-4}J \tag{13}$$

Therefore, energy consumed by a sensor node to transmit a message of 30 bytes is 0.24×10^{-4} J. In the same way, the total energy consumed by a cluster head to receive the message and to transmit further to the base station is $E_{tx}+E_{rx.}$ By adding equations (12) and (13), we obtained total energy consumed (14) as follows:

$$E_{tx}+E_{rx}=0.36\times10^{-4}J \tag{14}$$

Table 1 Parameters and their acronyms used in simulation

Parameter	Value
Network area($M\times M$)	(200M×200M)
Tx/Rx Electronic constant (E_{amp})	50 nJ/bit
Amplifier constant (ε_{fs})	10 pJ/bit/m^2
Cluster head Energy Threshold (E_{th})	10^{-4} J
Transmission range (d)	70M
Path loss Exponent (n)	2
Packet size (P)	30 bytes
Packet rate (B)	1 Packet/Sec

5.2 Simulation Results

Here, our objective is to discuss lifetime achieved and execution time taken in our heuristic. We compare the lifetime achieved and execution time of our heuristic with that of generic CPCP [11] by considering homogenous and heterogeneous WSNs. We discuss the network lifetime and execution time by considering four different coverage aware costs as discussed in [11]. These costs include *Cmw*, *Cws*, *Cea* and, *Ccc*. In homogenous WSNs, we have taken the initial energy of the nodes as $3*10^{-4}J$. Here, the network lifetime and the execution time of our heuristic and CPCP [11] with respect to all costs, i.e., (*Cmw*, *Cws*, *Cea* and *Ccc*) have been computed as shown in Fig. 4 to Fig. 7 respectively.

As evident from these Fig. 4 to Fig. 7, our heuristic provides better networks lifetime as compared to CPCP for all the costs. As the number of sensor grows, the gap in lifetime achieved by CPCP and our heuristic increases because, due to increase in number of sensors, connectivity between neighbour sensors is also increases. Therefore, our heuristic selects less number of sensors for coverage phase which result in increased total network lifetime.

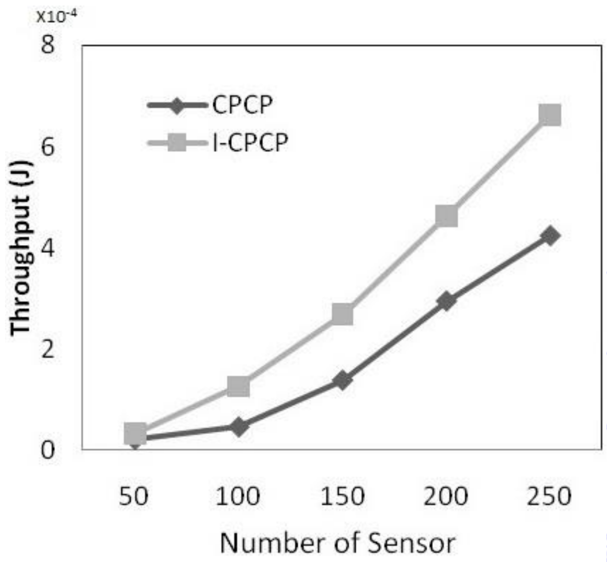


Fig. 4 Network lifetime (homogenous WSNs) achieved by proposed I-CPCP and CPCP [11] with cost *Cmw*

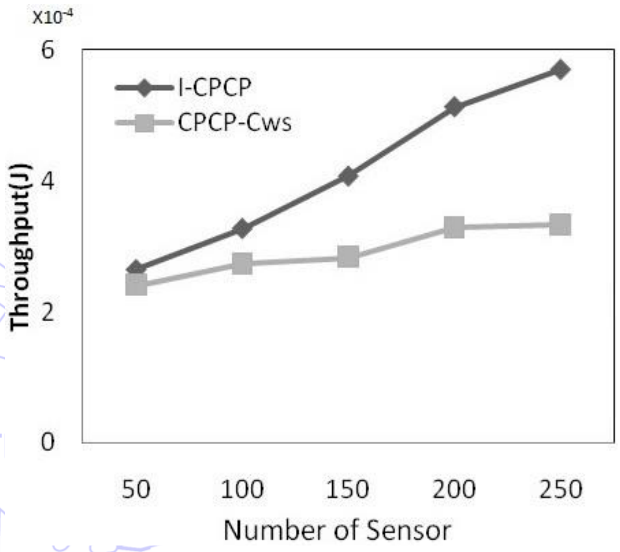


Fig. 5 Network lifetime (homogenous WSNs) achieved by proposed I-CPCP and CPCP [11] with cost *Cws*

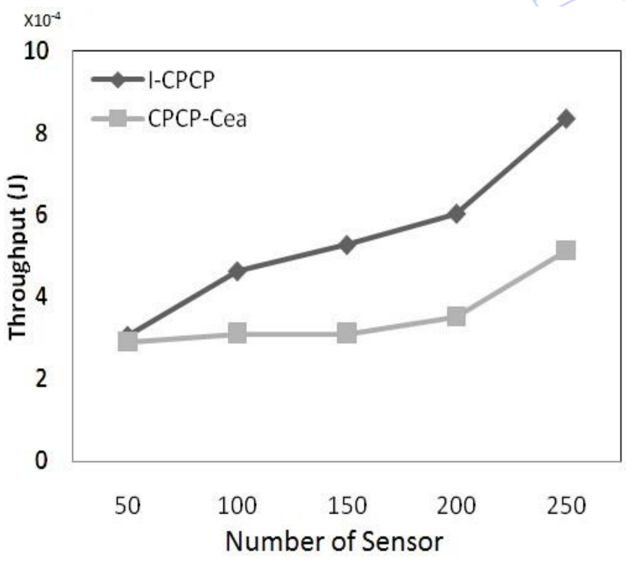


Fig. 6 Network lifetime (homogenous WSNs) achieved by proposed I-CPCP and CPCP [11] with cost *Cea*

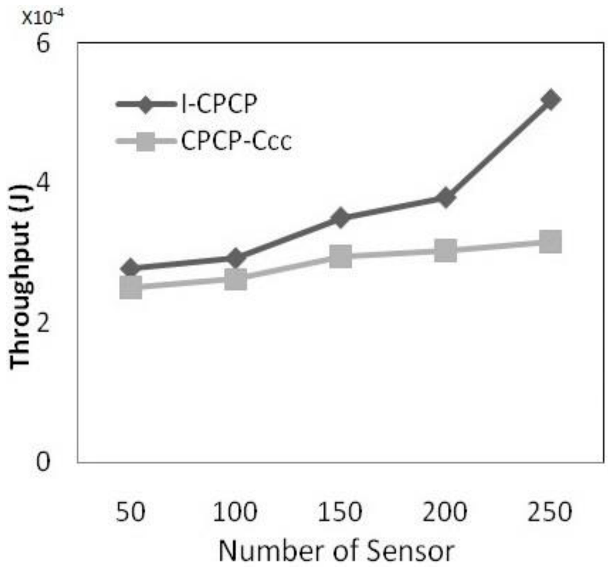


Fig. 7 Network lifetime (homogenous WSNs) achieved by proposed I-CPCP and CPCP [11] with cost *Ccc*

In proposed heuristic I-CPCP, we perform minimalization process to reduce number of sensors in sensor cover that in turn reduces the energy usages. The minimalization process needs some time for its execution, however, the total execution time needed by our heuristic is comparable to that of CPCP [11] as shown in Table 2.

Table 2 Execution time taken (in seconds) by proposed heuristic (I-CPCP) and CPCP [11] in homogenous WSNs

Sensor	Cmw		Cws		Ccc		Cea	
	CPCP	I-CPCP	CPCP	I-CPCP	CPCP	I-CPCP	CPCP	I-CPCP
50	0.11	0.15	0.28	0.28	4.25	4.45	0.26	0.33
100	0.14	0.18	0.42	0.55	9.11	9.65	0.51	0.92
150	0.21	0.32	0.63	0.92	13.95	16.2	0.67	1.35
200	0.35	0.49	0.9	1.46	20.13	23.53	0.81	1.62
250	0.45	0.61	1.16	2.07	26.39	36.51	1.1	2.15

In case of heterogeneous network, we have allocated initial energy to sensors by generating randomly in the range $1*10^{-4}J$ to $5*10^{-4}J$. As shown in Fig. 8 to Fig. 11, in this case also our heuristic (I-CPCP) has much better network life time as compared to that of CPCP [11] for all the four coverage aware costs. Again, increasing the number of nodes, it behaves similar to homogenous network.

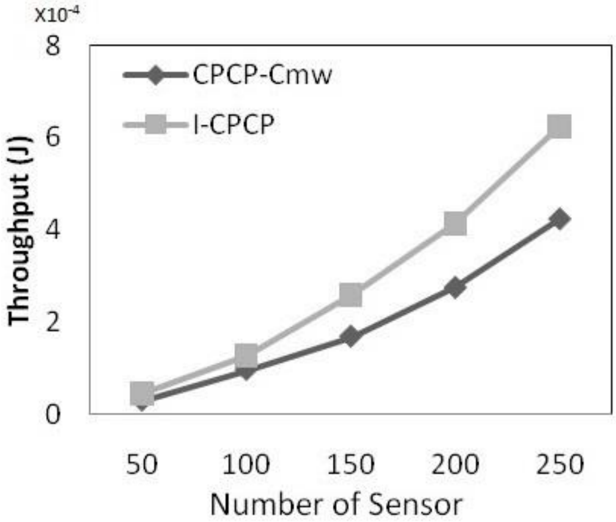


Fig. 8 Network lifetime (heterogeneous WSNs) achieved by proposed I-CPCP and CPCP [11] with cost *Cmw*

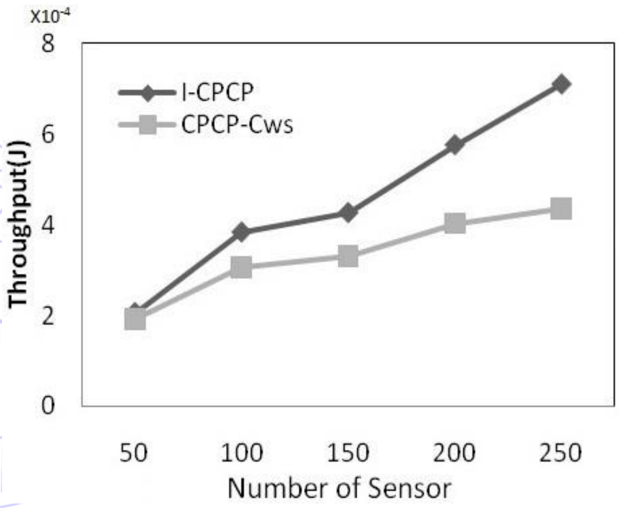


Fig. 9 Network lifetime (heterogeneous WSNs) achieved by proposed I-CPCP and CPCP [11] with cost *Cws.*

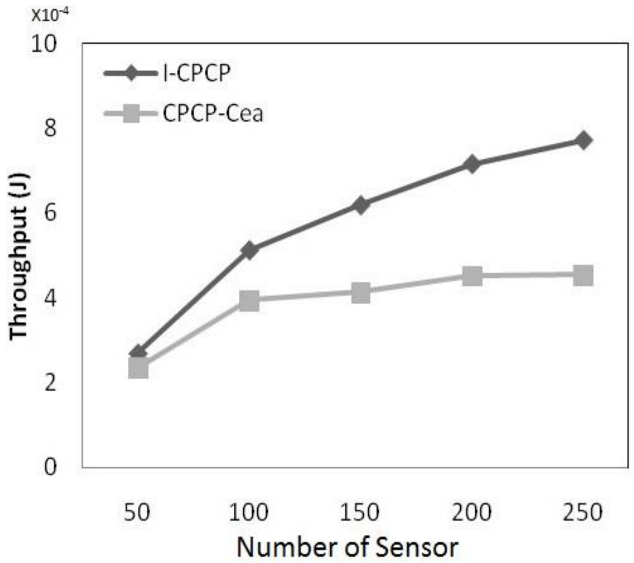


Fig. 10 Network lifetime (heterogeneous WSNs) achieved by proposed I-CPCP and CPCP [11] with cost *Cea.*

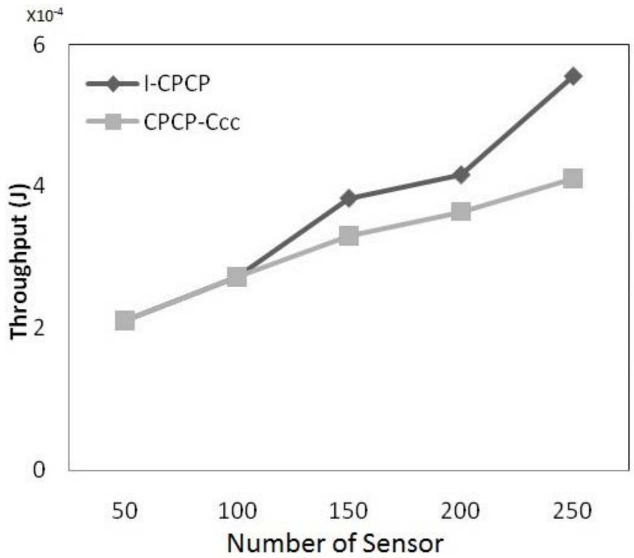


Fig. 11 Network lifetime (heterogeneous WSNs) achieved by proposed I-CPCP and CPCP [11] with cost *Ccc.*

The behaviour of execution time in this case is quite similar as that of the homogenous network as shown in Table 3. Thus, our heuristic provide much longer network lifetime without requiring much overhead in execution time. We may conclude that our heuristic has better performance as compared toRef. [11].

Table 3 Execution time taken (in seconds) by proposed heuristic (I-CPCP) and CPCP [11] in heterogeneous WSNs

Sensor	*Cmw*		*Cws*		*Ccc*		*Cea*	
	CPCP	I-CPCP	CPCP	I-CPCP	CPCP	I-CPCP	CPCP	I-CPCP
50	0.10	0.13	0.18	0.22	3.43	3.46	0.2	0.29
100	0.14	0.17	0.42	0.56	7.79	8.46	0.46	0.82
150	0.21	0.28	0.61	0.89	13.4	15.87	0.66	1.38
200	0.32	0.39	0.86	1.25	19.66	24.08	0.91	1.78
250	0.45	0.61	1.17	1.76	24.78	31.71	1.12	2.52

6. Conclusions

In this paper, we have discussed an improved coverage preservation clustering protocol (I-CPCP)) in clustered wireless sensor networks. We explore various weighted coverage costs in order to select cluster head and to form sensor covers to monitor the given region for longest possible period. In the proposed protocol, we jointly consider residual energy as well as coverage redundancy of sensors to generate sensor cover and cluster heads which results in improved network lifetime as compared to those approaches [2, 3, 4, and 5] which only consider either at a time. Further, we minimalize the generated sensor covers so that only minimum numbers of sensors are used in each sensor cover. Due to this minimalization process, our proposed protocol (I-CPCP) outperforms generic CPCP [11] in terms of lifetime achieved in homogeneous as well as in heterogeneous network. In our future work, we may extend this improved protocol for target connected coverage problem, target Q-coverage problem and target coverage with multiple sensing ranges. We will also implement I-CPCP and CPCP protocol on specific sensor test- bed with respect to all coverage aware costs discussed in this paper.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A survey on sensor networks,” IEEE Communications Magazine, vol. 40, no. 8, pp. 102-114, 2002.
[2] M. Cardei and D. Z. Du, “Improving wireless sensor network lifetime through power aware organization,” ACM Wireless Networks, vol. 11, pp. 333–340, 2005.
[3] M. Cardei, M. T. Thai, Y. Li, and W. Wu, “Energy-efficient target coverage in wireless sensor networks,” Proc. IEEE Infocom, IEEE press, March. 2005, pp. 1976-1984.
[4] S. Mini, S. K. Udgata, and S. L. Sabat, “Sensor deployment and scheduling for target coverage problem in wireless sensor networks,” IEEE SENSORS Journal, vol. 14, no. 3, pp. 636-644, 2014.
[5] Manju and A. K Pujari, “High-Energy-First (HEF) heuristic for energy efficient target coverage problem,” International Journal of Adhoc sensor and Ubiquitous Computing, vol. 2, no. 1, March 2011.
[6] A. Panconesi and M. Sozio, “Fast distributed scheduling via primal-dual,” Proc. the twentieth annual symposium on Parallelism in algorithms and architectures, 2008, pp. 229-235.
[7] V. Kumar, S. Jain, and S. Tiwari, “Energy efficient clustering algorithms in wireless sensor networks, A survey,” IJCSI International Journal of Computer Science Issues, vol. 8 , no. 2, pp. 259-268, October 2011.
[8] M. Chatterjee, S. Das, and D. Turgut, “WCA: A weighted clustering algorithm for mobile ad hoc networks,” Journal of Cluster Computing, Special issue on Mobile Ad hoc Networking ,vol. 5, no. 2, pp. 193-204, April 2002.
[9] S. Yi, J. Heo, and Y. Cho, “PEACH: power-efficient and adaptive clustering hierarchy protocol for wireless sensor networks,” Computer Communications, vol. 30, no. 14, pp. 2842-2852, 2007.
[10] Z. Liu, Q. Zheng, L. Xue, and X. Guan, “A distributed energy-efficient clustering algorithm with improved coverage in wireless sensor networks,” Future Generation Computer System, vol. 28, no. 5, pp. 780-790, May 2012.
[11] S. Soro and W. B. Heinzelman, “Cluster head election technique for coverage preservation in wireless sensor networks,” Ad Hoc Networks, vol. 7, no. 5, pp. 955-972, July 2009.

[12] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," IEEE Transactions on Wireless Communications, vol. 1, no. 4, pp. 660−670, 2002.

[13] S. Lindsey and C. S. Raghavendra, "PEGASIS: Power-efficient gathering in sensor information systems," Proc. IEEE Aerospace Conference, 2002, pp. 1125–1130.

[14] O. Younis and S. Fahmy, "HEED: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks," IEEE Transactions on Mobile Computing, vol. 3, no. 4, pp. 660-669, 2004.

[15] S. Gamwarige and C. Kulasekere, "An algorithm for energy driven cluster head rotation in a distributed wireless sensor network," Proc. the International Conference on Information and Automation, pp. 354-359, 2005.

[16] M. Qin and R. Zimmermm, "An energy-efficient voting-based clustering algorithm for sensor networks," Proc. Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, IEEE Press, May. 2005, pp. 444-451.

[17] G. Smaragdakis, I. Matta, and A. Bestavros, "SEP: a stable election protocol for clustered heterogeneous wireless sensor networks," Proc. the 2nd International Workshops on Sensor and Actor Network Protocols and Applications. Boston, USA, IEEE Press, 2004, pp. 223-233.

[18] D. Kumar, T. C. Aseri, and R. B. Patel, "EEHC: energy efficient heterogeneous clustered scheme for wireless sensor networks," Computer Communications, vol. 32, no. 4, pp. 662-667, March 2009.

[19] H. Zhen, Y. Li, and G. J. Zhang, "Efficient and dynamic clustering scheme for heterogeneous multi-level wireless sensor networks," Acta Automatica Sinica, vol. 39, no. 4, April 2013.

[20] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application specific protocol architecture for wireless microsensor networks," IEEE Transactions on Wireless Communications, vol. 1, no. 4, 2002.

[21] M. Chaudhary and A. K. Pujari, "Q-coverage problem in wireless sensor networks," Proc. Int. Conf. Distrib. Comput. Networks, Springer Berlin Heidelberg press, 2009, pp. 325–330.

[22] D. Bajaj and Manju, "Maximum Coverage Heuristic (MCH) for target coverage problem in WSN," Proc. IACC, IEEE press, Feb. 2014, pp. 300-305.

[23] S. Venkatesh and K. M. Mehata, "A competent fault tolerant system using simulated annealing approach for target tracking in wireless sensor networks," Proc. Computer Communication and Informatics (ICCCI), 2014 International Conference on, IEEE press, Jan. 2014, pp. 1-6.

[24] J. Kim, H. W. Lee, and S. Chong, "TAES: Traffic-aware energy-saving base station sleeping and clustering in cooperative networks," Proc. Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt), 2015, 13th International Symp, IEEE press, May. 2015, pp. 259-266.

[25] S. H. Seo, J. Won, S. Sultana, and E. Bertino, "Effective key management in dynamic wireless sensor networks," in Information Forensics and Security, IEEE Transactions on, vol. 10, no. 2, pp. 371-383, 2015.

[26] M. Collotta and G. Pau, "A novel energy management approach for smart homes using Bluetooth low energy," IEEE Journal on Selected Areas in Communications, vol. 33, no. 12, pp. 2988-2996, 2015.

[27] Y. C. Lee, H. Y. Lai, and P. J. Lee, "Cryptanalysis and improvement of the robust user authentication scheme for wireless sensor networks," International Journal of Engineering and Technology Innovation, vol. 2, no. 4, pp. 283-292, 2012.

[28] F. Carrabs, R. Cerulli, C. D'Ambrosio, and A. Raiconi, "A hybrid exact approach for maximizing lifetime in sensor networks with complete and partial coverage constraints," Journal of Network and Computer Applications, vol. 58, pp. 12-22, 2015.

[29] D. Izadi, J. Abawajy, and S. Ghanavati, "An alternative clustering scheme in WSN," IEEE SENSORS Journal, vol. 15, no. 7, pp. 4148-4155, 2015.

International Journal of Engineering and Technology Innovation, vol. 6, no. 1, 2016, pp. 30 - 54

Performance Comparison of A New Non-RSSI Based Wireless Transmission Power Control Protocol with RSSI Based Methods: Experimentation with Real World Data

Debraj Basu*, Gourab Sen Gupta, Giovanni Moretti, Xiang Gui

School of Engineering and Advanced Technology, Massey University, New Zealand

Received 13 August 2015; received in revised form 03 November 2015; accepted 11 December 2015

Abstract

In this paper, simulations with MATLAB are used to compare the performance of a RSSI-based output power control with non-RSSI based adaptive power in terms of saving energy and extending the lifetime of battery powered wireless sensor nodes. This non-RSSI (received signal strength indicator) based adaptive power control algorithm does not use RSSI side information to estimate the link quality. The non-RSSI based approach has a unique methodology to choose the appropriate power level. It has drop-off algorithm that enables it to come back from a higher to a lower power level when deemed necessary. The performance parameters are compared with the RSSI-based adaptive power control algorithm and fixed power transmission. In order to evaluate the protocols in the real world scenarios, RSSI data from different indoor radio environments are collected. In simulation, these RSSI values are used as an input to the RSSI based power control algorithm to calculate the packet success rates and the energy expenditures. In this paper we present extensive analysis of the simulation results to find out the advantages and limitations of the non-RSSI based adaptive power control algorithm under different channel conditions.

Keywords: wireless sensor network, energy consumption optimization, adaptive power control

1. Introduction

The proliferation of low power wireless sensor networks and their discreet presence have introduced a new paradigm in data collection and analysis of target parameters in both indoor and outdoor environments. This has been differently named in the literature and the industry, like 'invisible", "pervasive" or "ubiquitous" [1] computing. Others prefer to refer to it as "ambient intelligence" [2]. The broad idea is that there will be sensors that are able to exchange information with a certain base station or hub and perform an assigned task. The sensors, the computational and the communication units, along with the hub, form the ubiquitous sensor network (USN). The term ubiquitous is applied to the collection and utilization of information in real time, at any-time and any-where. The technology has enormous potential and a wide range of applications such as, environmental monitoring, health monitoring for assisted living (smart home environments) and industrial and plant monitoring (Industrial automation).

1.1 Related work in transmission power control for energy efficiency

Power saving approaches can be broadly classified into media access control (MAC) layer solutions and network layer

* Corresponding author. E-mail address: d.basu@massey.ac.nz

solutions [3]. Network layer solution means that the different transmission parameters can be modified to achieve the set goal. This paper discusses the use of received link quality information (RSSI/LQI) to adjust output transmission power.

1.2 RSSI/LQI based power control algorithms for energy efficiency

The RSSI-based power control approaches is guided by closed loop control between the transmitting node and the receiving base station mechanism. RSSI is a measurement of signal power and which is averaged over 8 symbols of each incoming packet [4]. On the other hand, LQI is usually vendor specific and is measured based on the first eight symbols of the received packet as a score between 50 and 100 [5].The general steps are described below as

- The transmitter sends packet at an updated power level to the receiver
- Receiver measures the RSSI
- If the RSSI is below the threshold that is required for faithful packet delivery, then the receiver sends the control packet with the new transmission power level.
- At the transmitter, the control packet is received and the current power level is updated for packet delivery

During initialization phase, the transmitter needs to know the power level at which it should transmit to successfully deliver the packet. In this phase, the transmitter sends several packets at all its available power levels. In return, it receives RSSI values for each power levels. Based on the mapping of the RSSI and the output power level, the transmitter selects the required power level.

In paper [4], Shan Lin et al. have introduced adaptive transmission power control (ATPC) that maintains a neighbor table at each node and a feedback loop for transmission power control between each pair of nodes. ATPC provided the first dynamic transmission power algorithm for WSN that uses all the available power output levels of CC2420 [6].

Practical-TPC [7] is a receiver oriented protocol that is considered robust in dynamic wireless environments and uses packet reception rate (PRR) values to compute the transmission power that should be used by the sender in the next attempt. While ATPC uses all 32 power levels, there are some algorithms that divide these 32 power levels into 8 levels, as in [3]. The work described is this paper aims to avoid the need for such probe packets and their associated energy cost. ART (Adaptive and Robust Topology control) protocol [8] has been designed for complex and dynamic radio environments. It adapts the transmission power in response to variation in link quality or degree of contention. Analysis of the paper has suggested that RSSI and LQI (Link Quality Indicator) may not be good or the most reliable indicators of link quality, especially in robust indoor radio environment.

Paper [3] also has an initialization phase and a maintenance phase while adjusting transmission power. In the initialization phase, each of the sensor nodes uses the 8 power levels of CC2420 to send 100 probe packets in each of the power levels. It sets the packet delivery ratio (PDR) threshold to 80% instead of the RSSI threshold to determine the minimum power level with which the nodes must communicate with each other. In the maintenance phase, the aim is to adjust the transmission power level with the changing environment.

REAL (reliable energy adept link-layer) [9] protocol uses error correction mechanism to maintain reliable communication. It chooses its data recovery strategy based on the overall information distortion and the available energy at a sensor node.

The data recovery actions have three options to choose from. They are

- Use of error correction code to recover the original data packet at the receiver
- Retransmit when the error correction mechanism has failed due to severe distortion
- Drop some packets to save energy for transmission of higher priority packets

In [10], the approach is similar to ATPC where the power-distance table is maintained at each node. The distance is the minimum power of one node with the neighboring node. In multi-hop wireless sensor network, optimization of the transmission power is, therefore, the shortest path problem based on the power-distance relationship. In [11], the authors have proposed a power control algorithm in which each sensor node also uses beacon messages to determine its neighbors and the corresponding minimum transmission power. After the neighbors are discovered, the adaptive algorithm finds the optimal power so that it is able to meet its target of communicating with a given number of neighboring nodes. Authors have combined dynamic transmission power control of the link layer protocol with the reduction of duty cycle of MAC layer to save energy.

Paper [12] has introduced the term "link inefficiency" while characterizing the link quality metrics of energy constrained wireless sensor nodes. Link inefficiency is defined as the inverse of the packet success probability as it represents the mean number of transmissions for a successful transmission at a given time. The expected energy consumption is, therefore, proportional to the link inefficiency. This paper proposes the time average energy consumption as the cost metrics.

The application of an adaptive power control algorithm for IEEE 802.11 in the technical report of [13] aims to modulate the transmit power based on the distance between the communicating nodes to the minimum level such that the destination node still achieves correct reception of a packet despite intervening path loss and fading. It used a radio module with configurable output power level (0 to 25 dBm). The receiver only sends the control packet containing the optimal transmission power level when there are significant changes in the RSSI values.

RSSI/LQI based adaptive power control algorithms are an attractive alternative to save energy. It is to be noted that these algorithms are mainly designed for multi-hop network where each sensor node broadcast beacon packets and discover its neighbor to which it can transmit at minimum power. However, there are two factors that are worth considering. They are

- There is an initial overhead cost for building up the RSSI vs. Power level table.
- In case the sensor is mobile, the refreshing frequency of the table becomes crucial and that also adds up to the cost.
- Even when the sensors are stationary, there is no clear indication in any of the papers ([3-4], [6-13]) as to what would be the ideal channel sampling frequency that would optimise energy efficiency.

Most of the network level power control algorithms that are discussed above use link quality information (RSSI or LQI) for adjusting the output power. The adaptive power control algorithm has a unique channel estimation method without RSSI side information. Therefore, this algorithm is best suited for those radio modules that do not support RSSI values. The nRF24L01p radio transceiver module from Nordic Semiconductor Inc. [14] has configurable four output power levels that do not provide RSSI information. The output power modes and their current ratings are presented in Table 1.

Table 1 Operational modes and current consumptions of NRF24L01+

Operational mode	Current consumed mA
Transmission @ 0 dBm output power (MIN)	11.3
Transmission @ -6 dBm output power (LOW)	9
Transmission @ -12 dBm output power (HIGH)	7.5
Transmission @ -18 dBm output power (MAX)	7

The transceiver can transmit at four power levels: -18 dBm, -12 dBm, -6 dBm and 0 dBm. In general, a wireless transceiver has different modes of operation.

Among other methods of energy aware data transmission in WSN or mobile application, the work of Zhang et.al has been noteworthy. In [15], the authors have proposed a novel topology control approach for wireless sensor networks (WSNs) where the edge weight and vertex strength take sensor energy, transmission distance, and flow into consideration. Zhang and Liang have proposed a novel method of service aware computing for uncertain mobile applications to ensure QoS for these devices supporting various applications [16]. In industrial applications of WSNs, an energy-balanced routing method based on forward-aware factor (FAF-EBRM) has been proposed by the authors in [17]. In this multi-hop routing algorithm, the next node is selected based on information about the link weight and forward energy density. In [18], authors have designed and implemented a solution of embedded un-interruptible power supply (UPS) system forward for long-distance monitoring and controlling of UPS based on Web. Zhang et.al, have proposed a novel multicast routing method with minimum transmission for WSN of cloud computing service [19].

In [20], Zhang has proposed a fusion decision method to support an attentive mobile learning paradigm. The learning method tracks the users' movement without any active devices. The objective is to achieve seamless mobility for mobile services, especially mobile web-based learning. In the RFID (radio-frequency identification) domain, one of the key issues is the packet losses due to collision when the RFID tags transmit within the collision window (time). In [21], a novel anti-collision approach has been proposed by Zhang et.al. In this method, a mapped correlation of the IDs of the RFID tags is used to increase the association between the tags. In [22], the authors have proposed agent-based proactive migrating method with service discovery and key frames selection strategy. The designed system is convenient to work and use during mobility, and which is useful for mobile user in the big data environments (BDE).

2. Adaptive transmission power control methods: Non-RSSI based and RSSI-based

This section discusses about the novel transmission power control protocol that does not use RSSI data for channel estimation and the RSSI-based power control method.

2.1 Non- RSSI/LQI based channel estimation and power control algorithms for energy efficiency

The non-RSSI based channel estimation and output power control algorithm is proposed in [23-24]. The basis of this lightweight adaptive algorithm is the states where each state represents one cycle of packet transmission. In each state there are output power levels in increasing order which can be used by the transmitter. State transition occurs depending on the power level at which the transmission is successful or failed. State 4 uses only the maximum power level and is allowed to transmit 4 times. There is no direct transition from state 4 to state 1 or 2. Similar conditions hold true when transiting from 3. The most energy efficient state is 1. The more it stays in state 1, the more it saves energy. State 4 is where the maximum energy may be used to transmit the packet. The adaptive algorithm is designed in such a way that it takes into account of performances in each state. It also has a unique drop-off algorithm that allows it to drop down to a lower state when deemed necessary. It is guided by the drop-off factor R. In this paper, R values of 0.01, 0.05, 0.1, 0.5 and 1 are used. Higher value of R means higher rate of drop-off. Fig. 1 shows the state transition diagram of the adaptive power control algorithm. State transition occurs depending on the power level at which the transmission is successful or has failed.

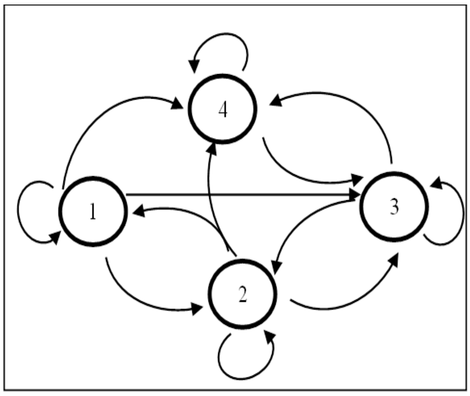


Fig. 1 State transition diagram of the adaptive algorithm

The objective of the adaptive power control algorithm is to respond to the packet error rate and move to a new state with different retry limits. The adaptive algorithm is designed in such a way that it takes into account the performance in each state. Each state has a different retry limit. Increasing state number indicates poorer channel quality. The proposed adaptive algorithm does not allow retransmission in the same power level except when it is in state 4 and transmitting at 0 dBm. When the system is in state 4, it is considered the worst channel condition and three retries are allowed. The retry limit of state 1 is three. However, the retry limit of states 2 and 3 have been set at 2 and 1. The asymmetry is because the increase in the retry limit in states 2 and 3 can increase the current consumption while only marginally improving the packet success rate.

Table 2 shows the available power levels based on the states. Transmission starts at the lowest available power level of that particular state. The transmitter can be in any one of the states during the start of transmission of a packet. There are two separate algorithms that determine the state transitions, one from a lower state to higher state and the other from a higher to lower states. The logic to transit to lower states also includes situations when it remains in the same state or transit to a lower state.

Table 2 States, power levels, and retry limits

State	1	2	3	4
Available power levels	Minimum (M)			
	Low (L)	Low (L)		
	High (H)	High (H)	High (H)	
	Maximum (X)	Maximum (X)	Maximum (X)	Maximum (X)
Number of retries	3	2	1	3

Table 3 describes the state transition matrix when state level goes up. All the state transition decisions depend on the success or failure of the packet being transmitted to the destination hub.

Table 3 State transition matrix when state levels go up

		Next State			
		1 (MLHX)	2 (LHX)	3 (HX)	4 (X)
Current State	1 (MLHX)	Succeed at level M	Succeed at level L	Succeed at level H	Failed or Succeed at level X
	2 (LHX)	Not applicable	Not applicable	Succeed at level H	Failed or Succeed at level X
	3 (HX)	No transition	Not applicable	Not applicable	Failed or Succeed at level X
	4 (X)	No transition	No transition	Not applicable	Not applicable

Table 4 State transition matrix when state levels go down

		Next State			
		1 (MLHX)	2 (LHX)	3 (HX)	4 (X)
Current State	1 (MLHX)	Success at state M	Not applicable	Not applicable	Not applicable
	2 (LHX)	Probabilistic model that depends on the number of successes in level L	Probabilistic model that depends on the number of successes in level L	Not applicable	Not applicable
	3 (HX)	No transition	Probabilistic model that depends on the number of successes in level H	Probabilistic model that depends on the number of successes in level H	Not applicable
	4 (X)	Not applicable	Not applicable	Probabilistic model that depends on the number of successes in level X	Probabilistic model that depends on the number of successes in level X

Table 4 describes the state transition logic when state level goes down. The primary objective of the adaptive algorithm is to save energy by transmitting at a power level that is enough to send the packet successfully through the channel. For example, when the system is in state 4, it is transmitting at the maximum power. With time, the channel condition can improve and packet can be successfully transmitted at a lower power level. If the system drops down to state 3, the transmission starts at a lower power level. This drop-off from a higher state to a lower state is determined by a drop-off algorithm which is probabilistic in nature.

In the proposed adaptive algorithm, the drop-off or the back-off process is dependent on the number of successes (S) in the higher power level and a drop-off factor (R). By default, the drop-off factor is 1. The probability of the system to drop-off to a lower power level is represented by Equation (1).

$$P_{(drop\text{-}off)} = 1 - e^{(-RS)} \tag{1}$$

Here, $P_{drop\text{-}off}$ = probability of drop-off

S = the number of successes in that power level of the higher state

R = drop-off factor

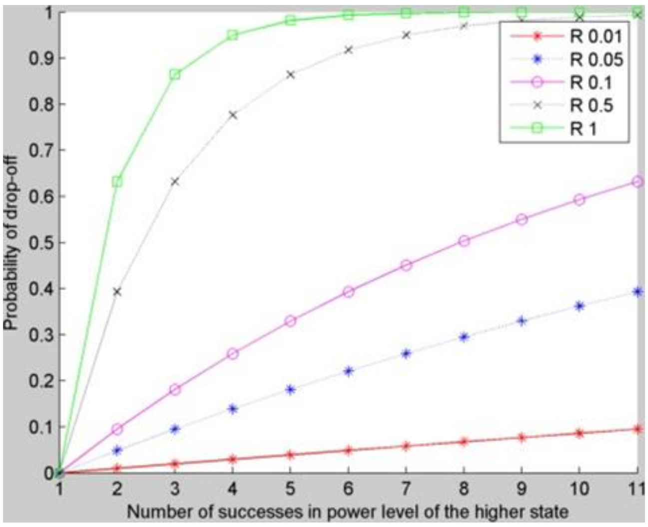


Fig. 2 The curves behave differently depending on the value of R. A low R value indicates slow back off while a high R indicates fast back off. When the number of successes is 0, the probability of transition is 0. This drop-off algorithm takes into account of all the previous successes indicating that it also uses past history while dropping-off

The plots in Fig. 2 show the state transition probability based on different values of R. When there is a state change, the value of S is reset to 0. Overall, the value of R indicates as to how fast the system will fall from a higher state to a lower state. When there is no success, the probability of state transition is 0, meaning that there will be no state transition. At the same time, when the number of successes is too high, it converges to 0.

Back-off algorithms are extensively used in data communication (both wired and wireless) by MAC protocols to resolve contention among transmitting nodes to acquire channel access. In a MAC protocol, the back-off algorithm chooses a random value from the range [0, CW], where CW is the contention window size. The contention window is usually represented in terms of time slots.

The number of time slots to delay before the nth retransmission attempt is chosen as a uniformly distributed random integer r in the range $0 < r < 2k$.

where $k = \min(n, 10)$, 10 is the maximum number of retries allowed.

The nth retransmission attempt also means that there have been n collisions. For example, after the first collision, it has to retransmit. Based on the back-off algorithm, the sender will choose between 0 and one time slot for the retransmission. After the second collision, the sender will wait anywhere from 0 to three time slots (inclusive). After the third collision, the senders will wait anywhere from 0 to seven time slots (inclusive), and so forth. As the number of retransmission attempts increases, the number of possibilities for delay increases exponentially [25-26].

Similarly, an exponential operator is used in this novel adaptive algorithm to decide to switch from a higher state to a lower state. The drop-off algorithm is dynamic as it re-evaluates at every successful transmission. It gets reset to 0 when it leaves the state and jumps to a lower state and starts a new packet transmission at a lower power level.

In this paper, this protocol is compared with the standard RSSI-based channel estimation and power control algorithm (ATPC) using real world RSSI data when sensors are stationary. The next section explains the ATPC protocol in details.

2.2 ATPC: Adaptive Transmission Power Control for wireless sensor networks

The first adaptive power control protocol for wireless sensor network was proposed in [4]. Results show that the ATPC only consumes 53% of the transmission energy of the maximum transmission power solutions and 78.8 % of that of network level transmission power solutions [3]. In ATPC, the sensor nodes build a model for each of its neighboring nodes that describe the correlation between the transmission power and the link quality. The radio link quality varies over time and with environment. The objective of the ATPC protocol is to find out the minimum transmission power level to maintain a good quality link (~PSR > 98%) and dynamically change the transmission power level over time to address the time varying nature of the wireless channel. Each node sends beacon packets at different transmission power level to the neighboring node and makes note of the RSSI value that it receives on the feedback path. Based on this information, the node builds a predictive model and uses least square approximation method to calculate the desired transmission power level.

In run time, it also monitors the link quality by using the information of RSSI and setting the upper and the lower limits to the link quality estimator. As long as the RSSI value is steady and within the range, the transmitter is not required to adjust the power level. Therefore, the upper and lower limits are critical design parameter to make ATPC energy efficient. The other important parameters of ATPC protocol are the link quality threshold, the frequency of transmission power control and the number of sample packets in the setup phase. Based on the empirical findings regarding the temporal variation of the link quality, the paper suggested that one packet per hour would maintain the freshness of the predictive model.

3. Simulation parameters

The general simulation parameters are presented in Table 5.

Table 5 General simulation parameters for comparison of ATPC with non-RSSI based adaptive protocol in MATLAB

Modulation technique	BFSK
Channel data Rate	250 kbps
Maximum Doppler spread	20 Hz
Packet size	41 bytes
Cyclic redundancy check	CRC-16
Multi-path Fading channel model	UMTS Indoor Office Test Environment [27]
E_b/N_0	Derived from the RSSI values
Retry limit in fixed power transmission and ATPC	3
Retry limit to highest power level in state 4 of adaptive protocol	3
RSSI threshold	-90 dBm

3.1 Simulation design of the working principle of ATPC

The ATPC protocol changes its output power based on the RSSI value of the most recent transmitted packet. It has the four power levels to choose from and uses the decision matrix as explained in Table 6. It is known that the minimum RSSI level to maintain high PSR (> 95%) is approximately -90 dBm [28] [5]. Therefore, it is set as the minimum threshold. Here RSSI_TH is the RSSI threshold and RSSI is the received signal strength indicator of the transmitter packet. In this simulation, the performances of the ATPC are observed at channel sampling/scanning interval of 1, 5, 10, 50 and 100 transmissions.

Table 6 Decision matrix table of ATPC on run time

		New transmission power level			
		-18 dBm	-12 dBm	-6 dBm	0 dBm
Current Power level	-18 dBm	RSSI >= RSSI_TH	RSSI_TH - RSSI <= 6 dB	6 dB < RSSI_TH - RSSI < 12 dB	RSSI_TH - RSSI > 18dB
	-12 dBm	RSSI - RSSI_TH >= 6	RSSI - RSSI_TH ~ 0	RSSI_TH - RSSI <=6	RSSI_TH - RSSI > 6
	-6 dBm	RSSI- RSSI_TH >= 12 dB	RSSI- RSSI_TH <=6 dB	RSSI - RSSI_TH ~ 0	RSSI-TH- RSSI >= 6 dB
	0 dBm	RSSI- RSSI_TH >= 18 dB	6 dB < RSSI – RSSI_TH <= 12 dB	RSSI – RSSI_TH <= 6 dB	RSSI - RSSI_TH ~ 0

During each cycle of power control, the ATPC compares the present RSSI with the RSSI_TH. If the difference between the current RSSI and RSSI_TH is negligible or equal to 0, then the new power level is same as the old power level. These conditions are highlighted in bold brown. Only when the current power level is -18 dBm and RSSI is greater than the RSSI_TH, it sticks to -18 dBm.

3.2 Conditions when to ramp up output power:

- When RSSI_TH > RSSI, it is required to ramp up power for subsequent packets.
- When RSSI_TH - RSSI ≤ 6 dB, the output power is incremented by 6 dB. For example, when current output power is -12 dBm, then new power level will be -6 dBm. These conditions are highlighted in bold blue.

- When 6 dB < RSSI_TH - RSSI < 12 dB, then the output power level is incremented by 12 dB. For example, if the current output power is -18 dBm, then new power level will be -6 dBm. These conditions are highlighted in bold black.
- When RSSI_TH - RSSI ≥ 12 dB and the current power level is -18 dBm, then the new power level will be 0 dBm.
- When RSSI_TH - RSSI ≥ 6 dB and the current power level is -12 dBm, then the new power level will be 0 dBm.

3.3 Conditions when to ramp down output power:

- When RSSI - RSSI_TH ≥ 18 dB and output power is 0 dBm, then power level can be decremented by 18 dB to -18 dBm as that will satisfy RSSI ≥ RSSI_TH.
- But when 6 dB < RSSI – RSSI_TH ≤ 12 dB, the output power level decrements to -12 dBm. When RSSI – RSSI_TH ≤ 6 dB, the output power level decrements by 6 dB.
- When RSSI - RSSI_TH ≥ 6 dB and current power level is -12 dBm, then the current power level can be decremented by 6 dB to -18 dBm.
- Finally, when the current power level is -6 dBm and RSSI- RSSI_TH ≤ 6 dB, the power level decrements to -12 dBm, while if RSSI- RSSI_TH ≥ 12 dB, it decrements by 12 dB.

It is known that the minimum RSSI level to maintain high PSR (> 95%) is -90 dBm. Therefore, -90 dBm is set as the minimum threshold. Here RSSI_TH is the RSSI threshold and RSSI is the received signal strength indicator of the transmitter packet. In this simulation, the performances of the ATPC are observed at channel sampling/scanning interval of 1, 5, 10, 50 and 100 transmissions.

4. Performance parameters

The performance parameters are:

- Average cost per successful transmission
- Expected success rate or protocol efficiency [15]

One of the parameters for the optimization is the energy consumed per useful bit transmitted over a wireless link [3, 29]. Similarly in this paper, the cost per successful transmission has been considered.

$$C_{s\_avg} = \frac{C_T}{P_S - P_L} \tag{1}$$

where

C_{s_avg} = average energy cost per successful transmission

C_T = total cost of transmission

P_L = number of lost packets

P_S = Number of packets to send

All cost values are measured in mJoules. The total cost of transmission includes the expenditure for the first transmission attempt of a packet and the subsequent retries if the first attempt fails. The total packet to send count does not include the retry packets. Therefore, the denominator in equation 3 is only the count of successfully transmitted packets.

The expected success rate or efficiency is defined as the expected number of successes and takes into account the average number of retries [3]. It can also be defined as the expected number of successes per 100 transmissions. Mathematically,

$$Succ_{rate} = \frac{P_S - P_L}{P_S + Ret_T} \tag{2}$$

where

$Succ_{rate}$ = expected success rate

Ret_T = total number of retries

Here $P_S - P_L$ = total number of successes (P_{succ}). If both the numerator and denominator are divided by Ps, then in percentage term,

$$Succ_{rate}(\%) = \frac{PSR}{1+Ret_{avg}} \tag{3}$$

where

Ret_{avg} = average number of retries per packet and is defined as

$$Ret_{avg} = \frac{Ret_T}{P_S} \tag{4}$$

Here,

$$PSR = \frac{P_{succ}}{P_S} * 100 \tag{5}$$

This parameter indicates the total number of transmissions (on average) to achieve a given packet success rate (PSR).

5. Collection of the RSSI values

In this paper, we have considered a star topology where all the sensors connect to one base station via a single hop. These sensors are responsible for transmitting, for example, temperature, humidity and occupancy data as well as health related vital information in a smart home environment with assisted living. In industrial set up, these sensors transmit key monitoring parameters like humidity and temperature, valve control position among others. These kind of indoor radio environments pose challenge in terms of reliable data delivery and energy efficiency of the sensor node. This is because the radio signal in indoor environment suffers from fading because of multipath propagation where the radio signal from the transmitter arrives at the receiver through multiple paths.

During the busy hour, there are lots of movement of people in between the hub and the transmitting sensor. These movements induce a time varying Doppler shift on multipath components. Fading effect due to frequency shift of the radio signal cannot be ignored when the sensor is stationary. Besides, there can be temporary signal attenuation if people have gathered around. All these affect the radio link quality over time. During the non-busy hours of the University, fading effect due to movement is minimal while the multipath effect still exists. In order to study the variation of the signal level in these kind of indoor radio environment, the RSSI values from the access points of wireless LAN setup in office environment, commercial setup (shopping center) and social setup (University dining hall) are collected.

The primary source of noise or interference to the signal that is considered in this paper is the signal attenuation because of distance, the partitions in between the transmitter and the base station and the momentary signal fluctuation due to multipath fading as well as movements of people in between the sensor and the base station.

The NetSurveyor tool [30] was used to collect RSSI of beacon signals from a wireless access point that are send every 5 seconds. NetSurveyor is an 802.11 (Wi-Fi) network scanning cum discovery tool that gathers information about nearby wireless access points in real time. These beacon frames are transmitted periodically by the WIFI access points (AP) to announce the presence of a wireless LAN [31-32]. It contains all the information about the network. Since the AP transmits at a fixed power level, any variation in the RSSI value is an indication of the link quality variation, therefore, the received E_b/N_0. There were several APs that were transmitting the radio beacon signal. For the simulation purpose, the RSSI data of that AP was used that was providing the strongest signal. The access point emulates the transmitting sensor while the data collection device (the laptop in this case) acts as the base station.

In simulation, these RSSI values correspond to the minimum output power level (-18 dBm). Hence, it indicates the base channel condition over which power ramping may be required to meet the link quality requirement. This link quality change can be transient or have an effect over a longer period of time. Therefore, the RSSI values can be used to adapt or manipulate the output power. This setup emulates the real world approach of ATPC protocol where the RSSI values from the neighboring node in response to the beacon signal are used to setup the output power level in the initialization phase or during run time. In the single hop topology, it is the hub that piggybacks the RSSI information to the sensor. A 5 second interval between fresh RSSI values indicates that the sensor is transmitting at a rate of 1 packet every 5 seconds and therefore has received the RSSI value as the feedback from the hub.

5.1 Data collection scenarios and calculation of the E_b/N_0

There are two types of RSSI variation scenarios that are investigated. They are collected from three different environments after an interval of 5 seconds. They are

- Three sets of long term data over a period of approximately 10 hours from within University campus building. The distance between the transmitter and receiver is approximately 24 meters.

- Three sets of short term data over the busy period of the day (approximately between 90 minutes and 120 minutes) from town shopping centre. The distance between the transmitter and receiver is approximately 30 meters.

- Three set of short term data over the busy period of the day (approximately between 90 minutes and 120 minutes) from University dining hall. The distance between the transmitter and receiver is approximately 25 meters.

5.2 Images of the mentioned scenarios are added in appendix

RSSI is a measurement of signal power and which is averaged over 8 symbols of each incoming packet [4]. The value of RSSI is dependent on the received power, the channel data rate and the noise spectral density (N¬0). If the received average bit energy is denoted by E_b and channel data rate as R, then by definition, in dBm,

$$RSSI = \frac{E_b}{N_0} + Noise\,Power \tag{6}$$

If the noise floor is assumed to be constant, the RSSI value will depend on the average bit energy and channel data rate. In dBm scale, the relationship between the RSSI and E_b/N_0 is linear with intercept of -119.9978 dBm at x-axis. Different data rate will have different intercept values when the noise floor level is kept constant. The value of -119.9978 is derived from the value of N_0 and channel data rate set at 250 kbps for simulation.

$$Noise\,Power = KTR \tag{7}$$

where

K = Boltzmann’s constant (1.28 x10-23 Joules/Kelvin)

T = Noise temperature in Kelvin (290 K) and

R = 250 kbps

Therefore, the linear relationship between RSSI and E_b/N_0 takes the following form in equation (9) which is derived from equation (7).

$$RSSI = \frac{E_b}{N_0} - 119.9978 \quad (8)$$

6. Comparison of the optimal cost values of ATPC, adaptive power control and fixed power transmission using RSSIdata from three different locations

In section 6, performance parameters of ATPC, fixed power and non-RSSI based adaptive power control algorithms are compared.

6.1 Long term RSSI data collected over a period of approximately 10 hours inside University building

Two sets of data are collected during the period of approximately 10 hours. The busy hour RSSI variation was captured by logging the data between 8:30 a.m. till 5 p.m. The variation of E_b/N_0 over time for one of the data sets is presented in Fig. 3. The RSSI data are collected using a laptop. Using equation (4), the RSSI values are converted to corresponding E_b/N_0 values. The occasional drop to 20 dB is due to fading. Overall, this indicates that the channel link quality is very good.

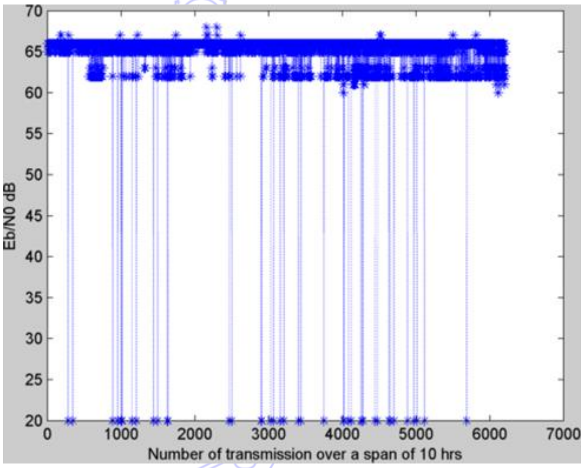


Fig. 3 The variation of the average E_b/N_0 over time. Along x-axis, the numbers of transmissions are noted. The average E_b/N_0 is quite high (>60 dB) and occasionally dropped to 20 dB. Since the distance between the access point and the laptop is constant, the drop in the value is attributed to human movements in between and multipath effects

The normalized frequency distribution plots of the E_b/N_0 of one of the data sets are shown in Fig. 4.

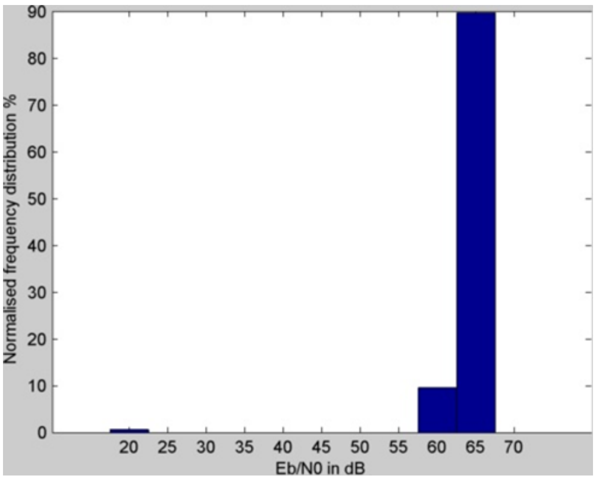


Fig. 4 The frequency distribution plot of the received E_b/N_0 from data set 1 shows that occasionally the signal level has dropped by 30-40 dB, primarily due to multipath fading. Overall, the link quality has been good

The normalized frequency distributions of the other two data sets are also similar to the one shown in Fig. 4. The frequency distribution plots also signify the amount of time (%) the channel link quality has remained above a certain value. In both these figures, a high % of time (>85%) the E_b/N_0 is more than or equal to 60 dB. The PSR and the efficiency values of all the transmission strategies are 100 and higher that 98% respectively and their differences are negligible. Therefore, they are not plotted. Table 7 shows the average values of the performance parameters of the different transmission strategies that are compared in this paper.

Table 7 Average cost, PSR and protocol efficiency of data sets inside University building

ATPC- RSSI based adaptive power control			
Channel scanning frequency	**PSR %**	**Avg. Cost per successful transmission mJ**	**Protocol Efficiency %**
Every time the sensor transmits	100	0.0308	99.10
Every 5 transmissions	100	0.0308	99.08
Every 10 transmissions	100	0.0308	99.07
Every 50 transmissions	100	0.0309	99.08
Every 100 transmissions	100	0.0310	99.15
Non-RSSI based adaptive power control			
Drop-off factor R	**PSR %**	**Avg. Cost per successful transmission mJ**	**Protocol Efficiency %**
0.01	100	0.0322	98.99
0.05	100	0.0317	98.70
0.1	100	0.0315	98.63
0.5	100	0.0312	98.68
1	100	0.0312	98.57
Fixed power transmission			
Output power	**PSR %**	**Avg. Cost per successful transmission mJ**	**Protocol Efficiency %**
-18 dBm	100	0.0306	99.27
-12 dBm	100	0.0326	99.94
-6 dBm	100	0.0390	100.00
0 dBm	100	0.0490	100.00

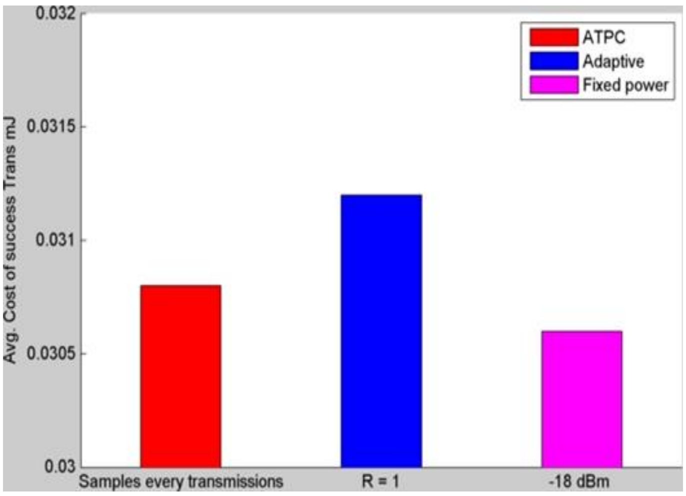


Fig. 5 University building- Comparison of the minimum cost due to different transmission strategy shows that there is hardly any difference in the average cost per successful transmission

Fig. 5 has compared the minimum average cost of successful transmission based on the three data sets of University building.

The optimal cost values of each transmission strategy are plotted. The average E_b/N_0 values due to transmission at lowest power level (-18 dBm) is high. That explains the facts from Fig. 5 that fixed power transmission at -18 dBm provides the most energy efficient solution. Any ramping-up in power level will always be wastage of power. The energy consumption values of the adaptive algorithm matches closely when the value of drop-off rate is 0.5. It signifies that the state-based system can perform most energy efficiently under very good link quality when it drops-off the fastest. Section 6.2 compares the energy cost due to ATPC, adaptive power control and fixed power transmission when the RSSI data are collected over a short and busy period of the day.

6.2 Short term busy hour RSSI data collected from University dining hall during busy hours

The variation of the E_b/N_0 and their normalized cumulative distribution are plotted in Fig. 6 and Fig. 7. The channel link quality is still good, with occasional drop to 20 dB due to multi-path fading effect. Since the over-all link quality is still very good (average $E_b/N_0 > 55$ dB), the PSR and the efficiency values in all these cases are 100 and higher than 98% respectively and their differences are negligible. Therefore, they are not plotted. The tabulated data of the performance parameters averaged over the three sets of observations are presented in table 8. Fig. 8 compares the minimum cost per successful transmission when University dining hall data are used. The difference in the cost is negligible. This is due to the very good quality of link quality (average E_b/N_0 >55 dB) for most of the time (> 93%).

Table 8 Average cost, PSR and protocol efficiency inside University dining hall

<table>
<tr><th colspan="4">ATPC- RSSI based adaptive power control</th></tr>
<tr><th>Channel scanning frequency</th><th>PSR %</th><th>Avg. Cost per successful transmission mJ</th><th>Protocol Efficiency %</th></tr>
<tr><td>Every time the sensor transmits</td><td>100</td><td>0.0319</td><td>98.65</td></tr>
<tr><td>Every 5 transmissions</td><td>100</td><td>0.0317</td><td>98.21</td></tr>
<tr><td>Every 10 transmissions</td><td>100</td><td>0.0317</td><td>98.14</td></tr>
<tr><td>Every 50 transmissions</td><td>100</td><td>0.0313</td><td>98.05</td></tr>
<tr><td>Every 100 transmissions</td><td>100</td><td>0.0314</td><td>97.86</td></tr>
<tr><th colspan="4">Non-RSSI based adaptive power control</th></tr>
<tr><th>Drop-off factor R</th><th>PSR %</th><th>Avg. Cost per successful transmission mJ</th><th>Protocol Efficiency %</th></tr>
<tr><td>0.01</td><td>100</td><td>0.0328</td><td>98.19</td></tr>
<tr><td>0.05</td><td>100</td><td>0.0319</td><td>98.21</td></tr>
<tr><td>0.1</td><td>100</td><td>0.0316</td><td>98.15</td></tr>
<tr><td>0.5</td><td>100</td><td>0.0315</td><td>97.98</td></tr>
<tr><td>1</td><td>100</td><td>0.0314</td><td>97.85</td></tr>
<tr><th colspan="4">Fixed power transmission</th></tr>
<tr><th>Output power</th><th>PSR %</th><th>Avg. Cost per successful transmission mJ</th><th>Protocol Efficiency %</th></tr>
<tr><td>-18 dBm</td><td>100</td><td>0.0310</td><td>97.74</td></tr>
<tr><td>-12 dBm</td><td>100</td><td>0.0328</td><td>99.17</td></tr>
<tr><td>-6 dBm</td><td>100</td><td>0.0391</td><td>99.59</td></tr>
<tr><td>0 dBm</td><td>100</td><td>0.0490</td><td>99.82</td></tr>
</table>

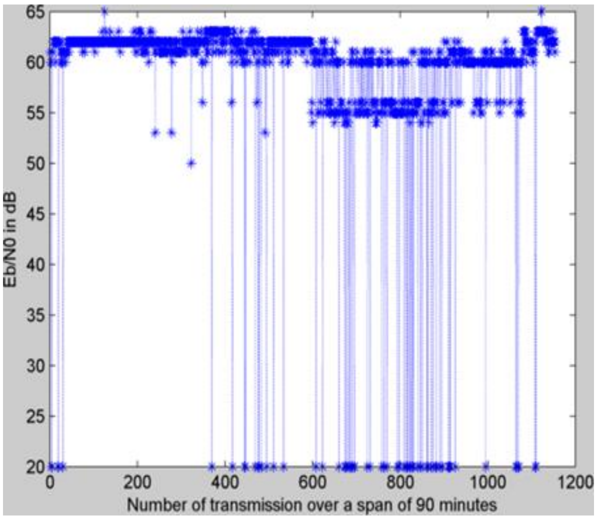


Fig. 6 The variation of the average E_b/N_0 from University dining hall during busy hour between 11:30 a.m. and 1:00 p.m. The average E_b/N_0 is quite high (>55 dB) and occasionally dropped to 20 dB. The busy hour period shows that the average E_b/N_0 can widely fluctuate between high E_b/N_0 (> 55 dB) and low E_b/N_0 (~20 dB)

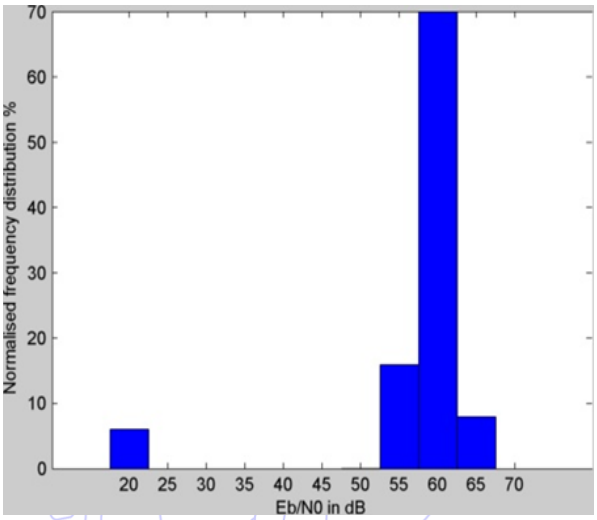


Fig. 7 The frequency distribution plot of the received E_b/N_0 from University dining hall during busy hour shows the rapid fluctuation in the signal level caused by movements of people in between the transmitting sensor and receiver

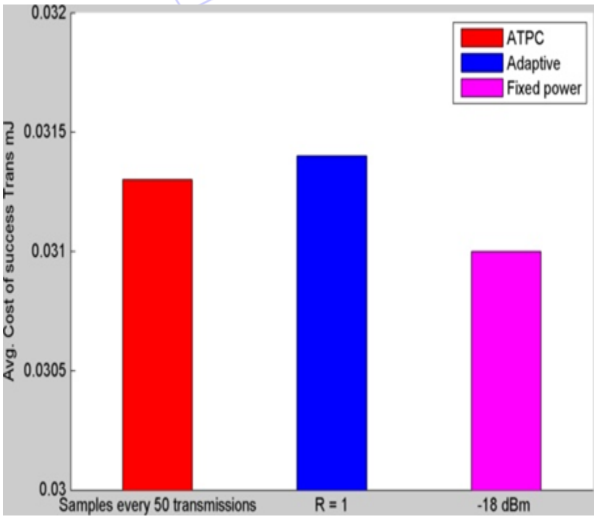


Fig. 8 University dining hall-Comparison of the minimum cost due to different transmission strategy shows that there is negligible difference in the cost per successful transmission

6.3 Short term RSSI data collected from town shopping center during busy hours of weekends

Three sets of short time data during busy hours of the town shopping center have been collected. The nature and the distribution of the variation of the E_b/N_0 values of the two data sets are almost similar. Therefore, the results of one of the sets are only presented. The variation of the E_b/N_0 is plotted in Fig. 9. It shows the significant variation of link quality during busy hours, primarily due to multipath fading effect as people move around in the shopping center.

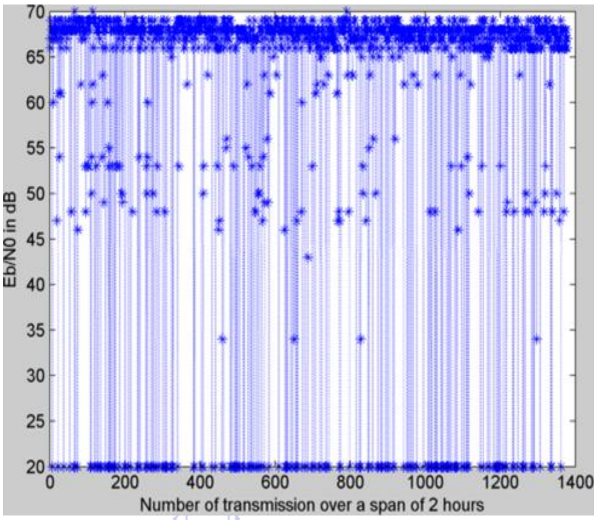


Fig. 9 It shows the variation of the average E_b/N_0 from town shopping center during busy hours between 11:30 a.m. and 1:30 p.m.

The distributions of the E_b/N_0 of the three sets of data are similar in nature. Fig. 1 shows the normalized distribution plot. It is expected that in a town shopping center during busy hours, the signal level will drop more frequently.

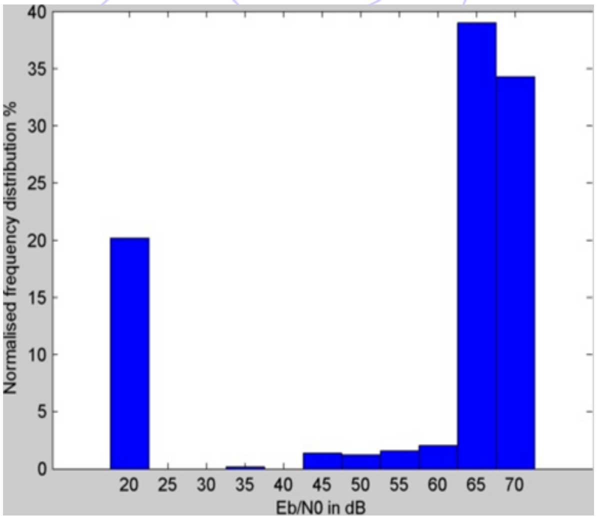


Fig. 10 The distribution plot of the received E_b/N_0 from town shopping centre during busy hours between 11:30 a.m. and 1:30 p.m. shows that E_b/N_0 at 20 dB is significantly high (~20%) which indicates that link quality has fluctuated frequently

Table 9 presents the average values the performance parameters based on the three sets of collected data.

Table 9 Average cost, PSR and protocol efficiency inside shopping centre

ATPC- RSSI based adaptive power control			
Channel scanning frequency	**Avg. Cost per successful transmission mJ**	**PSR %**	**Protocol Efficiency %**
Every time the sensor transmits	0.0315	100	98.44
Every 5 transmissions	0.0314	100	98.03
Every 10 transmissions	0.0314	100	98.02
Every 50 transmissions	0.0313	100	98.12
Every 100 transmissions	0.0311	100	97.88
Non-RSSI based adaptive power control			
Drop-off factor R	**Avg. Cost per successful transmission mJ**	**PSR %**	**Protocol Efficiency %**
0.01	0.0328	100	98.19
0.05	0.0319	100	98.11
0.1	0.0316	100	98.12
0.5	0.0300	100	97.90
1	0.0301	100	97.74
Fixed power transmission			
Output power	**Avg. Cost per successful transmission mJ**	**PSR %**	**Protocol Efficiency %**
-18 dBm	0.0310	100	97.91
-12 dBm	0.0327	100	99.24
-6 dBm	0.0391	100	99.67
0 dBm	0.0490	100	99.82

Fig. 11 compares the cost per successful transmission in different transmission strategies. There is no significant difference observed. The average E_b/N_0 is around 58 dB indicating a fairly good link quality.

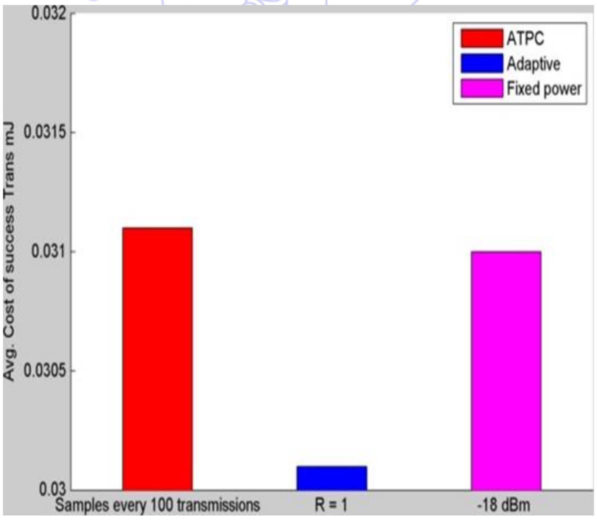


Fig. 11 Town shopping center- Comparison of the minimum cost due to different transmission strategy shows that there is no significant difference in the cost per successful transmission

It can be seen that when the average E_b/N_0 is high (> 55 dB), then even momentary fluctuations in the order of 35-40 dB caused by movements in between the transmitter and the receiver does not seriously affect the packet success rate, the cost of successful transmission and efficiency. In general, the adaptive power control algorithm can be energy efficient as compared to fixed power transmission when there is enough scope or space for manipulation. The results of these section show that when the link quality is good even at -18 dBm output power level, the adaptive algorithm has little scope to save energy. Transmission at the lowest power level provides the minimal solution. In section 7, the mean E_b/N_0 is decremented by 20 dB in order to compare the costs and efficiencies in different transmission strategies.

7. Comparison of PSRs, costs and efficiencies when mean E_b/N_0 is reduced by 20 dB

This section has studied the performance of the transmission strategies when the average or mean E_b/N_0 is reduced by 20 dB. It signifies the scenario when the distance between the sensor and the base station is further increases so that the net path loss is increased. With respect to each of the cases discussed in the previous section, the fluctuation of the E_b/N_0 is now roughly between 0 dB and 40 dB. The adaptive power control protocol (both RSSI and non-RSSI based) are now in a position to modulate the output power level when the signal level has dropped. Results in the next subsection shows that the optimal energy level in fixed power mode is no longer -18 dBm. This higher power level has also pushed the average cost of successful transmission high.

7.1 University building (set 1 and set 2) with average E_b/N_0 reduced by 20 dB

Table 10 shows the average of the performance parameters of the data that were collected. Fig. 12 compares the minimum or the optimal cost of each of the transmission strategies when University building data sets 1, 2 and 3 are used, along with their corresponding PSR and protocol efficiency values. It shows that the non-RSSI based adaptive protocol has proven to be more energy efficient than the fixed power transmission and ATPC. Fig. 12 suggests that the adaptive protocol can save approximately 7% energy as compared to ATPC and can consume 12% less energy than fixed power transmission. The PSR and the protocol efficiency of the non-RSSI based protocol are higher than that of the other two transmission strategies.

Table 10 Average cost, PSR and protocol efficiency inside University building

ATPC- RSSI based adaptive power control			
Channel scanning frequency	**PSR**	**Avg. Cost per successful transmission mJ**	**Protocol Efficiency %**
Every time the sensor transmits	95.68	0.0394	88.10
Every 5 transmissions	95.71	0.0400	88.27
Every 10 transmissions	95.87	0.0396	88.52
Every 50 transmissions	95.85	0.0402	88.51
Every 100 transmissions	94.96	0.0404	86.54
Non-RSSI based adaptive power control			
Drop-off factor R	**PSR**	**Avg. Cost per successful transmission mJ**	**Protocol Efficiency %**
0.01	99.70	0.0376	94.56
0.05	99.70	0.0370	92.28
0.1	99.70	0.0369	91.16
0.5	99.70	0.0371	88.65
1	99.70	0.0372	87.91
Fixed power transmission			
Output power	**PSR**	**Avg. Cost per successful transmission mJ**	**Protocol Efficiency %**
-18 dBm	93.52	0.0413	82.35
-12 dBm	94.69	0.0414	84.03
-6 dBm	99.41	0.0419	94.04
0 dBm	99.99	0.0498	98.67

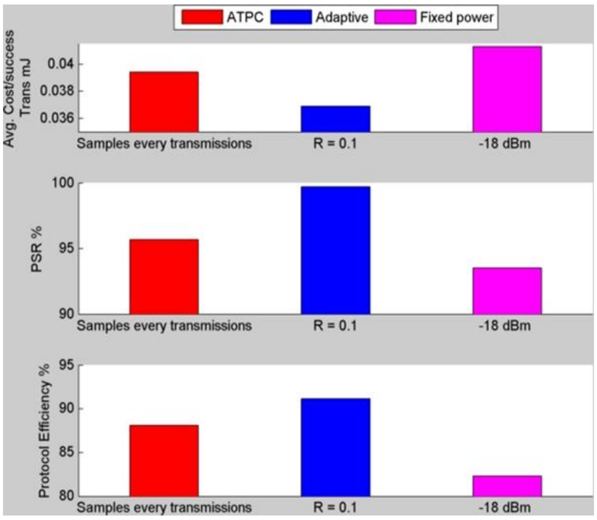


Fig. 12 University building - Comparison of the minimum cost and the corresponding PSR and protocol efficiencies due to different transmission strategy shows that the adaptive protocol can save 7% and 12% energy as compared to ATPC and fixed power transmission and outperforming the others in terms of PSR and efficiency

7.2 University dining hall with average E_b/N_0 reduced by 20 dB

Table 11 has tabulated all the performance parameter values of the data that were collected from University dining hall during busy hours.

Table 11 Average cost, PSR and protocol efficiency inside University dining hall during busy hour

ATPC- RSSI based adaptive power control			
Channel scanning frequency	**PSR%**	**Avg. Cost per successful transmission mJ**	**Protocol Efficiency %**
Every time the sensor transmits	92.44	0.0461	74.07
Every 5 transmissions	90.43	0.0514	70.45
Every 10 transmissions	89.88	0.0523	69.61
Every 50 transmissions	89.27	0.0534	68.38
Every 100 transmissions	86.62	0.0538	63.33
Non-RSSI based adaptive power control			
Drop-off factor R	**PSR%**	**Avg. Cost per successful transmission mJ**	**Protocol Efficiency %**
0.01	99.53	0.0448	92.20
0.05	99.37	0.0431	89.03
0.1	99.16	0.0431	87.22
0.5	98.98	0.0420	84.25
1	98.97	0.0422	83.04
Fixed power transmission			
Output power	**PSR%**	**Avg. Cost per successful transmission mJ**	**Protocol Efficiency %**
-18 dBm	83.84	0.0568	60.59
-12 dBm	86.98	0.0556	63.56
-6 dBm	98.46	0.0462	85.42
0 dBm	99.88	0.0508	96.34

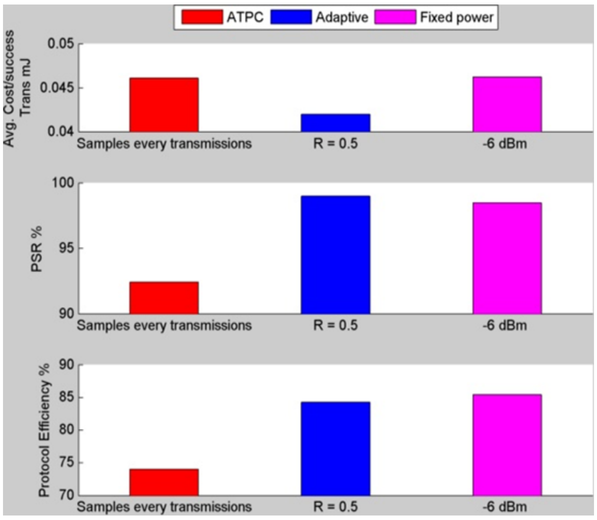


Fig. 13 University dining hall- Comparison of the minimum cost and their corresponding PSR and protocol efficiencies due to different transmission strategy shows that the adaptive protocol consumes 10% less energy than ATPC protocol and fixed power transmission, with comparable PSR and efficiency.

Results of the simulation are shown in Fig. 13. It suggests that the adaptive protocol has emerged out to be the best performer in terms of saving energy. It consumes approximately 10% less energy per successful transmission than ATPC and fixed power transmission.

7.3 *Town shopping center with average E_b/N_0 reduced by 20 dB*

Table 12 has tabulated all the performance parameter values of the data that were collected from University dining hall during busy hours.

Table 12 Average cost, PSR and protocol efficiency inside shopping centre during busy hour

ATPC- RSSI based adaptive power control			
Channel scanning frequency	**PSR %**	**Avg. Cost per successful transmission mJ**	**Protocol Efficiency %**
Every time the sensor transmits	90.78	0.0479	70.73
Every 5 transmissions	90.05	0.0498	68.11
Every 10 transmissions	89.37	0.0508	65.62
Every 50 transmissions	88.9	0.0504	65.93
Every 100 transmissions	88.92	0.0522	62.25
Non-RSSI based adaptive power control			
Drop-off factor R	**PSR %**	**Avg. Cost per successful transmission mJ**	**Protocol Efficiency %**
0.01	99.83	0.0438	92.20
0.05	99.83	0.0419	89.03
0.1	99.8	0.0417	87.22
0.5	99.85	0.0409	84.25
1	99.83	0.0410	83.04
Fixed power transmission			
Output power	**PSR %**	**Avg. Cost per successful transmission mJ**	**Protocol Efficiency %**
-18 dBm	83.84	0.0568	60.59
-12 dBm	86.98	0.0556	63.56
-6 dBm	98.46	0.0462	85.42
0 dBm	99.88	0.0508	96.34

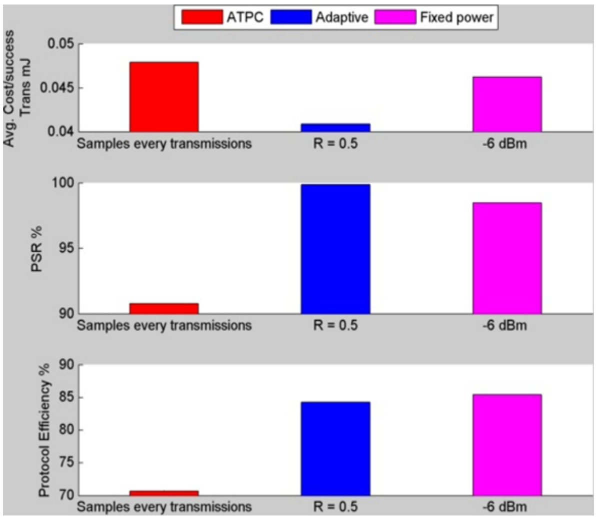


Fig. 14 Town shopping center- Comparison of the minimum cost and their corresponding PSR and protocol efficiencies due to different transmission strategy shows that the adaptive protocol consumes 17% less energy than the ATPC protocol, outperforming the others in terms of PSR and efficiency

Fig. 14 shows that the adaptive protocol has outperformed the ATPC protocol in terms of PSR, protocol efficiency and cost. Referring to Fig. 10, there was enough scope of output power modulation. The occupancy of the link quality around 20 dB (in terms of E_b/N_0) is high (more than 20%). The adaptive protocol makes full use of its adaptive power capability to save energy while maintain a high PSR and protocol efficiency. It is able to save more than 17% energy than ATPC and fixed power transmission.

8. Comparison of PSRs, costs and efficiencies when mean E_b/N_0 is further reduced by 20 dB

If the mean E_b/N_0 is further reduced by 20 dB, then the behavior of the channel is such that it oscillates between a good channel state (E_b/N_0 > 20 dB) and a bad channel state (E_b/N_0 ~ 0 dB). The sample normalized frequency distributions of the E_b/N_0 are shown in Fig.15.

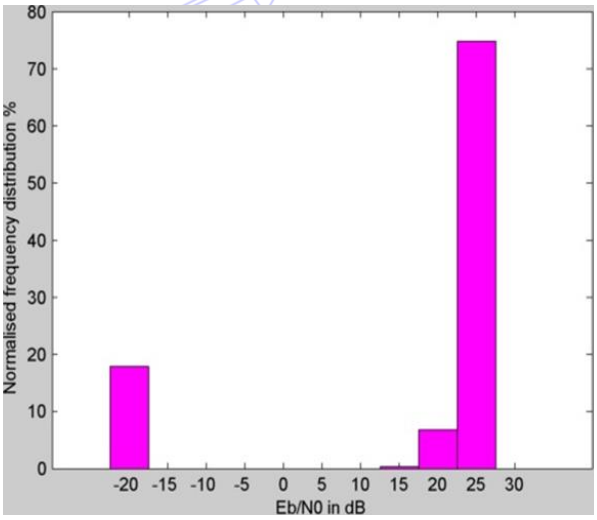


Fig. 15 University building- The normalized frequency distribution of the E_b/N_0 values suggests that channel is in bad condition for ~20% of time.

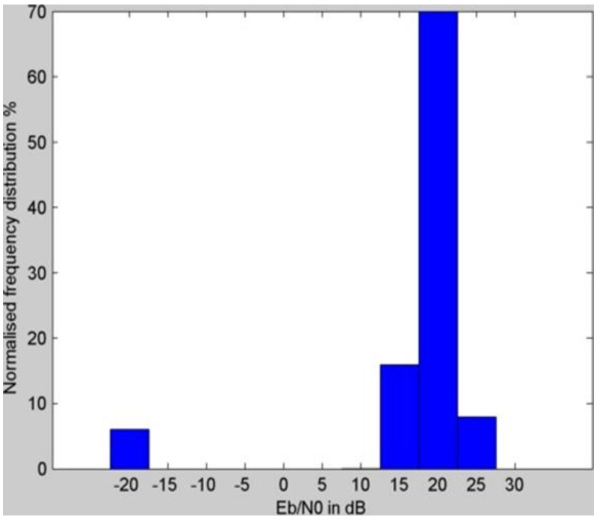


Fig. 16 University dining hall - The frequency distribution of the E_b/N_0 values in during busy hour shows that the channel quality has oscillated between good and bad. In good state most of the packets will be successfully transmitted, while in bad state almost no packet transmission will be successful

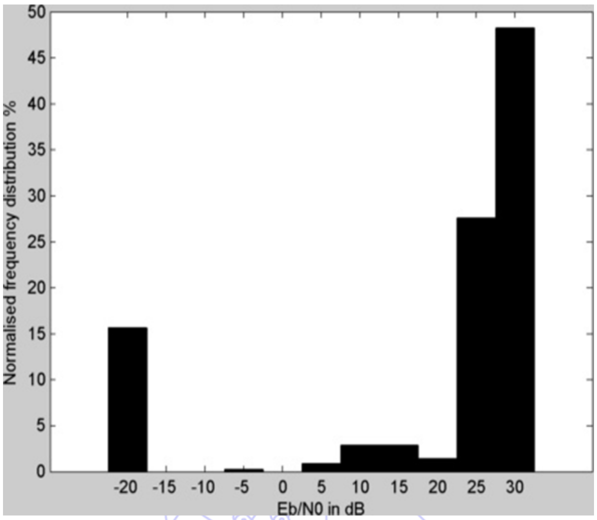


Fig. 17 Town shopping center- The frequency distribution of the E_b/N_0 values during busy hour shows that the channel quality has oscillated between good and bad. In good state most of the packets will be successfully transmitted, while in bad state almost

In such link conditions, the PSR will depend on the occupancy rate of the channel state at or below E_b/N_0 of -20 dB. This is because when the E_b/N_0 is -20 dB, power ramping up to 18 dB is not sufficient to make successful packet transmission possible. All the transmission strategies have approximately similar PSR and protocol efficiency. The fixed power transmission provides the optimal solution in terms of energy required per successful transmission in most of the scenarios that are discussed.

9. Conclusions

The results in this paper demonstrate the advantages and limitations of using power control under different channel conditions to achieve energy efficiency. When the link quality is good (mean E_b/N_0 > 55 dB for the lowest power level) with occasional drop by 20-30 dB due to fading, all the transmission strategies have comparable performances. This is because there was no scope of output power maneuvering to achieve energy efficiency. When the mean E_b/N_0 is dropped by 20 dB, the adaptive power control approach has proved to be energy-saving as compared to fixed power and ATPC when. Under this new channel condition, the mean E_b/N_0 is now approximately 30-35 dB and occasional signal drop to 0 dB has resulted in manipulation of power level. When the mean E_b/N_0 is further dropped by 20 dB, it represented a pure two state channel. The channel link quality oscillates between the good state and the bad state. In the good state, the average E_b/N_0 is around 15-20 dB and most of the packets were successfully transmitted. In the bad state, the average E_b/N_0 is -20 dB and all the packets were dropped. In this type of channel condition, the energy saving solution is provided by the fixed power transmission strategy because any ramping up of output power in bad state will not be sufficient to send a packet successfully. While in good state, it is not required to ramp up power for successful transmission. The non-RSSI approach can be a cost effective solution as compared to a RSSI based channel estimation method (ATPC) when the sensor and the hub are within the communicable distance.

References

[1] M. Weiser, "Ubiquitous computing", http://www.ubiq.com/weiser/UbiHome.html," March 17, 1996.

[2] Philips Research - Technologies, "What is ambient intelligence," http://www.research.philips.com/technologies/projects/ami/, 2015.

[3] J. P. Sheu, K. Y. Hsieh and Y. K. Cheng, "Distributed transmission power control algorithm for wireless sensor networks," Journal Of Information Science And Engineering, vol. 25, no. 5, pp. 1447-1463, 2009.

[4] S. Lin, J. Zhang, G. Zhou, T. H. Lin Gu and J. A. Stankovic, "ATPC: adaptive transmission power control for wireless sensor networks," in Proc. IEEE SenSys, Oct. 2006, pp. 223-236.

[5] N. Baccour, A. Koubaa, L. Mottola, M. A. Zuniga, H. Youssef, C. A. Boana and M. Alves, "Radio link quality estimation in wireless sensor networks: A Survey," ACM Transactions on Sensor Networks, vol. 8, no. 4, pp. 1-35, 2012.

[6] Chipcon products from Texas Instruments, "2.4 GHz IEEE 802.15.4 / ZigBee-ready RF Transceiver," http://www.ti.com/lit/ds/symlink/cc2420.pdf, Oct. 2014.

[7] Y. Fu, M. Sha, G. Hackmann and C. Lu, "Practical control of transmission power for wireless sensor networks," Proc. 20th IEEE International Conference on Network Protocols (ICNP '12), IEEE press, Oct. 2012, pp. 1-10.

[8] G. Hackmann, O. Chipara and C. Lu, "Robust topology control for indoor wireless sensor networks," Proc. ACM conference on Embedded network sensor systems, 2008, pp. 57-70.

[9] S. Soltani, M. U. Ilyas and H. Radha, "An energy efficient link layer protocol for power-constrained wireless networks," Proc. International Conference on Computer Communications and Networks (ICCCN '11), July. 2011, pp. 1-6.

[10] L. Zheng, W. Wang, A. Mathewson, B. O'Flynn and M. Hayes, "An adaptive transmission power control method for wireless sensor networks," in Proc. ISSC, June. 2010, pp. 261-265.

[11] J. Jeong, D. Culler and J. H. Oh, "Empirical analysis of transmission power control algorithms for wireless sensor networks," International Conference on Networked Sensing Systems, 2007, pp. 27-34.

[12] D. Lal, A. Manjeshwar, F. Herrmann, E. Uysal-Biyikoglu and A. Keshavarzian, "Measurement and characterization of link quality metrics in energy constrained wireless sensor networks," Proc. IEEE Global Telecommunications Conference (GLOBECOM '03), IEEE press, Dec. 2003, vol. 1, pp. 446-452.

[13] A. Sheth and R. Han, "An implementation of transmit power control in 802.11b Wireless Networks," University of Colorado at Boulder, 2002.

[14] "nRF24L01+ Single Chip 2.4GHz Transceiver Product Specification v1.0," http://www.nordicsemi.com/eng/Products/2.4GHz-RF/nRF24L01P.

[15] D. G. Zhang, Y. N. Zhu, C. P. Zhao, and W. B. Dai, "A new constructing approach for a weighted topology of wireless sensor networks based on local-world theory for the Internet of Things (IOT) ," Computers and Mathematics with Applications, Elsevier, 2012, pp. 1044-1055.

[16] D. G. Zhang and Y. P. Liang, "A kind of novel method of service-aware computing for uncertain mobile applications," Mathematical and Computer Modelling, Elsevier, Jun. 2012, pp. 344-356.

[17] D. G. Zhang, G. Li, K. Zheng, X. C. Min, and Z. H. Pan, "An energy-balanced routing method based on forward-aware factor for wireless sensor networks," IEEE Transactions On Industrial Informatics, vol. 10, no. 1, Feb. 2014, pp. 766-773.

[18] D. G. Zhang and X. D. Zhang, "Design and implementation of embedded uninterruptible power supply system (EUPSS) for web-based mobile application," Enterprise Information Systems, vol. 6, no. 4, pp. 473-489, DOI: 10.1080/17517575.2011.626872.

[19] D. G. Zhang, K. Zheng, T. Zhang, and X. Wang, "A novel multicast routing method with minimum transmission for WSN of cloud computing service," Soft Comput, Springer-Verlag Berlin Heidelberg, pp. 1817–1827, 2014.

[20] D. G. Zhang, "A new approach and system for attentive mobile learning based on seamless migration," Applied Intelligence, vol. 36, no. 1, pp. 75-89, 2012.

[21] D. G. Zhang, X. Wang, and X. D. Song, “A novel approach to mapped correlation of ID for RFID anti-collision”, IEEE Transactions on Services Computing,” vol. 7, no. 4, pp. 741-748, 2014.
[22] D. G. Zhang, X. D. Song, and X. Wang, “New agent-based proactive migration method and system for big data environment (BDE),” Engineering Computations, vol. 32, no. 8, pp. 2443- 2466, 2015.
[23] D. Basu, G. Sen Gupta, G. Moretti, and X. Gui, "Protocol for improved energy efficiency in wireless sensor networks to support mobile robots," Proc. International Conference on Automation, Robotics and Applications (ICARA), Feb. 2015, pp. 230-237.
[24] D. Basu, G. Sen Gupta, G. Moretti, and X. Gui, “Performance comparison of a novel adaptive protocol with the fixed power transmission in wireless sensor networks,” Journal of Sensor and Actuator Networks, Multidisciplinary Digital Publishing Institute (MDPI AG), pp. 274-292, Sep. 2015.
[25] A. S. Tanenbaum and Computer Networks, Chapter 4, 4th ed., Prentice Hall PTR, 2002.
[26] IEEE Standard for Information technology, IEEE Computer Society, 2002.
[27] "Universal Mobile Telecommunications System (UMTS); Selection procedures for the choice of radio transmission technologies of the UMTS (UMTS 30.03 version 3.1.0)," UMTS, 2011.
[28] K. Srinivasan and P. Levis, "RSSI is under appreciated," in Workshop on Embedded Networked Sensors (EmNets), 2006.
[29] D. Schmidt, M. Berning, and N. Wehn, "Error correction in single-hop wireless sensor networks - A case study," Conference & Exhibition of Design, Automation & Test in Europe, 2009, pp. 1530-1591.
[30] "NetSurveyor — 802.11 Network Discovery / WiFi Scanner," http://nutsaboutnets.com/netsurveyor-wifi-scanner/.
[31] "Beacon Frame," http://en.wikipedia.org/wiki/Beacon_frame.
[32] J. Geier, "802.11 Beacons Revealed," http://www.wi-fiplanet.com/tutorials/print.php/1492071, Oct. 2002.

Appendix A. **Some experimental scenarios and radio propagation environments**

A.1 Within University building

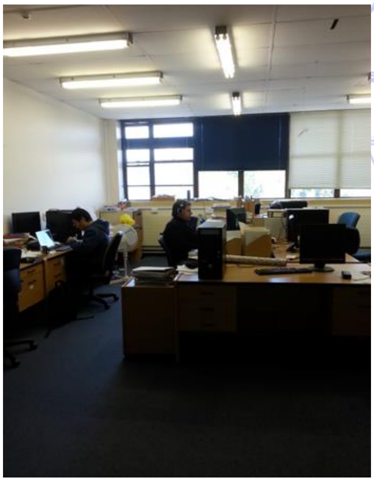


Fig A.1.1 Sample radio environment

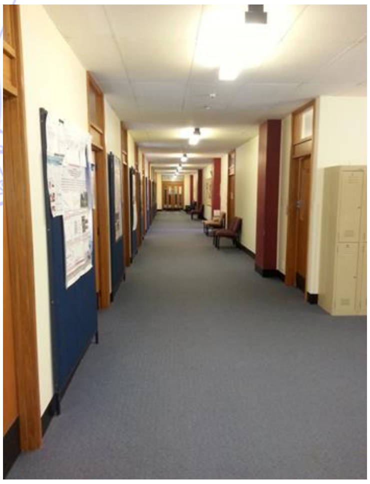


Fig A.1.2 Sample radio environment

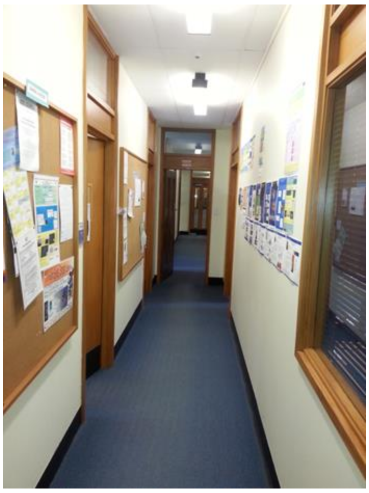


Fig A.1.3 Sample radio environment

A.2 Shopping Mall food court during busy hours

Fig A.2.1 Sample radio environment of Shopping Mall food court

A.3 University dining hall during busy hours

Fig A.3.1 Sample radio environment of University dining hall during busy hours

International Journal of Engineering and Technology Innovation, vol. 6, no. 1, 2016, pp. 55 - 67

Power Consumption Reduction for Wireless Sensor Networks Using A Fuzzy Approach

Giovanni Pau*

Faculty of Engineering and Architecture, Kore University of Enna, Italy.

Received 09 November 2015; received in revised form 08 December 2015; accepted 15 December 2015

Abstract

The increasing complexity of Wireless Sensor Networks (WSNs) is leading towards the deployment of complex networked systems and the optimal design of WSNs can be a very difficult task because several constraints and requirements must be considered, among all the power consumption. This paper proposes a novel fuzzy logic based mechanism that according to the battery level and to the ratio of Throughput to Workload determines the sleeping time of sensor devices in a Wireless Sensor Network for environmental monitoring based on the IEEE 802.15.4 protocol. The main aim here is to find an effective solution that achieves the target while avoiding complex and computationally expensive solutions, which would not be appropriate for the problem at hand and would impair the practical applicability of the approach in real scenarios. The results of several real test-bed scenarios show that the proposed system outperforms other solutions, significantly reducing the whole power consumption while maintaining good performance in terms of the ratio of throughput to workload. An implementation on off-the-shelf devices proves that the proposed controller does not require powerful hardware and can be easily implemented on a low-cost device, thus paving the way for extensive usage in practice.

Keywords: wireless sensor networks, fuzzy logic controller, power consumption, IEEE 802.15.4

1. Introduction

Technology advances in chip miniaturization, energy consumption and wireless communication have enabled the development and the deployment of new applications based on Wireless Sensor Networks (WSNs) [1]. A WSN is an ad-hoc network composed of tiny devices with limited energy and computational resources, and it equipped with sensors in order to gather physical measures from the monitored environment. A lot of research effort has been spent on WSNs and many architectures [2,3] and protocols [4,5] have been developed. Typical civil WSNs are basically not complex monitoring systems, whose applications encompass environment and habitat monitoring [6,7], home automation [8,9], industrial sensing [10,11] and intelligent transportation systems [12,13]. In these WSNs, sensors gather the required information, mostly, according to a fixed temporal schedule, and send it to the sink, which interfaces with a server or a computer. Only at this point data from sensors can be processed, before being stored.

A WSN is composed by several nodes that communicate among each other through a wireless channel. These nodes are typically battery-powered, and equipped with low-performance processors and small memories in order to reduce the power requirements. It is useful to note that a common WSN node comprises five main components [14]: a processing unit (microcontroller, processor, FPGA, ...), memories (DRAM, SRAM, Flash, ...), sensors and actuators, multiple communication

* Corresponding author. E-mail address: giovanni.pau@unikore.it

Tel.: +39-0935-536494; Fax: +39-0935-536623

layers (physical radio, Medium Access Control, Routing, ...) and a power supply (external power supply, batteries, solar cells, ...). During the design phase, the cooperation of all these components must be combined in order to identify the configuration that best fits the design objectives.

Recent advances in low power transceivers and microprocessor dimensions have led to cost effective tiny sensor devices that combine sensing with computation, storage and communication. However, one of the major issues in WSNs is the power consumption, due to the fact that sensors are mainly battery powered. For example, a battery-operated sensor device that wakes up once every few minutes to check an environmental parameter needs to consume as little power as possible in order to minimize the battery replacement. In several cases, nodes are deployed in harsh environments, such as underground or underwater, where replacing battery could be an unfeasible operation. Extending the network lifetime is a crucial concern. In fact, there is a main requirement that make a wireless protocol ideal for use in the WSNs, that is the energy efficiency.

A great deal of effort has been made by researchers to find effective strategies in order to increase network lifetime. These strategies encompass network node deployment, routing mechanisms and data aggregation. In fact, an appropriate node deployment is probably the most critical issue to be addressed in order to reduce communication costs within a WSN [15]. Subsequently, once the network is deployed, the use of appropriate routing mechanisms could help to considerably increase its lifetime because a convenient choice of paths to route data may result in significant energy conservation. Therefore, network lifetime optimization can be achieved not only with the reduction of packet transmission power, but also with the involvement of data processing in order to reduce the amount of data delivered to sinks; this is the principle behind node clustering protocols [16].

The energy consumption in WSNs is determined by three main components that are sensing, processing and transmission. Sensing energy consumption for sensor node is determined by the specific characteristics of the sensor, and its value is determined based on the device datasheet. On the contrary, processing energy consumption $E_{i,k}^{proc}$ for sensor node i and task k is proportional to the complexity of task k (the number of instructions I_k needed to complete it) and to the average energy consumption per instruction E_i^{ins} related to node i. This relationship can be expressed as follows:

$$E_{i,k}^{proc} = I_k \times E_i^{ins} \tag{1}$$

Furthermore, as shown in [17], there are two main components that must be taken into account, which are the transmission and reception energy consumption:

$$\begin{cases} P_{ij}^{T} = P_i^{T0} + P_i^{A}\left(\delta_{ij}\right) = P_i^{T0} + \dfrac{P_i^{Tx}(\delta_{ij})}{\eta_i} \\ P_j^{R} = P_j^{R0} \end{cases} \tag{2}$$

where P_{ij}^{T} and P_j^{R} are radio frequency power consumption values for transmitting and receiving respectively; $P_i^{A}(\delta_{ij})$ is the power consumption of the Power Amplifier (PA), which depends on the distance δ_{ij} between transmitting node i and receiving node j; P_i^{T0} and P_j^{R0} are the components of power consumption of the transmitting and receiving circuitry respectively; P_i^{Tx} is the output power at node i antenna which, for reliable transmissions depends on the distance δ_{ij}; η_i is the drain efficiency of the PA at node i. Considering a channel in which the path loss component is predominant, the transmitted power $P_i^{Tx}(\delta_{ij})$ can be expressed as follows:

$$P_{ij}^{Tx}\left(\delta_{ij}\right) = P_j^{Rx} \times A_{ij} \times \delta_{ij}^{\alpha PL} \tag{3}$$

where A_{ij} is a parameter determined by the characteristics of the antennas (such as gain and efficiency) and αPL denotes the path-loss exponent, which is about 2 for free space. This kind of modelling is typical of free space propagation but it might be extended taking into account other fading effects. Considering the equations 2 and 3, the following equation can be derived:

$$P_{ij}^{T} = P_i^{T0} + \frac{P_j^{Rx} \times A_{ij} \times \delta_{ij}^{\alpha PL}}{\eta_i} \tag{4}$$

Considering $\varphi_{ij} = P_j^{Rx\,min} \times A_{ij}$, where $P_j^{Rx\,min}$ is the minimum reception power at node j for a reliable communication, the equation 4 can be written as follows:

$$P_{ij}^{T} = P_i^{T0} + \frac{\varphi_{ij} \times \delta_{ij}^{\alpha PL}}{\eta_i} \tag{5}$$

Defining E_{ij}^{Tx} the energy per bit necessary to transmit data at rate R from node i to its adjacent node j, and E_j^{Rx} the per-bit energy consumed to receive data at node j, then they can be written as follows:

$$\begin{cases} E_{ij}^{Tx} = \frac{P_{ij}^{T}}{R} = \frac{1}{R}\left(P_i^{T0} + \frac{\varphi_{ij} \times \delta_{ij}^{\alpha PL}}{\eta_i}\right) \\ E_j^{Rx} = \frac{P_j^{R}}{R} = \frac{P_j^{R0}}{R} \end{cases} \tag{6}$$

Although several approaches have already been proposed in the literature in order to improve the energy management in a WSN, the use of innovative approaches based on soft computing techniques may be a viable solution in order to obtain an optimal power consumption management in a WSN.

The soft computing techniques fit themselves well in WSN applications, since they have been proposed for the construction of new generation artificial intelligence (high machine intelligence quotient, human-like information processing) and for solving non-linear and mathematically un-modeled systems. The use of rule-based Fuzzy Logic Controllers (FLCs) [18] enables the implementation of multi-criteria control strategies. Fuzzy logic is widely adopted in WSNs because it can deal with uncertain and vague values, such as the interference between two nodes or their estimated distance. In these cases, an accurate computation may be too complex and it could also be meaningless due to the quick change of the network conditions. On the contrary, the use of smart setting and tuning techniques for FLCs can improve the energy savings in a WSN. For this reason, the FLCs, based on linguistic rules instead of inflexible reasoning, can be the right choice to describe a mechanism for energy saving in order to prolong the lifetime of network nodes.

To cope with the power consumption problem in WSNs, this paper proposes a novel fuzzy logic based approach in an environmental monitoring context. The proposed wireless network architecture is based on the IEEE 802.15.4 protocol [19] and is organized in Environmental Monitoring Cells (EMCs). The energy saving of the network is obtained though a fuzzy module. In fact, the goal is to improve the low energy consumption of IEEE 802.15.4 through a fuzzy logic controller. The IEEE 802.15.4 protocol has been also chosen because it has a main advantage in its range since many IEEE 802.15.4 based technologies (e.g. ZigBee) support mesh whereby coverage can be extended by using routers. It is useful to note that this paper introduces a novel approach (not an application) to manage and to optimize the power consumption in WSNs.

The paper is organized as follows. In Section 2, the main related works in order to deduce the innovations introduced with this work are shown. In Section 3, the system architecture is described, while the fuzzy based approach and the considered membership functions are introduced in Section 4. In Section 5, the performance obtained by the proposed approach are shown and, finally, in Section 6, the paper is summarized, reporting conclusions and future works.

2. Related Works

Energy saving is one of the fundamental issue that characterize the WSNs because most wireless devices are usually battery-powered. Therefore, it is essential to manage these devices to best utilize the scarce power resources over long time. In the literature adequate and complete research works that describes the application of a fuzzy logic controller in order to reduce the power consumption over WSNs are missing. In fact, different approaches, methodologies and technologies have been proposed in the literature in order to reduce the energy consumption of wireless sensor networks based on the IEEE 802.15.4 protocol. They achieve their goal but often their implementation is particularly difficult and requires modification of the stack protocol. On the contrary, in this paper, an approach to improve the energy consumption of a WSN nodes based on fuzzy logic is introduced, trying also to fill the gaps and difficulties in implementation of other approaches in the literature

The authors of [20] propose a better conditioned energy-saving frame format of an enhanced distributed queuing medium access protocol for body sensor networks based on IEEE 802.15.4 protocol in healthcare scenarios. In fact, the main aim is to overcome the limitations of the IEEE 802.15.4 MAC in order to obtain an improved energy efficiency. The approach proposed by the authors is evaluated analytically and the results show that it outperforms the IEEE 802.15.4 MAC in terms of overall energy-consumption per information bit. Whereby, the authors highlight that their approach represents a remarkable improvement of the overall network energy efficiency, which scales well for very dense body sensor networks and it is particularly suitable in medical scenarios.

A global study on energy considerations in the context of a wireless network based on the IEEE 802.15.4 technology is carried out in [21]. The authors want to emphasize that the lifetime announced by hardware manufacturer is clearly not suited when devices are used for networking. For this reason, they first show the impact of variable loads on battery chemistry reducing the node lifetime. In fact, these effects are non linear and are hardly predictable. Even at low duty cycle, the high currents needed for transmit/receive operations drastically reduce the node lifetime. Subsequently, the authors focus on the power consumption during the receive mode because it represents the worst case in terms of power consumption, at the end, they demonstrate the effectiveness of low power modes for increasing node lifetimes, obtained sacrificing the processing power in applications.

In [22], an approach for load balancing to fairly distribute energy consumption among nodes in IEEE 802.15.4-based WSNs is introduced. In fact, the proposed approach aims to conserve energy of each node in order to extend the network lifetime. In order to achieve this goal, the authors propose a new dynamic association/re-association approach allowing path alternation relative to association criteria and their threshold parameters. Simulations results show the efficiency of the approach proposed by the authors in term of both energy distribution, which leads to a longer lifetime, and lower latency, for real-time communication.

In [23], a reliable energy-efficient multi-level routing algorithm, based on a fuzzy inference system, for WSNs is proposed. The fuzzy-based approach proposed by the authors considers several network parameters, such as the residual energy, the number of neighbors' nodes and the centrality for cluster formation, which are critical for well-balanced energy dissipation of the network. The main aim of the fuzzy reasoning algorithm is to choose the cluster heads and to construct a multi-hop routing among cluster heads in an energy efficiently way. Simulation results demonstrate that the approach introduced by the authors obtains significant energy savings and, at the same time, prolongs network lifetime when compared to other approaches in the literature.

A fuzzy logic system is presented in [24] in order to save the battery life of WSN nodes and to have an efficient, robust and cost effective sensing network that can monitor events of interest, for example those of a home automation environment. The proposed fuzzy logic system helps efficiently to decide the situation of on/off state for active/sleep mode of the processing and communication parts of the sensor node. Simulation results show that the proposed system is energy efficient and have property of liveness, soundness, without any deadlock state during execution.

The authors of [25] present transmission power control algorithms, based on soft computing techniques, in order to reduce the energy consumption in wireless sensor network, without affecting its throughput. Two algorithms are proposed, one using artificial neural network and the other using fuzzy logic control. The experimental results show that the proposed algorithms obtain a marked improvement in performance when compared to the conventional Medium Access Control protocol of IEEE 802.15.4 standard.

The analysis of these research works clearly has shown that fuzzy logic can be applied in order to reduce the power consumption in several contexts. Therefore, although IEEE 802.15.4 has a low power consumption, it is clear that a fuzzy mechanism could be introduced in order to further improve the power consumption in environmental monitoring applications.

3. The Proposed Network Architecture

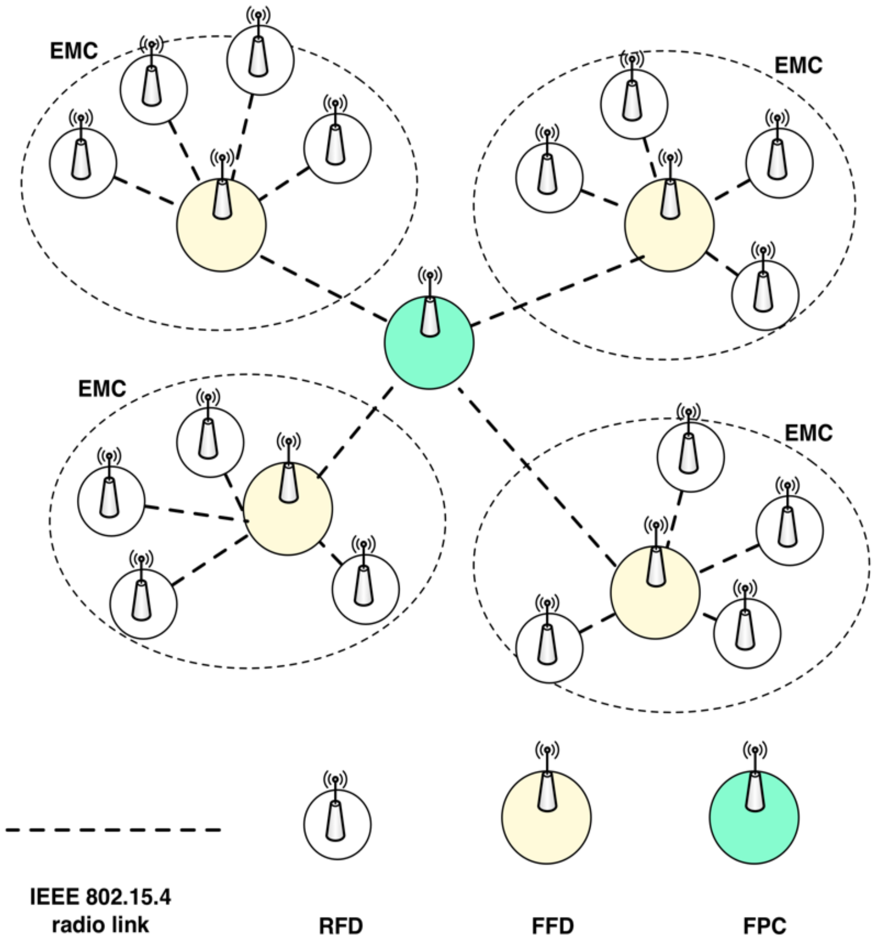


Fig. 1 Environmental monitoring IEEE 802.15.4 architecture

The main aim of this paper is not to satisfy all the requirements of a WSN, but rather to improve the power consumption. In Fig. 1, the proposed network architecture is depicted and is composed by several Environmental Monitoring Cells (EMCs) based on the IEEE 802.15.4 protocol [19]. This standard defines the physical layer (PHY) and data-link specifications (MAC) in order to ensure low data-rate wireless communications among devices requiring low power. The wireless network for environmental monitoring proposed in this work is composed by all the EMCs within which there are different devices dealing with a specific task. Reduced Function Device (RFD) nodes measure a physical parameter of the monitored environment, such as temperature, humidity, light, smoke density, carbon monoxide, etc.. Then, they send acquired data to their Full Function Device (FFD) nodes. An FFD node could be a ZigBee router [26]. It forwards data received from RFD nodes to the First Pan Coordinator (FPC) that processes information and send appropriate command messages to the sensor nodes.

It is useful to note that in IEEE 802.15.4 the transmission range varies considerably depending on the nature of the path that must be line of sight (LOS) for the most part. Transmit power level and receiver sensitivity are also factors to take into account. Under the best conditions, the range can be as great as 1000 meters with a clear outdoor path. Most applications cover a shorter range of 10 to 75 meters. Anyhow, it is necessary to take into account the energy consumption of the devices. For this reason, the Energy Saving Fuzzy Controller proposed in this work, described in Section 4, is necessary in order to ensure the power consumption management. This module dynamically manages sampling times in order to prolong sleeping periods of sensor nodes. In this way, it is possible to improve the energy savings and, at the same time, prolong batteries and the network life-cycle.

4. Energy Saving Fuzzy Controller

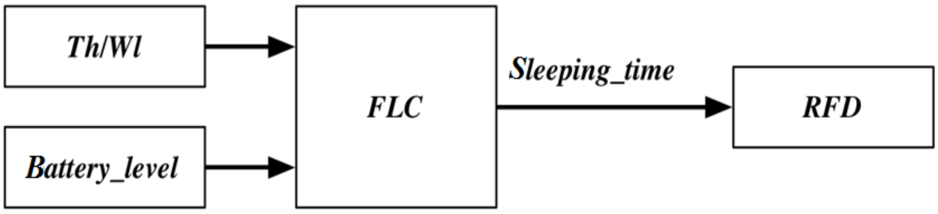
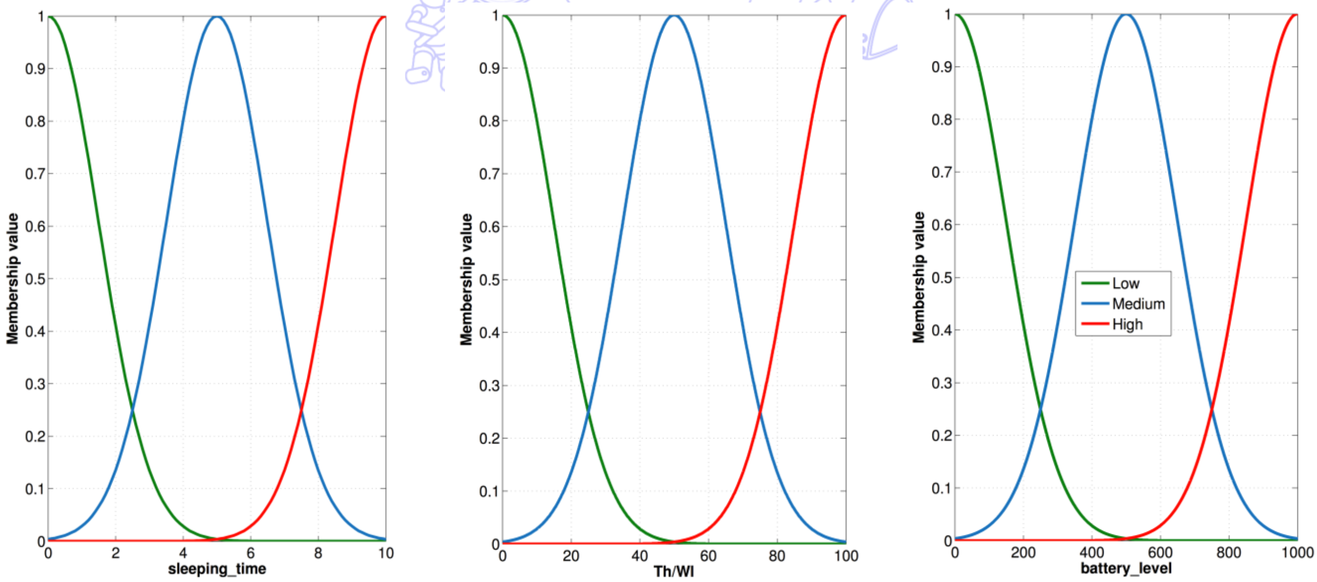


Fig. 2 Proposed FLC Architecture

Considering the proposed WSN, it is not possible to determine "a priori" the behavior of nodes since they are often used to monitor sporadic events. However, traffic flows generated by WSNs can be considered as periodic [27]. In this paper, a centralized mechanism is proposed in order to regulate the sleeping time of the field devices in a IEEE 802.15.4 network, with the aim of reducing their power consumption. Each RFD sends information about its operating conditions to the FPC when its sleeping time is expired. The FPC is a special device properly equipped for executing computational tasks. Anyhow, it is useful to note that generally in IEEE 802.15.4-based network the FPC nodes are not always battery powered. As shown in Fig. 2, in the proposed approach, the FPC node uses a Fuzzy Logic Controller (FLC) in order to calculate the new values of the *sleeping_time* of each RFD. The FLC determines the *sleeping_time* of the RFD according to the *battery_level* and to the ratio of Throughput to Workload (*Th/Wl*). The throughput is the sum of both periodic and aperiodic packets sent by the device. The workload is the total number of packets that the device has to send.



(a) sleeping_time (b) Th/Wl (c) battery_level

Fig. 3 Gaussian membership functions

The proposed FLC considers three membership functions (*Low*, *Medium*, *High*) for each input variable. These functions fuzzify the crisp inputs, while the ranges of which are:

- *Th/Wl percentage*: [0-100] (percentage);
- *battery_level*: [0-1024], where 0 represents the lowest level of battery while 1024 represents the highest level of battery

and it is the maximum value at the output of a 10-bit AD converter (with an appropriate electronic signal conditioning circuit).

In the same way, three membership functions (*Low*, *Medium*, *High*) are defined for the *sleeping_time*. In this case, the range of the crisp values of this output variable is:

- sleeping_time: [0-10] ∗ sampling_time (seconds).

where the *sampling_time* value is a constant value defined at design time for each field device. Gaussian fuzzy membership functions of the *Th/Wl*, the *battery_level* and the *sleeping_time* are depicted in Fig. 3 respectively, where the degree of membership is represented by normalized values [0-1]. Considering that the Gaussian function [28] is defined by a central value m and a standard deviation k > 0, the smaller k is, the narrower the "bell" is, it can be represented as follows:

$$\mu_A(x) = e^{\frac{-(x-m)^2}{2k^2}} \tag{7}$$

Table 1 Values of variables

Universe of discourse	Set	k	m
Th/Wl	Low	15	0
	Medium	15	50
	High	15	100
battery_level	Low	150	0
	Medium	150	500
	High	150	1000
sleeping_time	Low	1.5	0
	Medium	1.5	5
	High	1.5	10

The different values of the variables are shown in Table 1. As shown in Table 2, the output value is determined through 9 fuzzy rules based on the IF-THEN statement of classic programming languages. For instance, considering rule 7, if *Th/Wl* is *High* and *battery_level* is *Low*, *sleeping_time* will be *High*. The outputs of the inference mechanism are fuzzy output variables. The FLC must convert its internal fuzzy output variables into crisp values, through the defuzzification process, so that the actual system can use these variables. Defuzzification can be performed in several ways. In this paper, the Centroid of Area (COA) method [29] has been chosen. In this method, the centroid of each membership function for each rule is first evaluated. The final output is then calculated as the average of the individual centroid weighted by their membership values as follows:

$$sleeping\_time = \frac{\sum_{i=1}^{n} Out_i * C_i}{\sum_{i=1}^{n} C_i} \tag{8}$$

Table 2 Inference rules

Rule	Antecedent (Th(Wl)	Antecedent (battery_level)	Consequent (sleeping_time)
1	Low	Low	Medium
2	Low	Medium	Low
3	Low	High	Low
4	Medium	Low	Medium
5	Medium	High	Low
6	Medium	Medium	Medium
7	High	Low	High
8	High	Medium	High
9	High	High	High

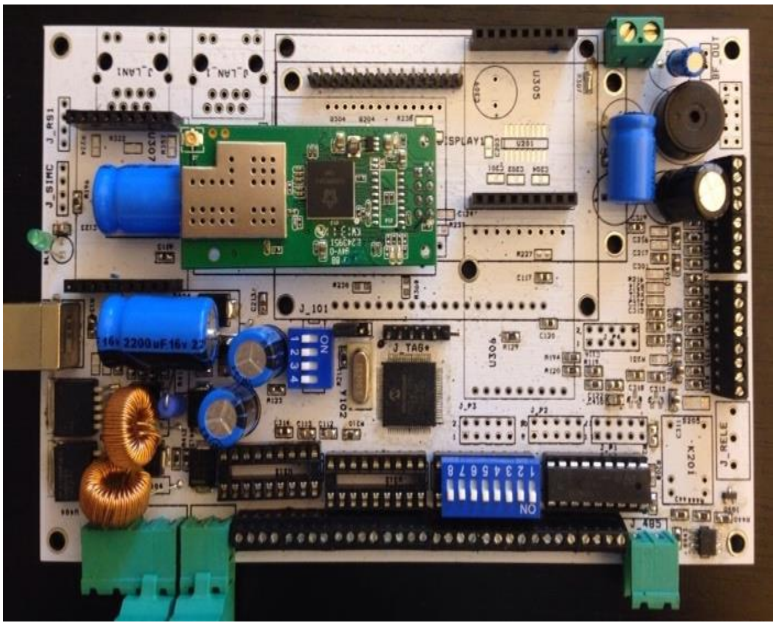


Fig. 4 Used prototyping board

5. Performance Evaluation

Using the proposed fuzzy-based approach, presented in the previous Section, several simulations have been performed. However, before showing the obtained results, it is necessary to describe the used simulation model. A generic star topology and a clustered one have been chosen for the performance evaluation. In a star topology, all the different nodes are required to talk only to the central PAN coordinator (FPC). Even if the nodes are FFDs and are within range of each other, in a star network topology, they are only allowed to communicate with the coordinator node. This does not occur in the other considered network topology. Both the FPC, FFD and RFD nodes have been implemented in a prototyping board, shown in Fig. 4, equipped with a micro-controller and a wireless module IEEE 802.15.4 compliant:

- 16 bit MCU - Microchip PIC24F family (PIC24FJ256GB108) [30];
- MRF24J40MB Radio Frequency Transceiver IEEE 802.15.4 2.4 GH [31].

The simulations have been conducted with a model built in Simulink/Matlab depicted in Fig. 5. The main aim of this model is to simulate the behavior of the coordinator node and of a sensor node, in the same way as the approach proposed in [32]. Specifically, the Sensor Node block manages the battery consumption of the RFD node. The *sleeping_time* and the transmission power (TXPower) are acquired as input parameters of this block through a feedback loop system. This block produces two output variables (*Th/Wl* and *battery_level*) that are used as input variables by the FLC.

It is useful to highlight that the activities of the micro-controller and of a connected sensor have a low impact on the battery consumption. In fact, their energy requirement is estimated as 50mA (MC_{PC}). This value has been measured in an electronic board characterized by the PIC24FJ256GB108 micro-controller and a DS18B20 [33] temperature sensor. Whereby, the battery consumption mainly depends on the working state of the device. When the device is sleeping, the battery consumption is 50mA + 5μA (5μA is the power consumption of the IEEE 802.15.4 module in sleeping mode, obtained from the datasheet [31]). On the contrary, when the device is transmitting, the transmission power heavily affects the energy consumption. The battery consumption trend is evaluated in relation to the sleeping time by means of the Simulink/Stateflow environment. This tool uses flow charts and finite state machines in order to represent the evolution of a system. The Chart section, created in Simulink/StateFlow, related to the behavior of the battery consumption is represented in Fig. 6.

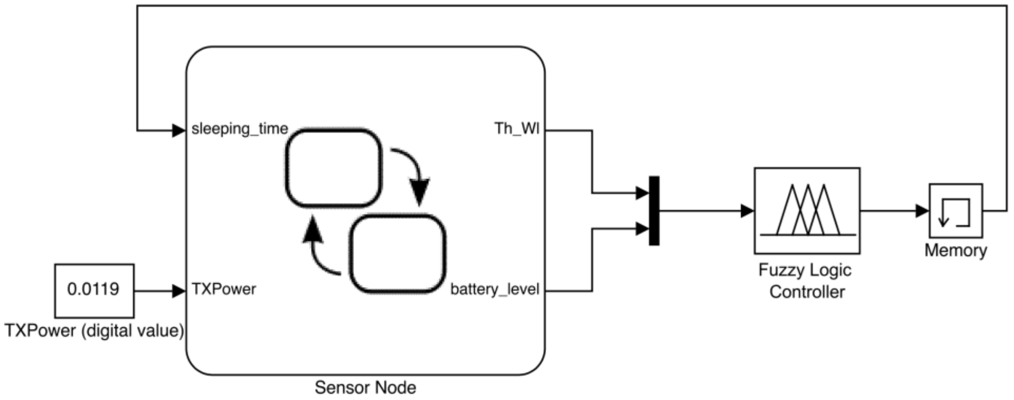


Fig. 5 Simulation model scheme

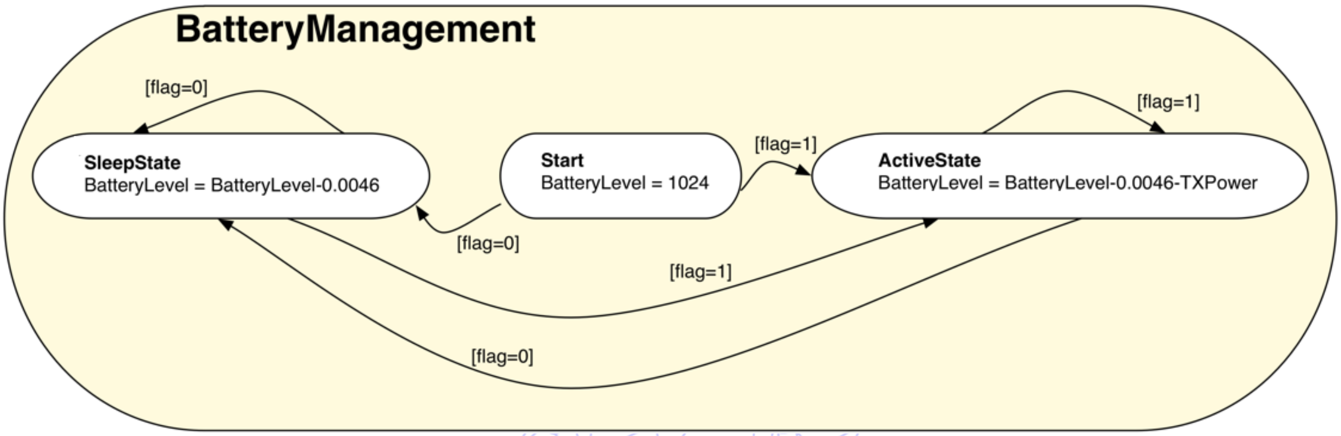


Fig. 6 Battery consumption flowchart

Considering a 10.8V lithium-ion battery, the maximum level of the battery when it is fully charged is 3100mA, while the corresponding digital value, acquired through a 10bit AD Converter, is 1024 (*MaxDV*). It is necessary to note that when the device is in sleeping mode, the power consumption is mainly due to the micro-controller and to the sensor, since the radio frequency transceiver consumption is negligible. In this case, the consumption is 0.0046bit/s. The following relation determines this value:

$$SleepMode_{consumption} = \frac{MC_{PC} * MaxDV}{FullBattery * 3600} \tag{9}$$

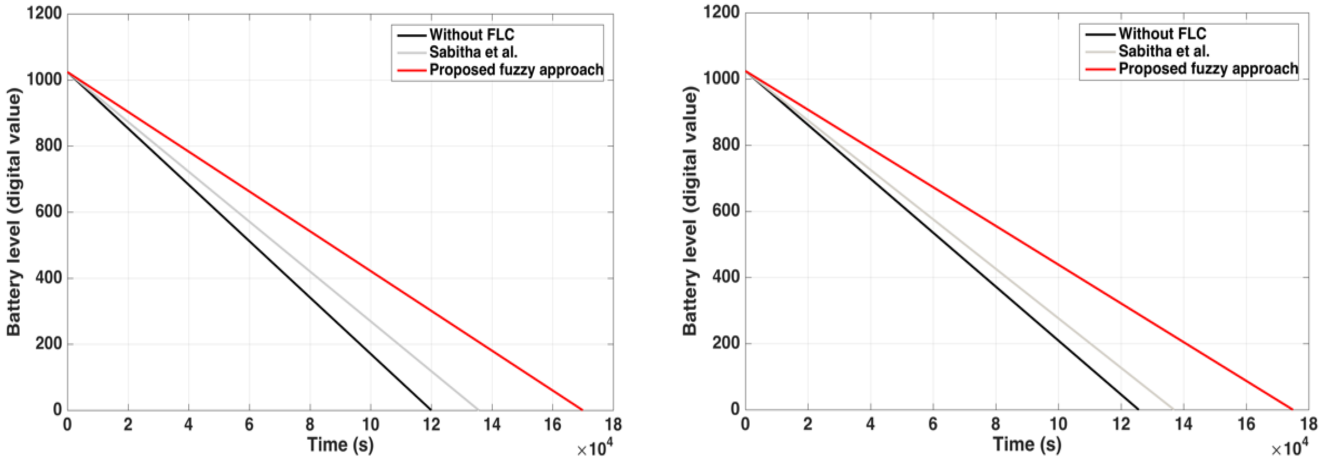
where 3600 is the number of seconds in an hour. By applying the equation 9 the power consumption of each device in sleeping mode is:

$$SleepMode_{consumption} = \frac{50 * 1024}{3100 * 3600} = 0.0046 \tag{10}$$

In case of maximum transmission power (0dB), the radio frequency transceiver consumption is 130mA [31]. As depicted in Fig. 5, the TXPower value is obtained as follows:

$$TransmissionMode_{consumption} = \frac{130 * 1024}{3100 * 3600} = 0.0119 \tag{11}$$

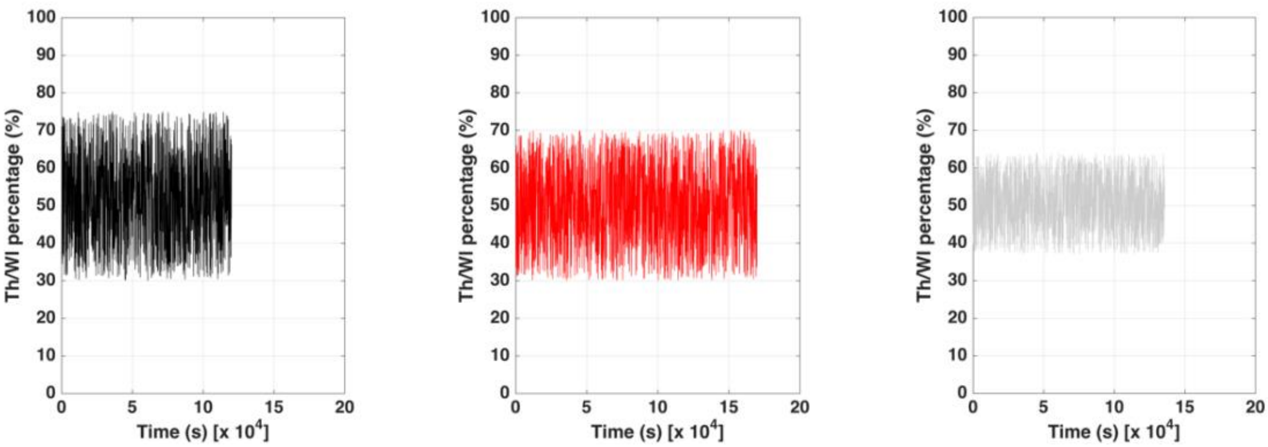
and it is necessary to underline that this value is fixed since the transmission power does not change dynamically in this paper. Finally, the proposed FLC takes the *Th/Wl* and the *battery_level* as input variables and dynamically produces the *sleeping_time*.



(a) Star topology (b) Cluster topology

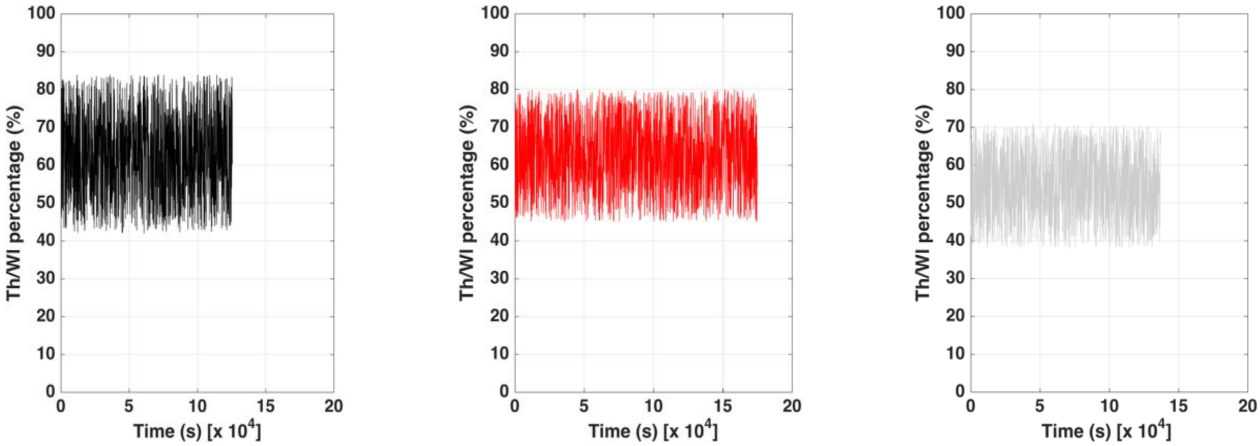
Fig. 7 Battery consumption comparison

During simulations, both the battery level and the ratio of Throughput to Workload (*Th/Wl*) have been evaluated. The battery level during a simulation period of 48 hours is depicted in Fig. 7. In particular, using the proposed approach, a comparison has been carried out considering an approach without the FLC, i.e. assuming that the transmission power (0dB) and the sleeping time (equal to 1, coinciding with the sampling time) are both fixed and with the approach proposed by Sabitha et al. [25]. Fig. 7 clearly shows that the proposed fuzzy-based approach obtains a concrete power consumption reduction and, as a consequence, it prolongs the device lifetime. In both topologies, the best result is obtained by the proposed approach using Gaussian membership functions. In fact, the battery is fully discharged after 169, 950s, i.e. after 47 hours and 21 minutes in the star topology and after is fully discharged after 174, 945s, i.e. after 47 hours and 53 minutes in the cluster topology. This can be explained because, as shown in [34], using Gaussian membership functions the accuracy increases greatly, without degrading the computational performance. For this reason, the FLC based on Gaussian membership functions has been able to better manage the *sleeping_time* of sensor nodes, prolonging much more the battery life. On the contrary, in the case without FLC, the battery is fully discharged much earlier, after 120, 000s, i.e. after about 33 hours and 50 minutes in the star topology and after 125, 671s, i.e. after about 35 hours in the cluster topology. Moreover, using the approach proposed by Sabitha et al. [25] the battery is fully discharged after 135, 683s, i.e. after about 38 hours in the star topology and after 136, 984s, i.e. after about always 38 hours in the cluster topology. Whereby, it is clear that the approach with Gaussian MFS has obtained an improvement of about 30% respect to the case without FLC.



(a) Without FLC (b) Proposed approach (c) Sabitha

Fig. 8 *Th/Wl* behaviour on star topology [25]



(a) Without FLC (b) Proposed approach (c) Sabitha

Fig. 9 *Th/Wl* behaviour on clustered topology [25]

During the simulation, the ratio of *Th/Wl* have been evaluated and its behavior is depicted in Figs. 8 and 9. As it is possible to see, using the fuzzy-based approach proposed in this work the *Th/Wl* fluctuates between 70% to 30% with Gaussian membership functions in the star topology and between 80% to 45% in the clustered topology. With the approach proposed by Sabitha et al. [25], the *Th/Wl* fluctuates between 64% to 37% in the star topology and between 71% to 38% in the clustered topology. On the contrary, the values obtained without FLC fluctuate from 75% to 30% and 84% to 42% in the star and clustered topologies respectively, representing the best results in terms of the ratio of *Th/Wl* but at the expense of the battery consumption. However, it is necessary to note that the values obtained with the proposed FLC, are acceptable especially in those application fields with a moderate variation of data, e.g. temperature, humidity or light detection, where the most important thing is to prolong, as much as possible, the battery lifetime rather than to ensure high *Th/Wl* performance. On the contrary, the *Th/Wl* behavior obtained with the proposed FLC would not be appropriate in a context characterized by real-time constraints, in which the *sleeping_time* of network nodes should not be increased too much in order to obtain and ensure high values of the ratio *Th/Wl*.

6. Conclusions

A fuzzy logic based mechanism has been presented in this paper in order to improve the lifetime of devices in a wireless sensor network for environmental monitoring. An analysis has been done on power consumption of WSNs in order to determine what are the factors to take into account and what approaches can be implemented in order to solve this problem. For this reason, even specific soft computing techniques, such as FLCs, can be used in such a way as to further reduce the energy consumption in a WSN. In fact, the fuzzy-based approach presented in this work dynamically changes the sleeping time in order to increase the battery duration of the sensor devices.

Simulations results have been very promising and demonstrate that using the proposed FLC a substantial reduction of the energy consumption is obtained compared to simulations carried out with fixed sleeping time. The results have been obtained using the Gaussian membership functions because an improvement of 30% has been achieved compared to the approach without FLC. Regarding to the *Th/Wl*, using the proposed fuzzy-based approach a fluctuation between 70% and 20% has been measured and these values are acceptable especially in those application fields with a moderate variation of data where the most important thing is to prolong, as much as possible, the battery lifetime rather than to ensure high *Th/Wl* performance.

Regarding future works, besides the analysis and the use of other membership functions, in order to further validate the proposed approach, several simulations will be carried out to make it suitable also in contexts characterized by real-time constraints.

References

[1] L. Borges, F. Velez and A. Lebres, “Survey on the characterization and classification of wireless sensor network applications,” IEEE Communications Surveys Tutorials, vol. 16, pp.1860-1890, 2014.

[2] J. Gustafsson, R. Kyusakov, H. Makitaavola and J. Delsing, “Application of service oriented architecture for sensors and actuators in district heating substations,” Sensors, vol. 14, pp. 15553-15572, 2014.

[3] D. K. Lee, T. H. Kim, S. Y. Jeong and S. J. Kang, “A three-tier middleware architecture supporting bidirectional location tracking of numerous mobile nodes under legacy WSN environment,” Journal of Systems Architecture, vol. 57, pp. 735-748, 2011.

[4] C. Buratti, A. Conti, D. Dardari and R. Verdone, “An overview on wireless sensor networks technology and evolution,” Sensors, vol. 9, pp. 6869-6896, 2009.

[5] P. Rawat, K. Singh, H. Chaouchi and J. Bonnin, “Wireless sensor networks: a survey on recent developments and potential synergies,” The Journal of Supercomputing, vol. 68, pp. 1-48, 2014.

[6] M. Collotta, G. Scata, S. Tirrito, R. Ferrero and M. Rebaudengo, “A parallel fuzzy scheme to improve power consumption management in Wireless Sensor Networks," Proc. IEEE Emerging Technology and Factory Automation (ETFA), IEEE Press, Sep. 2014, pp. 1-4.

[7] N. Dessart and P. Hunel, “Data collection using WSN for counting individuals and habitat characterization,” Journal of Computational Science, vol. 5, pp. 624-632, 2014.

[8] M. Collotta, G. Nicolosi, E. Toscano and O. Mirabella, “A ZigBee-based network for home heating control,” Proc. 34th Annual Conference of IEEE Industrial Electronics (IECON ’08), IEEE press, Nov. 2008, pp. 2724-2729.

[9] A. B. Gokbayrak, S. Divarci and O. Urhan, “Wireless sensor network gateway design for home automation applications,” Proc. Signal Processing and Communications Applications Conference (SIU), IEEE press, Apr. 2014, pp. 1770-1773.

[10] M. Collotta, S. Tirrito, R. Ferrero and M. Rebaudengo, “An innovative parallel fuzzy scheme for low-power consumption in IEEE 802.11 devices,” Proc. 13th IEEE International Conference on Industrial Informatics, IEEE Press, July 2015, pp. 908-913.

[11] M. Collotta, L. Lo Bello, E. Toscano and O. Mirabella, “Dynamic load balancing techniques for flexible wireless industrial networks,” Proc. 36th Annual Conference on IEEE Industrial Electronics Society(IECON ’10), IEEE press, Nov. 2010, pp. 1329-1334.

[12] F. Losilla, A. J. Garcia-Sanchez, F. Garcia-Sanchez, J. Garcia-Haro and Z. J. Haas, “A comprehensive approach to WSN-Based ITS applications: A survey,” Sensors, vol. 11, pp. 10220-10265, 2011.

[13] M. Collotta, M. Denaro, G. Scata, A. Messineo and G. Nicolosi, “A self-powered wireless sensor network for dynamic management of queues at traffic lights,” Transport and Telecommunication, vol. 15, pp. 42-52, 2014.

[14] M. A. Pasha, S. Derrien and O. Sentieys, “System-Level synthesis for wireless sensor node controllers: A complete design flow,” ACM Trans. Des. Autom. Electron. Syst., vol. 17, pp. 02-24, 2012.

[15] S. Sengupta, S. Das, M. Nasir and B. Panigrahi, “Multi-objective node deployment in WSNs: In search of an optimal trade-off among coverage, lifetime, energy consumption, and connectivity,” Engineering Applications of Artificial Intelligence, vol. 26, pp. 405-416, 2013.

[16] D. Wei, Y. Jin, S. Vural, K. Moessner and R. Tafazolli, “An energy-efficient clustering solution for wireless sensor networks,” IEEE Transactions on Wireless Communications, vol. 10, pp. 3973-3983, 2011.

[17] Q. Wang, M. Hempstead and W. Yang, “A realistic power consumption model for wireless sensor network devices,” Proc. 3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks (SECON ’06), IEEE press, Sep. 2006, pp. 286-295.

[18] L. A. Zadeh, “The concept of a linguistic variable and its application to approximate reasoning II,” Information Sciences, pp. 301-357, 1975.

[19] “Telecommunications and information exchange between systems- local and metropolitan area networks- specific requirements part 15.4: wireless medium access control (MAC) and physical layer (PHY) specifications for low-rate wireless personal area networks (WPANs),” IEEE Standard Information for Technology, Technical report, April 2006.

[20] B. Otal, L. Alonso and C. Verikoukis, “Energy-Efficiency analysis of a distributed queuing medium access control protocol for biomedical wireless sensor networks in saturation conditions,” Sensors, vol. 11, pp. 1277-1296, 2011.

[21] N. Fourty, A. van den Bossche and T. Val, “An advanced study of energy consumption in an IEEE 802.15.4 based network: Everything but the truth on 802.15.4 node lifetime,” Computer Communications, Special issue: Wireless Green Communications and Networking, vol. 35, pp. 1759-1767, 2012.

[22] S. Ouni and Z. Ayoub, “Cooperative association/re-association approaches to optimize energy consumption for real-time IEEE 802.15.4/ZigBee wireless sensor networks,” Wireless Personal Communications, vol. 71, pp. 3157-3183, 2013.

[23] Z. Yu, X. Fu, Y. Cai and M. C. Vuran, “A reliable energy-efficient multi-level routing algorithm for wireless sensor networks using fuzzy petri nets,” Sensors, pp. 3381-3400, 2011.

[24] M. R. Tripathy, K. Gaur, S. Sharma and G. S. Virdi, “Energy efficient fuzzy logic based intelligent wireless sensor network,” Proc. Progress In Electromagnetics Research Symposium, July 2010, pp. 91-95.

[25] R. Sabitha, K. Bhuma and T. Thyagarajan, “Design and analysis of fuzzy logic and neural network based transmission power control techniques for energy efficient wireless sensor networks,” Proc. 3rd International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA), Springer International Publishing, 2014, pp. 295-303.

[26] T. Kalaivani, A. Allirani and P. Priya, “A survey on Zigbee based wireless sensor networks in agriculture,” Proc. 3rd International Conference on Trendz in Information Sciences and Computing (TISC), IEEE press, Dec. 2011, pp. 85-89.

[27] O. Khader, A. Willig and A. Wolisz, “Distributed wakeup scheduling scheme for supporting periodic traffic in wsns,” European Wireless Conference, pp. 287-292, 2009.

[28] I. A. Hameed, “Using gaussian membership functions for improving the reliability and robustness of students’ evaluation systems,” Expert Syst. Appl., vol. 38, pp. 7135-7142, 2011.

[29] T. A. Runkler and M. Glesner, “DECADE - Fast centroid approximation defuzzification for real time fuzzy control applications,” Proc. ACM Symposium on Applied Computing(SAC ’94), 1994, pp. 161-165.

[30] Microchip Technology. PIC24FJ256GB108 - 16-bit PIC and dsPIC Microcontrollers, 2009.

[31] Microchip Technology. MRF24J40MB - rfPIC, 2009.

[32] M. Collotta, “FLBA: A fuzzy algorithm for load balancing in IEEE 802.11 networks,” Journal of Network and Computer Applications, vol. 53, pp. 183-192, 2015.

[33] Maxim Integrated. DS18B20 - Programmable Resolution 1-Wire Digital Thermometer, 2008.

[34] V. O. S. Olunloyo, A. M. Ajofoyinbo and O. Ibidapo-Obee, “On development of fuzzy controller: The case of gaussian and triangular membership functions,” Journal of Signal & Information Processing, vol. 2, pp. 257-265, 2011.

International Journal of Engineering and Technology Innovation, vol. 6, no. 1, 2016, pp. 68 - 78

Reliable and Congestion Control Protocols for Wireless Sensor Networks

Kirti Kharb*, Bhisham Sharma, Dr. Trilok C. Aseri

Department of Computer Science Engineering, PEC University of Technology, Chandigarh, India

Received 05 November 2015; received in revised form 13 November 2015; accepted 20 November 2015

Abstract

The objective of this paper is to analyze review and different congestion control protocols that are employed at the transport layer and some of them working at the medium access control layer in wireless sensor networks. Firstly, a brief introduction is given about wireless sensor networks and how congestion occurs in such networks. Secondly, the concept of congestion is discussed. Thirdly, the reason of occurrence of congestion in wireless sensor networks is analyzed. After that, congestion control and why it is needed in the wireless sensor networks is discussed. Then, a brief review of different congestion control and reliable data transport mechanisms are discussed. Finally, a comparative analysis of different protocols is made depending on their performance on various parameters such as - traffic direction, energy conservation characteristic, efficiency etc. and the paper is concluded.

Keywords: wireless sensor networks, reliability, congestion avoidance, transport layer protocols

1. Introduction

Wireless sensor networks are a group of heterogeneous nodes called sensor nodes, spread over a large field with a central processing node called as sink. Basically, wireless sensor networks perform two main actions – wireless sensing and data networking. They provide a bridge between the real physical and virtual worlds. Wireless sensor networks have diversified arena of applications, some of them being in health care, animal care monitoring and surveillance, logistics & transportation, soil health maintenance for agriculture, real time security and surveillance, infrastructure etc. However, there are also some concern issues pertaining to wireless sensor networks. The main concern areas related to wireless sensor networks are - resources, energy wastage, memory utilization, and computational power. [1] Most discussed issue is that of energy consumption and battery usage problems. Next issue in wireless sensor networks is the issue of the occurrence of congestion in the networks. Another issue is that of the security. In this paper, we aim to review and compare different reliable and congestion control algorithms working at the transport layer in wireless sensor networks. How and why it is important to address, detect and control the congestion, and the many mechanisms/ algorithms which help us in doing so is also discussed. Section II provides the study that motivates why it is inadvertent to go for congestion control, the section III deals with the problem of congestion, congestion avoidance and congestion control. Section IV reviews different schemes employed to control congestion in wireless sensor networks. A comparative analysis of different congestion control algorithms is done in the concluding section.

2. Congestion in Wireless Sensor Networks

This section makes a brief explanation about congestion, the congestion process and the different types of congestion happening in wireless sensor networks.

* Corresponding author. E-mail address: kharb.kirti46@gmail.com

2.1 Brief Overview

Traffic flows in sensor networks is shaped according to the physical structure of the fields that they work in. Wireless sensor networks generally operate under light load conditions and become active whenever an event occurs. If the corresponding application load is high, it may result in the generation of huge continuous data flows leading to disrupted performance of the network, which may lead to congestion. In such condition, collision occurs in the network, due to which data packets start getting dropped, buffer overflows start happening at the nodes in the network.

Therefore, we can say that the congestion is said to occur in a network, if the speed of the incoming traffic is larger than the data processing rate of the network it. Following are some of the probable causes due to which congestion happens in wireless sensor networks:

- processing speed of nodes is low
- incoming traffic arrives at faster rate than it can handle
- if packet collisions happen on data links, leading to alternate routing of packets & excessive re-transmission of packets

2.2 The Congestion Process

The Congestion happens in a network either due to buffer overflow (when packets are getting stored at the source because the previous packets have not been delivered) or/and link collisions (which results into packet losses and large number of re-transmissions) [2]. Depending on physical topology of the network, both types of above situation may occur. In Buffer overflow situation, data packets get dropped which negatively impacts the application, since the throughput is restricted to the maximum data sending rate of node. As a result continuously packets get dropped; a lot of power is wasted in re-transmissions. To counter this, a congestion control algorithm is to be used which either reduces the data rate of the transmitting nodes or reroutes the excess data packets through alternative paths.

If link collisions occur, all the nodes in the network receive limited number of packets and the reliability of the application running on the network is badly affected. For mitigating this, a congestion control algorithm that focuses on the MAC layer may be used, that maintains optimum and collision free access to the medium.

2.3 Different types of Congestion in Wireless Sensor Networks

The congestion in WSNs can be stratified in two major classes depending upon

- how data packets are lost, and
- where in the network congestion is taking place

2.3.1 How data packets are lost

Due to congestion, packets can be lost in the following two ways in the wireless sensor networks:

1) **Packet Collisions in the Medium**: At a particular instant of time in a geographical area, many nodes within vicinity of one another attempt to transmit data simultaneously, resulting in data losses due to interference and thereby reduce the throughput of all nodes in the area. [3] Proper local synchronization among neighbor nodes can reduce this type of loss, however, we cannot eliminate it completely because non-neighboring nodes may still interfere in the data transmission.

2) **Packet Drops Due to Buffer Overflow**: Within a particular sensor node, if the queue or buffer used to hold data packets that are to be transmitted, overflows then congestion occurs. In this case, nodes receive packets with a rate higher than that they can transmit or process. This kind of congestion usually occurs in wired networks.

2.3.2 Where in the network packets are lost

After congestion occurs, there are three possible sites for the data packets to get lost. These three sites are:

1) **Hotspot Near Source—Source Congestion:** The hotspot referred in this paper means the area in the network where congestion occurs and packets start getting dropped. Whenever the sensors are deployed densely, the data packets which are generated during a crucial event will create hotspots very close to the sources. In this scenario, localized and quick time-scale mechanisms which are capable of providing backpressure messages from the nodes that cause congestion, back to the sources would be helpful for immediate traffic controlling until the congestion is removed by other means.

2) **Hotspot Near the Sink-Sink Congestion:** Even sparsely deployed sensors that generate data at comparatively low data rates may create hotspots in the sensor network, but likely farther away from the sources, near the sink. In such a scenario, combined use of localized back-pressure and packet dropping methods would be more effective. Another way of removing the sink congestion is to employ multiple sinks that are uniformly distributed across the sensor network and, therefore, traffic is balanced among these sinks.

3) **Forwarder Congestion:** A sensor network usually has more than one flow (sink-source pair), and these flows will intersect with one another. The area around such points of intersection is likely to become a hot spot for congestion. In a tree-like communication network, every intermediate node may suffer from the problem of forwarder congestion. [4] Compared to the other congestion scenarios, forwarder congestion is way more challenging because it is very difficult to predict the intersection points due to the highly dynamic nature of the network. In this case, even sparsely situated sensor nodes generating data will create both transient as well as persistent hotspots distributed throughout the sensor field.

3. Congestion Control Methods

In this section, firstly the congestion control process is discussed, then the different congestion control schemes are analyzed and finally the congestion control mechanism is discussed.

3.1 Congestion Control

Congestion control is the mechanism through which congestion is prevented from being occurring in a wireless sensor network, and if congestion has already happened then to detect where it has occurred, to monitor its status and controlling its aftermath. . In buffer overflow scenario, reduction of data sending rate is done on the nodes whose buffer is overflowing or re-transmissions are done through some alternative paths. To overcome congestion in link collisions, a congestion control algorithm that focuses on the MAC layer may be used to help co-ordinate the network access among the nodes.

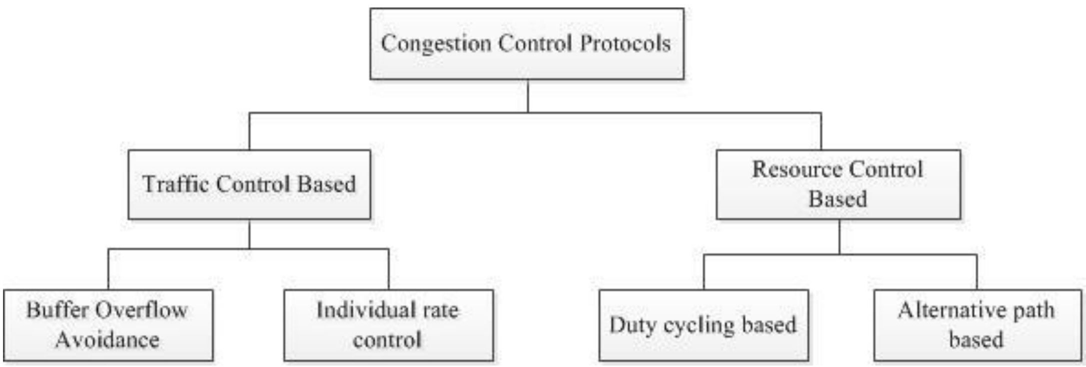


Fig. 1 Different categories of congestion control protocols

3.2 Congestion Control Schemes

Generally, protocols that deal with congestion control in WSNs can be basically classified in three major categories. These categories are: Congestion Control, Congestion Avoidance, and Reliable Data Transmission protocols.

3.2.1 Congestion 'Control' Protocols

These protocols can be classified on the way they detect congestion, the way they notify the other nodes for the occurrence of congestion, as well as how the congestion countering mechanisms are performed. Congestion detection by these protocols is done by: checking the level of buffer occupancy at the nodes in the network, measuring the load level of the channel, and by counting the packet service timings and packet inter-arrival time. [5] Congestion notification is done by either using additional notification packets that are sent across the network or by overhearing the data packets that are being transmitted across. Congestion counter mechanisms can be performed either through reduction of traffic or through the creation of alternative paths from the source node(s) to the sink node(s).

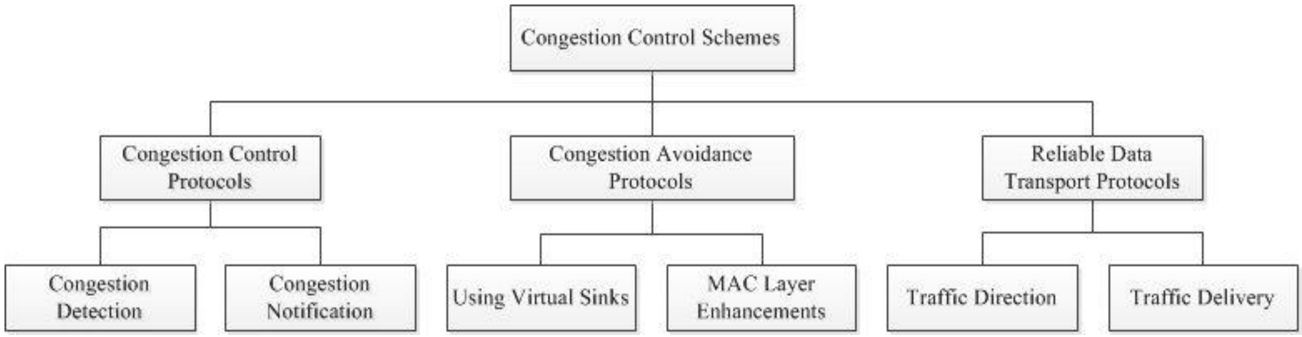


Fig. 2 Different types of Congestion Control Schemes

3.2.2 Congestion 'Avoidance' Protocols

These protocols are classified on the basis of how congestion is going to occur and on what mechanisms will be used avoid the congestion from even occurring in a network. Congestion detection is done same as in congestion mitigation protocols. Congestion can be avoided by using virtual sinks for those sinks which are getting highly congested and by using MAC layer enhancements.

3.2.3 Reliable Data Transport Protocols

The reliable data transmission protocols are the protocols that, not only put effort to control congestion in a network, but also attempt to recover all or part of the information which is lost. These are normally employed when all of the information in network is critical for the application and is using some transport layer mechanisms. These protocols are divided on the basis of three attributes: their traffic direction (whether downstream or upstream), whether they provide hop-to-hop or end-to-end delivery, and the parameter on which reliability focuses (packet or event).

4. Different Congestion Control, Avoidance and Reliable Data Transport Protocols

The different protocols employed for congestion control in wireless sensor networks, basically deal with three core functions- controlling the congestion, avoiding the congestion and providing for reliable data transportation. Following sections deal with this concept in detail.

4.1 Congestion Control Protocols

These protocols mainly perform three functions - congestion detection, congestion notification (notifying the nodes in the network that congestion has occurred) and the congestion mitigation strategy.

4.1.1 Adaptive Rate Control (ARC) Algorithm

The ARC scheme was proposed by J. Zhao et al. in 2001 [6]. This algorithm does not involve any congestion detection or notification mechanisms. It uses an AIMD-like traffic control mechanism for congestion mitigation, which works as follows: an intermediate node increases its sending rate by a constant *a* if it senses a successful packet forwarding by its parent node. Otherwise, the intermediate node multiplies its sending rate by a factor *b*, where $0 < b < 1$. In order to guarantee fairness, ARC basically maintains two independent sets of these two factors: *a* and *b*, for source and transit traffic respectively. Studies done by authors have shown that ARC is not only effective in achieving fairness but it also maintains good bandwidth along with reasonable energy efficiency, especially during low traffic situations that are the common case in wireless sensor networks.

4.1.2 Congestion Detection and Avoidance algorithm (CODA)

This algorithm [7] deals with both - congestion control and avoidance in wireless sensor networks. Present and past channel load conditions are used by CODA as the congestion indication parameters. The channel load is listened by each node before it starts transmitting data i.e. carrier sensing is pro-active here, this is done to avoid congestion. If congestion is detected, the receiver node broadcasts an explicit back-pressure message to its neighbors signaling that congestion has happened and correspondingly neighbor nodes reduce their local data sending rates. If congestion persists for a longer time, this back-pressure is sent all the way to the sink node. A constant feedback in the form of ACK packet is maintained from the source to the sink nodes. If the source node does not receive ACK, then it reduces its data transmission rate. The CODA algorithm does not ensure fairness in the network and the bandwidth is not utilized efficiently.

4.1.3 Congestion Control and Fairness (CCF)

Saima Zafar [8] proposed this scalable algorithm for many to-one routing in wireless sensor networks. Here, congestion detection is performed on the basis of packet servicing time which is the time period taken in sending the data packet form the transport layer to the network layer and reception of the successful data transmission. Traffic control is used to carry out congestion mitigation. Congestion control is performed in hop-by-hop fashion and each node uses exact rate adjustment based on its available packet servicing rate. Each child node divides its data sending rate by the number of children nodes it has, compares this newly arrived data rate with its parent data rate. If the defined threshold congestion level is reached in the network, then the algorithm requests the child nodes to reduce their data transmission rate. Since a separate queue is maintained for every child node, a considerable amount of bandwidth is exhausted in this algorithm.

4.1.4 Fusion Algorithm

Proposed by Mohamed Amine Kafi et al., [9] this scheme is used for mitigating the congestion in wireless networks. This algorithm uses explicit congestion detection method i.e. a congestion bit in the header of every data packet. Whenever, congestion occurs it is notified to all the nodes in the neighborhood by setting the value of congestion bit to 1 in every outgoing packet from a node. Congestion detection can also be done by checking upon the queue size of each sensor node. If the queue length crosses a specified threshold limit, the congestion bit is set, else it is reset. If congestion lasts for longer, then hop-by-hop back pressure message is sent to the source notifying it to reduce the data transmission rate. A prioritized MAC scheme is used to provide access to channel medium when the nodes are congested.

4.1.5 Biased Geographical Routing (BGR) Algorithm

Proposed by Dashkova et al. [10], in this protocol whenever congestion is detected the nodes reactively split traffic. Detection of congestion is done depending on the level of buffer occupancy and wireless usage of the network. The wireless usage is calculated by taking periodic samples of wireless medium. An implicit congestion bit added to each data packet is used for congestion notification. Each node listens to the packets sent by its neighbors to detect the congestion status of the network. Congestion is resolved using two algorithms: In-Network Packet Scatter (IPS) and End-to-End Packet Scatter (EPS). IPS removes short term congestion in the network by splitting the traffic immediately before the congested areas. Whereas, EPS removes long term congestion by splitting the data flow at source node and then performs rate control by the Additive Increase Multiplicative Decrease (AIMD) strategy.

4.1.6 Hierarchical Tree Alternative Path Algorithm (HTAP)

Hierarchical Tree Alternative Path algorithm is proposed for event-based sensor applications. Propounded by Charalambos Sergiou et al., [11] it tries ensuring application reliability during overload periods without reducing the source node's data rate while sending information during critical events. HTAP works on a combination of two algorithms, Alternative Path Creation (APC) and Hierarchical Tree Creation (HTC), and it uses the network density to choose between them. When congestion takes place or a node's battery is about to draining, APC and HTC form alternative paths to the sink by unused nodes. APC uses these nodes by randomly exploiting neighboring table, while in HTC these nodes are placed in a hierarchical levelled tree starting from 0 for the leaves nodes. Every node piggybacks its buffer occupancy, reflecting its congestion state, when sending packets and the neighbors refresh their neighboring state tables when overhearing packets. A congested receiver sends a back-pressure packet to the sender in the purpose to remove congestion. The sender stops transmitting to this node and searches for a less congested receiver which leads to alternative paths creation.

4.2 Congestion Avoidance Protocols

These protocols are more focused on the 'avoidance part' of the congestion control process. Basically involve two strategies- congestion detection methodology and congestion avoidance mechanism.

4.2.1 Siphon

This scheme was proposed by G. Srinivasan *et al.* [12]. It is a source to-sink congestion control protocol. Congestion detection is done through – buffer occupancy level, sink load level and wireless channel load conditions. Congestion notification method involved is implicit in nature. Congestion mitigation is performed by traffic redirection through virtual sinks (VSs). These virtual sinks form a dynamic adhoc network and split traffic when nodes get congested, thereby preventing the data packets from getting deleted. This algorithm comprises mainly these steps – first being the discovery of virtual sinks that will be selected for congestion control in particular congested areas. For this every sensor node maintains a table of VS it has in its vicinity. The second step is to use the service of a VS, a congested node enables a redirection bit which signals a VS to re-route the traffic out of the congested neighborhood, for this every VS has dual radio interfaces maintained on it – a long range interface to communicate with other VSs and a short range interface to communicate with the sensor nodes in the network. Finally, when the VS detect the value of redirection bit as set, it re-routes the packets using a combination of hop-by-hop and end-to-end transmission.

4.2.2 Buffer and Rate Control Based Congestion Avoidance Algorithm

M. M. Alam *et al.* [13] proposed the "Buffer and Rate Control Based Congestion Avoidance" protocol. There are three main steps involved in this algorithm. These are: the Upstream Source Counting, the Buffer Occupancy based rate control, and the Snoop based MAC level ACK. The first two steps are used to control the rate of upstream nodes. It actually provides two advantages. The first is that congestion is reduced by media access contention, as the upstream nodes proactively decrease their data transmission rate, while the second advantage is that the congestion due to buffer overflow is avoided as the upstream nodes delay the transmission of packets whenever the buffer capacity of the downstream nodes is full. In the third step, explicit ACK packets are avoided. Instead, each node may overhear its own transmitted packet while forwarded by its downstream node, hence eliminating the need for ACK packets. To accomplish this overhearing task, the upstream node MAC address and a sequence number need to be appended into the MAC frame of the node. Many advantages of this protocol are that this can reduce collision drop Rate, increase delivery ratio and improve the network's energy efficiency.

4.2.3 Priority Based Medium Access Protocol

The Priority Based Medium Access Protocol for Congestion Avoidance was proposed by Patil et al [14]. This is a MAC layer protocol that helps in avoiding congestion by giving proportional access to the nodes based on their source count values. For example, a node carrying a higher load of data traffic gets more access time than others. Each sensor node then calculates its contention window on a provided equation. This contention window has different size for different node data conditions. Simulation series that have been performed by scholars in MATLAB resulted that there is an optimal contention window size through which the collision in the MAC layer can be minimized and enables all the sensor nodes to transmit their data packets without delays.

4.2.4 LACAS

The concept of “Learning Automata-Based Congestion Avoidance Algorithm in Sensor Networks” (LACAS) was propounded by Vrisha Tickoo *et al.* [15] which is actually an adaptive learning solution for avoiding the congestion in wireless sensor networks. The target of this algorithm is to control the data rate of intermediate nodes in order to avoid congestion before it reaches to the sink in the network. To monitor the data rates, automatas are developed at each of the network’s nodes that are capable of controlling the rate of flow of data at the intermediate nodes. These states are based on probabilistically how many packets are likely to get dropped if a particular flow rate is maintained. In this case, an “automaton” “learns” from past behaviors, will increase data rate if packets are not being dropped or else will reduce the data rate from the previous level of data rate in order to avoid congestion. This algorithm actually works on the reinforcement learning strategy and keeps on optimizing the data rate depending upon its past performance achieved.

4.3 Reliable Data Transport Protocols

These protocols mainly aim at providing reliable data transmission and traffic control in the wireless sensor networks.

4.3.1 Event to Sink Reliability (ESRT) Protocol

Sankarasubramaniam *et al.* [16] proposed this protocol. ESRT aims for reliability at the application level and provides reliable delivery of packets from sensors to the sink. By regulating sensor frequency, this protocol tries to guarantee end-to-end reliability. However, reliability is maintained for the whole application and not for each single data packet. Congestion feedback from sensor nodes is broadcasted, notifying to adjust the reporting rate in the network so that the sensor nodes are able to receive sufficient number of packets but only as much as packets necessary in order to avoid congestion and save energy. ESRT runs on the sink, with negligible functionality needed at the sensor nodes. The protocol operates by determining the reliability achieved and congestion condition in the current network state. [17] Firstly, it periodically computes the reliability r based on how many packets are received successfully in a time interval. In the second step, the protocol deduces the required frequency f of the sensor nodes from r. Finally, ESRT informs all the sensor nodes about f through an assumed channel with high power. ESRT identifies five distinct regions in which it operates: i) No Congestion, Low reliability, ii) No Congestion, High reliability, iii) Congestion, High Reliability, iv) Congestion, Low Reliability and v) Optimal Operating Region—which actually is the region with No Congestion, Medium-High Reliability. The aim is to identify the current operating state of the network and to bring it into Optimal Operating Region. The event-to-sink reliability is checked, if found lower than required, the reporting frequency of source nodes is adjusted aggressively in order to still maintain the target reliability level; if the reliability is higher than required, then the reporting frequency is conservatively reduced so that energy can be conserved while still maintaining the reliability of network. Thus, this self-configuring nature of ESRT protocol makes it robust even with dynamic changing topologies in the network. The best benefit resulting from ESRT [18] is its capability of energy-conservation by dynamically controlling the sensor reporting frequency. A disadvantage associated with ESRT is the fact that all nodes are treated equally due to which in case of congestion in one region of the network, all the nodes are forced to reduce their data rate, negatively affecting the network’s throughput. Thus, it is able to provide fairness among the nodes since data rate reduction is applied on all the nodes in the network, even if there is congestion in a particular area in the network.

4.3.2 GARUDA Protocol

S. Brahma et al. [19] have discussed this protocol which is a reliable data transport protocol. This protocol provides reliable point-to-multipoint data delivery from the sink node to the sensor nodes. It comprises of the following elements

- an efficient pulse based solution for reliable short messages delivery;
- a virtual infrastructure called the core, that is instantaneously constructed during the course of a single packet flood; which is used to approximate a near optimal assignment of all the local designated servers,
- a two-stage NACK (negative acknowledgment) based recovery process that minimizes the overheads resulting from the retransmission processes in the network, and performs out-of-sequence forwarding to leverage the significant spatial re-use possible in a WSN;

The traffic direction flow implemented in GARUDA is downstream and it provides both: packet and destination related packet reliability.

4.3.3 STCP Protocol

It is a scalable and reliable transport layer protocol where the majority of functionalities are dealt at the sink [20]. It supports networks with multiple applications and provides additional functionalities such as controlled variable reliability as well as congestion detection and avoidance. In this protocol, before transmitting any packet, the sensor nodes inform the sink through a "Session Initiation Packet". [21] Through this initiation packet, the sink gets to know about the number of flows initiated from a source node, the type of data that is to be transmitted, the transmission rate, and the required reliability. When the moment the sink node receives this packet, it sends an ACK packet to the source node, and only then the source node starts transmitting the packets. Since the sink is aware of the rate of transmission from the source, the expected arrival time of the next packet can be determined. The sink node maintains a timer and sends a negative acknowledgement packet, if it does not receive a packet within the expected arrival time. Reliability is measured as the fraction of total packets successfully received by the network. For controlling the congestion, nodes inform the sink [22] whether they are experiencing any buffer overflow situation, and by setting their congestion notification bit value equal to 1, while the sink informs the source node about a congested path by setting the congestion bit on the ACK packet. In this case, a source node may change its routing path or decrease the data sending rate to mitigate congestion.

4.3.4 RCRT Protocol

Paek *et al.* proposed this protocol [23], which focuses on reliable delivery of data from the source to sink, while avoiding any intermediate congestion collapse. It works on the transport layer and its traffic management functionality is implemented on the sink. RCRT attempts to achieve 100% reliable data delivery in the network based on a NACK scheme. So, in case, there are packet losses, the sink requests the source for retransmission of the missing packets by sending a NACK with the missing packet numbers. RCRT implements following basic components at the sink:

- A congestion detection component which detects congestion in the network by checking upon the round trip time values, rate adaptation, and rate allocation, which if found more than the expected values means congestion has occurred and there is a need to decrease the flow rates to control congestion;

- The 'time to recover loss' [24] is used as an indicator of congestion detection in the wireless sensor network. Therefore, as long as the network is able to repair the packet losses (within around the Round Trip Time) the network in not congested.
- However, if the packet losses cannot be redeemed by the network then it figures out that there are congested spots in network.
- In case congestion occurs, RCRT applies a rate adaptation mechanism [25] to control it. With the help of the rate allocation mechanism, specific transmission rates are allocated to each data flow whenever the application on which the network is running changes.

5. Comparative analysis of the congestion control protocols discussed in the paper

The following table (Table 1) compares the congestion control protocols- ARC, CODA, CCF, Fusion, BGR and HTAP over the parameters of congestion detection mechanism, congestion notification methodology used [26] (whether it is implicit or explicit), the direction of traffic flow (whether from source to sink, or from sink to source [27]), whether the protocol is able to achieve fairness or not and the performance of protocol on the energy conservation [28].

Table 1 Congestion 'Control' Protocols

Protocol/ Mechanism	Congestion Detection	Congestion Notification	Congestion Mitigation	Traffic Direction	Fairness Achieved	Energy Conservation
ARC	Detected by whether the packets are successfully forwarded or not	Implicit	Traffic Control	Source to Sink	Yes	Medium
CODA	Buffer occupancy level and load level of the wireless channel	Explicit	Traffic Control	Source to Sink	No	High
CCF	Packet Service Time	Implicit	Traffic Control	Source to Sink	Yes	Low
Fusion	Buffer occupancy level and load level of the wireless channel	Implicit	Traffic Control	Source to Sink	No	High
BGR	Buffer occupancy level and load level of the wireless channel	Implicit	Resource and Traffic Control	Source to Sink	Yes	N/A
HTAP	Buffer Occupancy level	Implicit	Recourse Control	Source to Sink	No	High

The following table (Table 2) compares the congestion avoidance protocols – Siphon, LACAS, Priority based Medium Access Protocol and the Buffer and Rate Control based Congestion Avoidance Algorithm on two parameters – the congestion detection mechanism adopted by the protocols and the congestion avoidance mechanism implemented [29].

Table 2 Congestion Avoidance Protocols

Protocol/ Mechanism	Congestion Detection	Congestion Avoidance Mechanism
Siphon	Buffer Occupancy, Sink load and wireless channel load conditions	Traffic redirection through virtual sinks
Buffer and Rate Control based Congestion Avoidance Algorithm	Buffer Occupancy and wireless channel load	Traffic Control
Priority based Medium Access Protocol	Buffer Occupancy	The most congested nodes are given the highest priority and given the channel access also on priority
LACAS	N/A	Learning automata states to adjust flow rates

The following table(Table 3) compares ESRT, GARUDA, STCP, and RCRT, the reliable data transport protocols on three parameters – the reliability mechanism adopted by the protocol(whether it is hop-by-hop or end-to-end), the traffic direction (whether it is downstream or upstream) and the level at which the reliability is achieved by the protocol [30].

Table 3 Reliable Data Transport Schemes

Protocol/ Mechanism	Reliability Mechanism	Traffic Direction	Reliability level achieved
ESRT	Hop-by-Hop	Downstream	Packet
GARUDA	Hop-by-Hop	Downstream	Packet and Destination Related
STCP	End-to-End	Upstream	Event and packet
RCRT	Hop-by-Hop	Upstream	Packet

6. Conclusion

In this paper, we presented a brief review of reliable and congestion control protocols in wireless sensor networks. Congestion control protocols are reactive protocols whereas the congestion avoidance protocols work in pro-active manner. The reliable transport protocols basically ensure that the data to be transmitted through the wireless sensor networks reaches correctly at its intended destination and, in this process, the data is not getting corrupted, that is the integrity of the data is maintained. The data reliable transmission protocols are needed in areas having real time and crucial data applications. The study of different protocols is done and a comparative analysis of the protocols is concluded. Congestion has a deep impact on the energy consumption, efficiency, packet delivery ratio, delay in data delivery and the lifetime of a wireless sensor network, hence it is very important to monitor and control the congestion. The review conducted on congestion control protocols has shown that the type of application and data flow type influence the traffic control deeply. Since reliability is the crux functional area of the transport layer, it is crucial to ensure the dependability of the applications operating on wireless sensor networks.

References

[1] B. Sharma and T. C. Aseri, “A comparative analysis of reliable and congestion-aware transport layer protocols for wireless sensor networks,” International journal of Sensor Networks, vol. 2012, 14 pages, Dec. 6, 2012.

[2] B. Sharma and T. C. Aseri, “A hybrid and dynamic reliable transport protocol for wireless sensor networks,” Computer and Electrical Engineering, vol. 48, pp. 298-311, November 2015.

[3] Rahman, Md Obaidur, M. M. Monowar, and C. S. Hong, "A capacity aware data transport protocol for wireless sensor network," Computational Science and Its Applications(ICCSA ’09), Springer Berlin Heidelberg, pp. 491-502.

[4] Flora, D. F. Jenolin, V. Kavitha, and M. Muthuselvi, "A survey on congestion control techniques in wireless sensor networks," International Conference on Emerging Trends in Electrical and Computer Technology (ICETECT ‘11), IEEE press, 2011.

[5] R. Prabha, P. K. Gouda, Manjula S H1, K R Venugopal, and L M Patnaik, “ MTADF : multi hop traffic aware data for warding for congestion control in wireless sensor networks,” International Journal of Wireless & Mobile Networks (IJWMN ‘15), Feb. 2015, vol. 7, no. 1.

[6] J. Zhao, L. Wang, S. Li, X. Liu, Z. Yuan, and Z. Gao, (2010, October). “A survey of congestion control mechanisms in wireless sensor networks,” The Sixth International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP ‘10), IEEE press, 2010, pp. 719-722.

[7] J. Wan, H. Yan, Q. Liu, K. Zhou, R. Lu, and D. Li, “Enabling cyber–physical systems with machine–to–machine technologies,” International Journal of Ad Hoc and Ubiquitous Computing, vol. 13, no. 3-4, pp.187-196.

[8] S. Zafar, “A survey of transport layer protocols for wireless sensor networks,” International journal of Computer applications, vol.33, no.1, pp.44-50, November 2011.

[9] M. A. Kafi, et al., “Congestion control protocols in wireless sensor networks: A survey,” IEEE Communications surveys and tutorials, vol.16, no. 3, pp.1369-1390, March 2014.

[10] E. Dashkova and A. Gurtov, “Survey on congestion control mechanisms for wireless sensor networks,” Internet of Things, Smart Spaces, and Next Generation Networking, vol.7469, pp.75-85, 2012.

[11] C. Sergiou, V. Vassiliou, and A. Paphitis, “Hierarchical tree alternative path (HTAP) algorithm for congestion control in wireless sensor networks,” Journal of Ad-hoc Networks, vol.11, no. 1, pp. 257-272, January 2013.

[12] G. Srinivasan and S. Murugappan, "A survey of congestion control techniques in wireless sensor networks," International Journal of Information Technology and Knowledge Management, pp. 413-415, 2011.

[13] Alam, M. Mahbub, and C. S. Hong, "Buffer and rate control based congestion avoidance in wireless sensor networks," Proceedings of Korea Information Processing Society, pp. 1291-1293, May 2007.

[14] Patil, Dipti, and S. N. Dhage, "Priority-based congestion control protocol (pccp) for controlling upstream congestion in wireless sensor network," International Conference on Communication, Information & Computing Technology (ICCICT ‘12), IEEE press, 2012.

[15] V. Tickoo and S. Gambhir, "A comparison study of congestion control protocols in WBAN," International Journal of Innovations and Advancement in Computer Science, vol. 4, no. 6, June 2015.

[16] Sankarasubramaniam, Yogesh, Ö. B. Akan, and I. F. Akyildiz, "ESRT: event-to-sink reliable transport in wireless sensor networks," Proc. of the 4th ACM International Symposium on Mobile Ad hoc Networking & Computing, pp. 177-188, 2003.

[17] V. Michopoulos, L. Guan, G. Oikonomou, and I. Phillips, “A comparative study of congestion control algorithms in IPv6 wireless sensor networks,” Proc. IEEE Int. Conf. Distributed Computing in Sensor Systems and Workshops (DCOSS ‘11), June 2011, pp. 1–6.

[18] M. Palattella, N. Accettura, X. Vilajosana, T. Watteyne, L. Grieco, G. Boggia, and M. Dohler, “Standardized protocol stack for the internet of (Important) things,” IEEE Communication Surveys Tutorials, vol. 15, no. 3, pp. 1389–1406, 2013.

[19] Brahma, Swastik, M. Chatterjee, and K. Kwiat, "Congestion control and fairness in wireless sensor networks," The 8th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops), IEEE press, 2010, pp. 413-418.

[20] W. W. Fang, J. M. Chen, L. Shu, T. S. Chu, and D. P. Qian, “Congestion avoidance, detection and alleviation in wireless sensor networks,” J. Zhejiang Univ.—Sci. C, vol. 11, no. 1, pp. 63–73, 2010. Available: http://dx.doi.org/10.1631/jzus.C0910204

[21] K. Zheng, F. Hu, W. Wang, W. Xiang, and M. Dohler, “Radio resource allocation in LTE-advanced cellular networks with M2M communications,” IEEE Communications Magazine, vol. 50, no. 7, pp. 184–192, 2012.

[22] J. Jin, M. Palaniswami, and B. Krishnamachari, “Rate control for heterogeneous wireless sensor networks: Characterization, algorithms and performance,” Computer Networks, vol. 56, no. 17, pp. 3783–3794, November 2012. Available: http://www.sciencedirect.com/science/article/pii/S1389128612003131

[23] Paek, Jeongyeup, and R. Govindan, "RCRT: rate-controlled reliable transport for wireless sensor networks," Proc. of the 5th international conference on Embedded networked sensor systems, pp. 305-319, 2007.

[24] Spachos, Petros, P. Chatzimisios, and D. Hatzinakos, "Cognitive networking with opportunistic routing in wireless sensor networks," IEEE International Conference on Communications (ICC ‘13), IEEE press, 2013, pp. 2433-2437.

[25] P. Spachos, Petros, P. Chatzimisios, and D. Hatzinakos, "Energy aware opportunistic routing in wireless sensor networks," Globecom Workshops (GC Wkshps), IEEE press, 2012, pp. 405-409.

[26] C. sergiou et al, "A comprehensive survey of congestion control protocols in wireless sensor networks," IEEE Communications surveys and tutorials, vol.16, no. 4, pp.1839-1859, December 2014.

[27] Ali Ghaffari, "Congestion control mechanisms in wireless sensor networks: A survey," Journal of Network and Computer Applications, vol. 52, pp. 101–115, June 2015.

[28] I. Khan, F. Belqasmi, R. Glitho, N. Crespi, M. Morrow, P. Polakos, "Wireless sensor network virtualization: A survey," Communications Surveys & Tutorials, vol.18, no.1, pp. 553-576, 2016.

[29] Mocanu, D. Constantin, Vega, M. Torres, Liotta, and Antonio, "Redundancy reduction in wireless sensor networks via Centrality Metrics," IEEE International Conference on Data Mining Workshop (ICDMW ‘15), pp. 14-17, November 2015.

[30] M. Collotta, G. Pau, “A solution based on bluetooth low energy for smart home energy management,” Energies, vol. 8, no. 10, pp. 11916-11938, October 2015.

International Journal of Engineering and Technology Innovation, vol. 6, no. 1, 2016, pp. 79 - 91

Pollution Monitoring System Using Gas Sensor based on Wireless Sensor Network

M. Udin Harun Al Rasyid*, Isbat Uzzin Nadhori, Amang Sudarsono, Yodhista Tulus Alnovinda

Politeknik Elektronika Negeri Surabaya (PENS), Surabaya, Indonesia.

Received 03 November 2015; received in revised form 10 December 2015; accepted 15 December 2015

Abstract

Carbon monoxide (CO) and carbon dioxide (CO_2) gases are classified as colorless and odorless gas so we need special tools to monitor their concentration in the air. Concentration of air pollution of CO and CO_2 that are high in the air will give serious effects for health status. CO is a poisonous gas that damages the circulation of oxygen in the blood when inhaled, while CO_2 is one of the gases that causes global warming. In this paper, we developed an integrated pollution monitoring (IPOM) system to monitor the concentration of air pollution. This research implemented three sensor nodes (end-device) which each node contains CO and CO_2 sensors on the gas sensors board to perform sensing from the environment. Furthermore, the data taken from the environment by the sensor will be sent to the meshlium gateway using IEEE 802.15.4 Zigbee communications and processed by the gateway in order to be sent to the computer server. The data is stored in meshlium gateway using MySQL database as a backup, and it will be synchronized to the MySQL database in the computer server. We provide services for public to access the information in database server through a desktop and website application.

Keywords: CO and CO_2, gas sensor, zigbee, meshlium, wireless sensor network

1. Introduction

The air quality monitoring has become important issues to the living quality. Recently, with the fast growing industrial activities, the problem of air pollution is becoming a major concern of the people, and people's health has been seriously affected by the air pollution. The high concentration of Carbon monoxide (CO) and Carbon dioxide (CO2) in the air is very dangerous for human life. Red blood cells carry Oxygen (O2) in the air to the tissues of the body through the respiratory system. When the CO gas is absorbed by the red blood cells, the body will lack O2. The absorption of CO during a certain period may result in damage to the central nervous system and cardiovascular system. Symptoms include headache, drowsiness, lethargy, dizziness, nausea and fainting. The high concentrations of CO can cause heart rate, coma, heart failure, and damage to the respiratory function. In contrast to CO, CO2 does not directly impact the human body because it is not classified as toxic gas. The high concentration of CO2 in the air causes too much hot air trapped in the earth, thus the earth's temperature increases and the environment becomes hotter [1].

Wireless sensor network (WSN) is wireless network of small, low-cost sensors, which collect and send environmental data. WSN facilitates monitoring and controlling of physical environments from remote locations with good accuracy. WSN

* Corresponding author. E-mail address: udinharun@pens.ac.id

has applications in a variety of fields such as environmental monitoring, indoor climate control, surveillance, structural monitoring, medical diagnostics, disaster management, emergency response, ambient air monitoring, and gathering sensing information in inhospitable locations [2-3].

A number of researches have been conducted related to pollution monitoring. Yang et al. [4] have developed remote online monitoring system for geological carbon dioxide (CO2) leakage which consists of monitoring equipment, server, and clients. The system gets localization and time information through GPS, and then saved into SD cards storage module. The general packet radio service (GPRS) wireless transmission module will send the collected data wirelessly to the data centre server. Sivaraman et al. [5] developed HazeWatch project which uses several low-cost mobile sensor unit attached to vehicles to measure air pollution concentrations such as CO, nitrogen dioxide (NO2), and ozone (O3). The user can tag and upload the data in real time using their mobile phone. The spatial granularity of obtained data was collected to create of pollution maps which viewable in real-time over the web. The mobile personalized apps show the individual exposure history and route planning which less pollution. Peng et al. [6] implemented a total volatile organic compound (VOC) air pollution monitoring system to report value of temperature and humidity at indoor environment with the consideration of the cost, development complexity, and the operation convenience. Kadri et al. [7] developed an ambient real-time air quality monitoring system based on utilizing multi-gas (MG) monitoring stations that communicate with a platform by means of machine-to-machine (M2M) communications. Each MG station contains O3, CO, NO2, and H2S sensors. Jelicic et al. [8] developed flexible wireless system able to detect polluted air in a large environment. The system involves three levels: sensor level, node level and network level which contains metal oxide semiconductor (MOS) gas and a pyroelectric infrared (PIR) sensors. The system adopted duty-cycling of the gas sensor activity to extend the lifetime of node compared to the continuously driven gas sensor. Dian et al. [9] developed a CO2 sequestration monitoring and gas sensing by combining techniques of optical fiber sensing technology and wireless communication and analyzed the power consuming task such as sensing unit, processing unit, and transceiver unit. The result research recommended that renewable energy system such as solar cell maintain long-term and stable work. Pau et al. [10] developed a fuzzy logic system in wireless sensor network for environmental monitoring according to the battery level and to the throughput to reducing power consumption. The experiments result shows that fuzzy-based approach dynamically changes the sleeping time in order to reduce power consumption and increase the lifetime of sensor nodes.

In this paper, we implement CO and CO2 sensors on the gas sensors board to perform monitoring environment gas condition. The sensed data will be sent to MySQL database on meshlium gateway by using Zigbee wireless. The data in meshlium gateway as a backup will be synchronized to the MySQL database on backend computer server. We implement synchronization database between local database at meshlium and database at a server with the following information: local and external database names, local and external database sizes, local and external tables, total local and external entries, synchronized and unsynchronized local frames. Then, we provide desktop-based and web-based visualization to monitor environment condition by distant.

2. The System Design

This section will explain pollution-monitoring system, which is containing system architecture, component of system, sensor modules, configuring waspmote sensor node, and configuring meshlium gateway.

2.1 System Architecture

Currently, we are developing wireless sensor network for monitoring CO and CO_2 system as shown in Fig. 1. Gas sensor node is a device composed of CO sensor (TGS2442), CO_2 sensor (TGS4161), Waspmote Gases 2.0 board, and Waspmote PRO 1.2. A sensor node has to be connected to the battery as a power source and XBee module (IEEE 802.15.4 Zigbee protocol) as a means of wireless communication. Gas sensor nodes are placed in some areas that you want to sense and to gather data of CO and CO_2 concentration in the environment. There is a computer server that is connected to a gateway directly using Ethernet protocol. The gateway uses Meshlium from Libelium. Data from the sensor nodes are sent to a gateway using the IEEE 802.15.4 Zigbee protocol. Computer server is in charge of storing the data into the database and processing it by monitoring application.

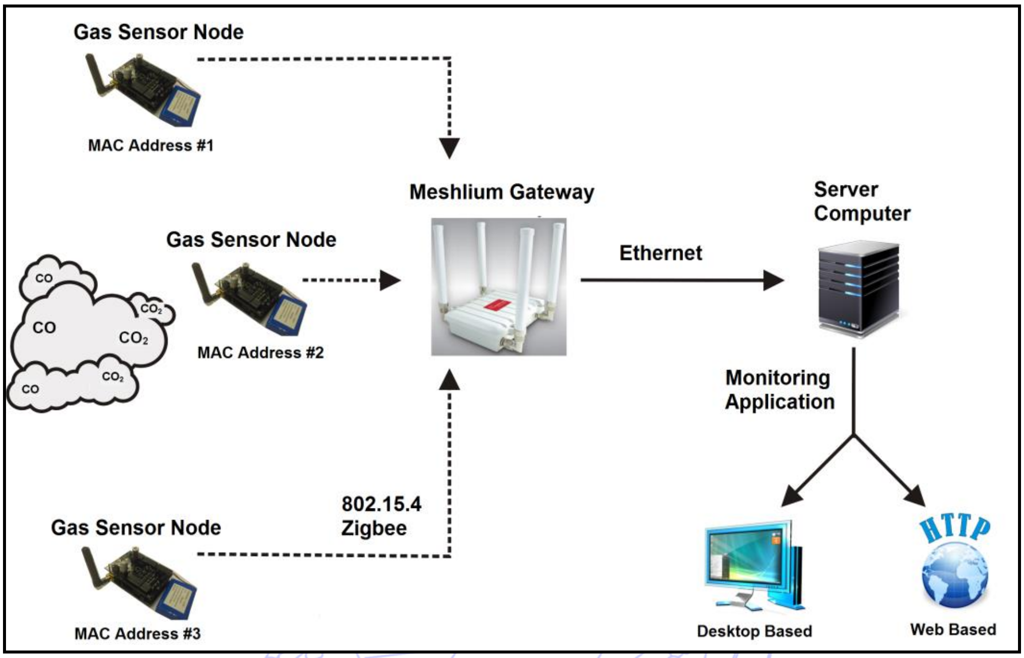


Fig. 1 System Architecture

Gas sensor node has a MAC address on the XBee module that distinguishes between one another. XBee module is in charge of sending the data wirelessly using IEEE 802.15.4 Zigbee protocol to the gateway that has been determined. Meshlium gateway serves as a receiver data obtained from the environment (via sensor nodes) to be processed by the computer server. Retrieval of data by sensor nodes performed every 60 seconds and immediately sent to the computer server through Meshlium gateway. Furthermore, the data received by the server computer will be saved to the database and can be processed.

To be able to communicate and transmit data wirelessly, a sensor node or end-device requires a Zigbee module. In addition, to be able to use the entire module is used (including Gas sensor), such devices require additional power from the battery.

2.2 Component of System

The specifications of the hardware used in this paper are shown on Table 1:

Table 1 The specifications of the hardware

1. Computer as server	a. CPU: 3.20 GHz (Intel Core i5) b. Memory: 4.0 GB RAM
2. Sensor node and Gateway	a. Waspmote PRO 1.2 b. Waspmote Gases 2.0 board c. CO sensor (TGS2442) d. CO_2 sensor (TGS4161) e. Xbee S1 module f.Meshlium Gateway

The installation contains sensor node, Meshlium gateway and switch for connecting to the local network. A server computer is connected to the switch directly using Ethernet. It's also done on Meshlium to make them have the same network.

2.3. Sensor Modules

We use TGS2442 sensor for sensing CO and TGS4161 sensor for sensing CO2. These sensors will be planted on Waspmote Gases 2.0 board connected to the microcontroller Waspmote PRO 1.2.

The TGS2442 is a resistive sensor sensitive to the changes in concentration of carbon monoxide (CO) and, very slightly, hydrogen (H2). It measures a value between 30 and 1000ppm. This sensor will be placed in socket 4 of Waspmote Gases 2.0 board. Reading this sensor requires a cycle of one second throughout which two power supply pulses are generated on heat resistance and sensor resistance of 14ms and 5ms each (average consumption throughout the power supply cycle is 3mA). The execution of this cycle and the reading of the sensor can be done automatically using the functions of the SensorGasv20 library.

The TGS4161 sensor provides a voltage output proportional to the CO2 concentration in the atmosphere. This sensor must be placed in socket 1A of Waspmote Gases 2.0 board. It shows a value between 220 and 490mV for a concentration of 350ppm (approximately the normal CO2 concentration in the air) decreasing as the amount of gas increases. Different sensors may show a large variability in the initial voltage values at 350ppm and sensitivity, so it is recommended to calibrate each sensor before including it in the application.

2.4. Configuring Waspmote Sensor Node

To perform the role of end-device, sensor nodes must be programmed beforehand. First of all, we set up the source code to activate the function of CO and CO_2 sensors. Source code can be opened using Waspmote IDE based C programming language. Fig. 2 shows source code to get data from the environment.

```
{ //begin sensor configuration
SensorGasv20.ON(); //Turn on the sensor board
RTC.ON(); //Turn on the RTC

//Configure the CO2 sensor socket
SensorGasv20.configureSensor(SENS_CO2, GAIN);
SensorGasv20.setSensorNode(SENS_ON, SENS_CO2);
//Configure the CO sensor socket
SensorGasv20.configureSensor(SENS_SOCKET4CO, GAIN, RESISTOR);
delay(5000);

} //end setup

void loop()
{
  coVal = SensorGasv20.readValue(SENS_SOCKET4CO);
  co2Val = SensorGasv20.readValue(SENS_CO2);
  //Convertion from CO voltage into kiloohms
  coVal = SensorGasv20.calculateResistance(SENS_SOCKET4CO, coVal, GAIN, RESISTOR);
  //Convertion from CO sensor resistance in kiloohms in PPM
  coValueInOOM = SensorGasv20.calculateConcentration(coCalibrationConcentration,coCalibrationOutput,coVal);
  //Convertion from CO2 sensor volts to PPM
  co2ValueInPPM = pow(10, (co2Val + 158.631)/62.877);
 }
```

Fig. 2 Get Data from Environment Code

Sensors will take gas concentration values of the environment in the form of electric voltage (volts) using readValue() function. Then, the value will be converted to part per million (ppm). For CO gas, the voltage value is converted to resistance using calculateResistance() function. After having obtained the resistance value, we convert that value into ppm using calculateConcentration() function. The value of CO2 sensor voltage can be converted into ppm directly using the formula pow(10, (co2Val + 158,631) / 62,877) [11], where co2Val is variable storing the voltage of CO2 sensor.

To be able to transmit data to Meshlium gateway, a Waspmote which acts as an end-device has to be programmed sending a frame to the Meshlium's MAC address. The following source code for delivering data that has been collected by sensor nodes to the gateway Meshlium as seen in Fig. 3.

We have to determine the type of frame that we use. There are two types of frame that can be used, ASCII and BINARY. ASCII frame structure is designed to make users read the frames sent conveniently. With ASCII frame, the payload written with ASCII characters is certainly more readable, while the BINARY frame structure made to compress the frame. The main goal of defining binary fields is to save bytes in frame's payload in order to send as much information as possible. Certainly with BINARY frame, it's not easy to read the data in the frame. We can also enter the ID Waspmote to give identifier for devices that we use. Frames can be made using frame.createFrame() function followed by the type of frame that we choose and Waspmote ID that we want. And then, we have to determine the data we send. We specify the sensor used and the variable stored the sensor data by using frame.addSensor() function. Furthermore, we need to set the packet carrying the frame. To be able to temporary stored the data carried by the frame, a packet memory should be allocated in advance by using packet = (packetXBee*) calloc(1, size of(packetXBee)). We also have to set the mode of packet transmission. There are two types of packet transmission mode, unicast and broadcast. We use unicast mode because the sensor node will send packet only to the Meshlium gateway directly. Unicast mode is used to send packets from a single node to one another node. The broadcast is used to send packets from a single node to the entire node in the same network. Lastly, we must determine where the packet containing the frame sent. We can use xbee802.setDestinationParams() function to determine the packet destination by entering packet's variable and destination MAC address. And then, the packet will be sent to the destination by using xbee802.sendXBee(packet).

```
//begin XBEE Communication
//1.Create ASCII frame

frame.createFrame(ASCII,"sensor");
frame.addSensor(SENSOR_CO, coValueInPPM);
frame.addSensor(SENSOR_CO2, co2ValueInPPM);
//frame.showFrame();

//2. SEND packet
//set parameters to packet:
packet=(packetXBee*) calloc(1,sizeof(packetXBee));                //Memory allocation
packet->mode=UNICAST   //Choose transmission mode: UNICAST or BROADCAST

//set destionation XBee parameters to packet
xbee802.setDestinationParams(packet, MAC_ADDRESS, frame.buffer, frame.length);
xbee802.sendXBee(packet); // Send Xbee packet
```

Fig. 3 Xbee Communication Code

2.5. Configuring Mehlium Gateway

Meshlium gateway is a Linux router that can contain five different radio interface: WiFi 2.4GHz, WiFi 5GHz, 3G/GPRS, Bluetooth, and ZigBee. Meshlium can also integrate GPS modules for mobile and vehicle applications, and it can have energy sources such as solar and battery system. These features are fitted with a protective aluminum IP65 allowing Meshlium to be placed outside the room. Meshlium is a complete Linux tool that offers a wide range of services, programming environment and storage systems. Meshlium has enabled HTTP / HTTPS and SSH in it. Meshlium also supports a wide variety of programming environments including C, C ++, Java, PHP, Python, Perl, and Ruby. Related information storage, Meshlium equipped with two different database systems, MySQL and Postgre. Meshlium has a local database that can hold the data that is sent to it until 29,5 GB, depending on the storage size. All information coming from all interfaces (ZigBee, Bluetooth, 3G/GPRS, WiFi and GPS module) can be stored in the Local File System Database or even exported to external database to the internet.

In this system, after the sensor node is able to get data from the environment and ready to send it, Meshlium also has to be set in advance to be able to receive the data from sensor node. Meshlium has user interface for system configuration called Manager System, which is similar to configuration application on router or gateway in general. By default, Manager System can be accessed using a web browser via Ethernet using 192.168.1.100 and via WiFi using 10.10.10.1. The initial view of Manager System will appear after we enter the username and password correctly. The main settings that must be done are PAN ID (Network ID) and Channel as shown in Fig. 4. PAN ID (Personal Area Network) is a network ID with a hexadecimal value between 0 and 0xFFFF. So there are 65536 possibilities that can be used. XBee can only communicate with one another if they have the same network ID. CH (Channel) sets the frequency band used by XBee to communicate with each other. XBee generally operates on the 2.4 GHz band. To be able to connect our system with local network, we have to setting the Meshlium Ethernet configuration. This configuration can be done in the Interfaces tab in the Manager System as shown in the Fig. 5.

The network topology as shown in Fig. 6. There is a switch connected directly to the Meshlium gateway using an Ethernet cable. The switch has an IP 10.252.13.1, and we will set the Meshlium IP to 10.252.13.225. There is a computer, acting as a server, which is connected directly to the switch using an Ethernet cable. The computer's IP address will set to 10.252.13.200. The server computer in charge of storing the external database synchronized with Meshlium local database.

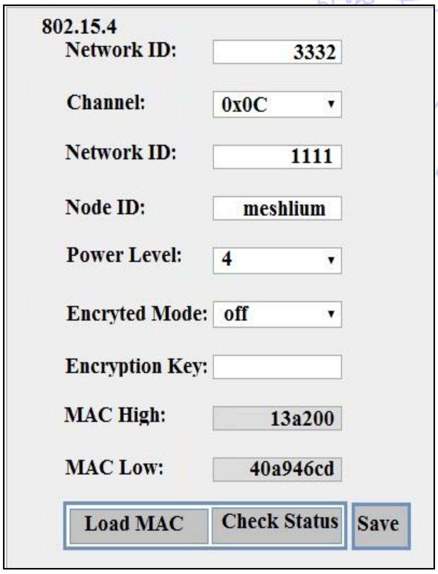


Fig. 4 Meshlium Zigbee Configuration

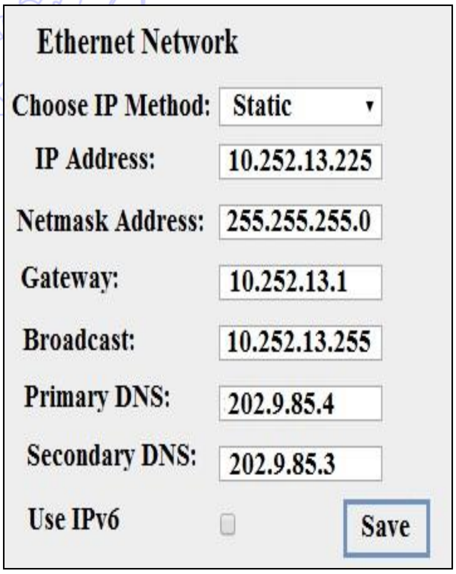


Fig. 5 Meshlium Ethernet Configuration

3. Experiment Result

In the experiment phase, we use three sensor nodes connected to the Meshlium gateway. Gateway is connected to a computer server via a local network using a switch. Network topology in the experiment is shown in Fig. 6.

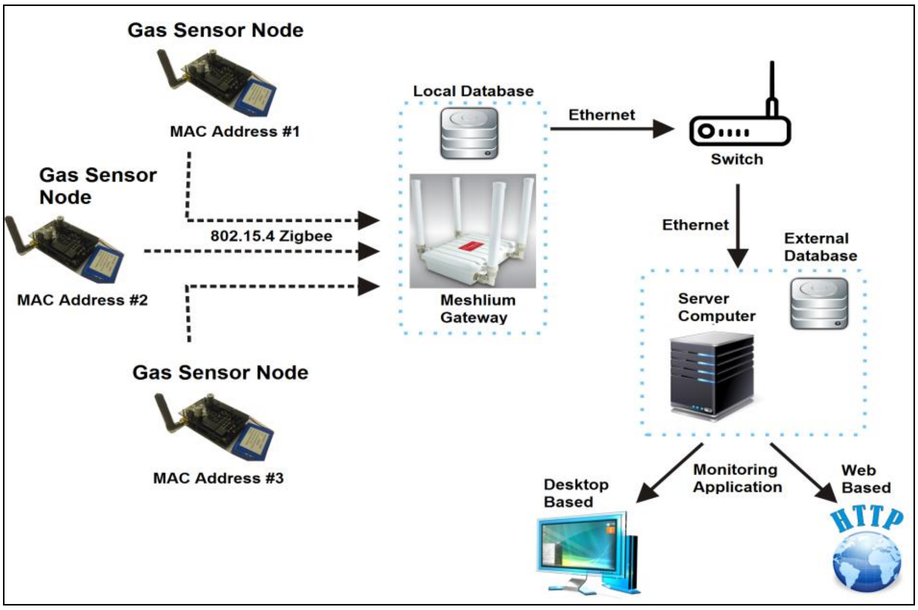


Fig. 6 Network Topology

Each sensor node has a MAC address that is different and represented by id_secret in the database. Meshlium gateway is a device with a Linux operating system. To know the contents of its Linux directory, we can access the Meshlium using an SSH connection. One of the most important components in Meshlium is sensor parser. This file is used to convert the frames sent by the sensor node and processed by the gateway. After the frame can be read, each value carried by the frame will be in inserted into Meshlium local database. Meshlium has MySQL local database to accommodate the values of the converted frame. Meshlium local database would be very vulnerable to damage and memory limitations of storage, so it is advisable to back-up the data to an external database. Meshlium has data synchronization system to external database directly at certain intervals. To synchronize, we need to set the IP address where the database created, the name of the database that will be used, username and password. In addition, the table structure used should adjust the structure of the tables in the local database. This is because the system synchronization follows the structure of the table on the local database Meshlium, while the structure of the table on the local database follows the insertion pattern of the sensor parser.

Furthermore, at the Sensor Network tab, there is Capturer tab in the side bar. In Capturer tab, we can see the contents of the frame sent by the sensor node. As we mentioned earlier, the sensor node transmits data in the form of a frame which is converted by Meshlium sensor parser. When we set the Local database to store all data received, we can see all data in the local database that have been received as shown in the Fig. 7.

Frame conversion result would be inserted into the table by following the structures below:

Id is the numbering of the records. Id is an integer auto-increment value with 11 digits maximum length.

Id_wasp is a string defined by the user which may identify each sensor node (Waspmote) inside the user's network. When the user do not want to give any identifier, the field remains empty indicated.

Id_secret is an id that distinguish device between one another. Id_secret represents the MAC address of each sensor node.

Frame_type is used to determine the frame type. There are two types of frame: Binary and ASCII.

Frame_number is a numbering for each delivery made by each device.

Sensor indicated type of sensor we use.

Value is the value read by the sensor node associated sensor used.

Timestamp shows the time when a frame is received by the gateway.

Raw is used to store a frame that failed to be converted by the sensor parser.

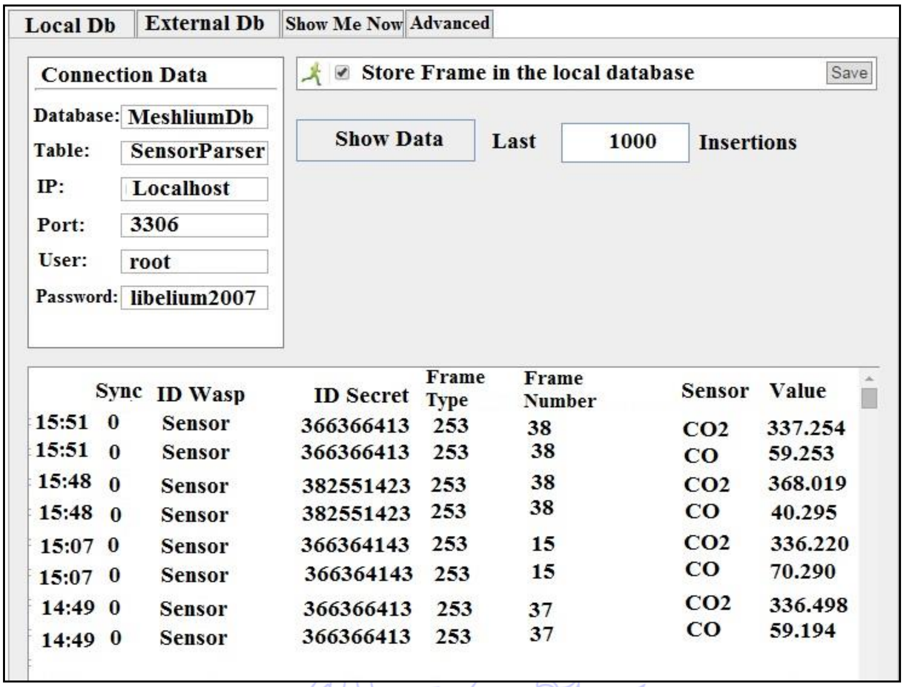


Fig. 7 Meshlium Local Database

Fig. 8 shows the synchronization of the local database to an external database. For synchronization, we use the IP address of computer server that has been connected to our system. We can also synchronize data automatically by specifying a particular interval. Synchronization can be performed automatically during Meshlium and server computers alive. Synchronization result can be viewed by pressing the Show Data.

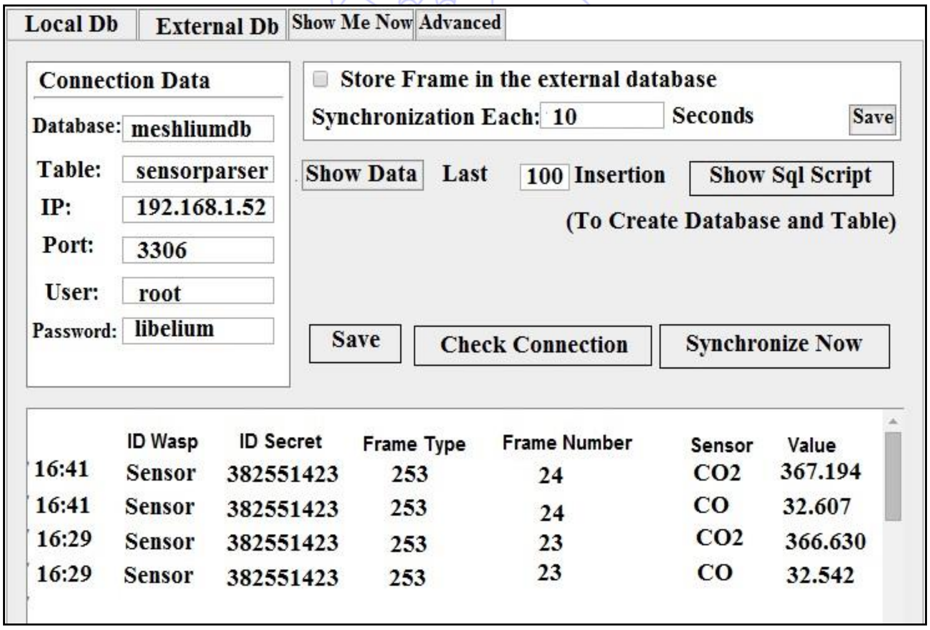


Fig. 8 External Database Synchronization

We can also see the results of the synchronization to the external database on the computer server. Fig. 9 shows the results of external database synchronization.

id	id wasp	Id_Secret	Frame_Type	Frame_Number	Sensor	Value	Timestamp	Raw
4187	eSN1	366366413	253	0	CO	25.591	2015-07-22 16:09:41	null
4188	eSN2	366366413	253	0	CO2	373.36	2015-07-22 16:09:41	null
4185	eSN2	382551423	253	118	CO	21.060	2015-07-22 15:58:46	null
4186	eSN2	382551423	253	118	CO2	365.79	2015-07-22 15:58:46	null
4183	eSN2	382551423	253	117	CO	21.070	2015-07-22 15:57:51	null
4184	eSN2	382551423	253	117	CO2	365.66	2015-07-22 15:57:51	null
4181	eSN2	382551423	253	116	CO	20.690	2015-07-22 15:56:55	null
4182	eSN2	382551423	253	116	CO2	366.07	2015-07-22 15:56:55	null
4179	eSN2	382551423	253	115	CO	20.417	2015-07-22 15:56:00	null
4180	eSN2	382551423	253	115	CO2	365.73	2015-07-22 15:56:00	null
4177	eSN2	382551423	253	114	CO	20.501	2015-07-22 15:55:04	null

Fig. 9 External Database

Our system has a desktop application to display the data synchronized. The application runs on the computer server and can only be accessed by authorized user. We can monitor the changes of value in real-time with a 5 minute interval. This interval is actually according to the interval data transmission by using desktop application. To display the data in real-time, we can open the Real-time tab as seen in Fig. 10. First, we have to choose the device that will be displayed, click Device drop down to do so. There will be three devices that can be selected. Then, to start reading its value, we can press the Sense button. Sense is a toggle button; this button will turn into Stop button (after you click Sense). It is used to stop the reading of the received value. The result can be seen in Fig. 10 below.

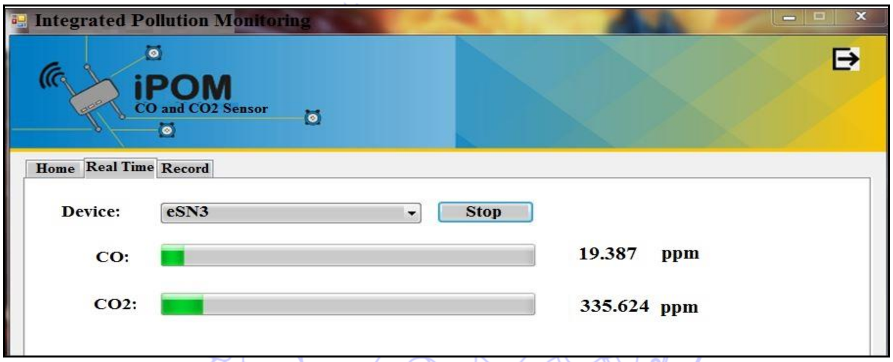
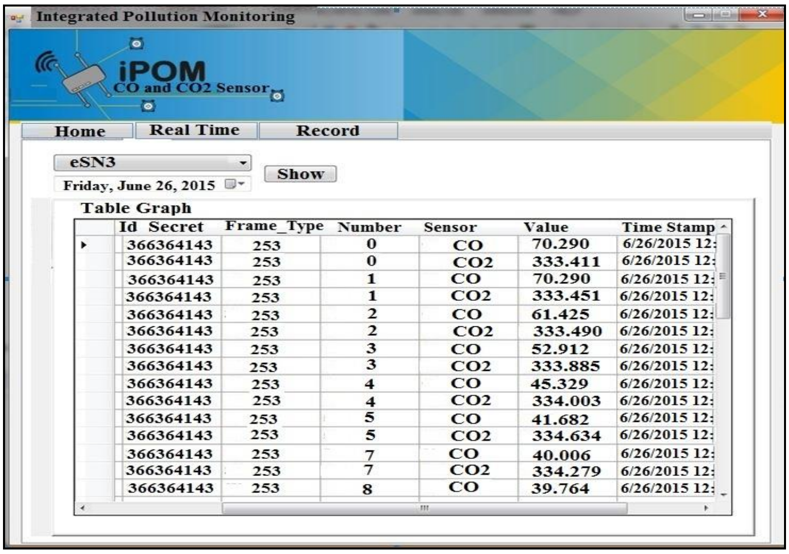


Fig. 10 Desktop Application Real-time Feature

We also can see data from a certain date through the Record tab. In Record tab, there are two sidebars: Graph and Table sidebar as shown in Fig. 11. Once we select the device and the date, click the Show button to display the data according to the device and the date we set. Then the data will be taken from external database and then displayed in graphs and tables. The field shown in Table sidebar is all fields in the database, so basically we display the information of frame stored. We design this feature for users who use desktop application, so they can perform further data analysis.



Id Secret	Frame_Type	Number	Sensor	Value	Time Stamp
366364143	253	0	CO	70.290	6/26/2015 12:
366364143	253	0	CO2	333.411	6/26/2015 12:
366364143	253	1	CO	70.290	6/26/2015 12:
366364143	253	1	CO2	333.451	6/26/2015 12:
366364143	253	2	CO	61.425	6/26/2015 12:
366364143	253	2	CO2	333.490	6/26/2015 12:
366364143	253	3	CO	52.912	6/26/2015 12:
366364143	253	3	CO2	333.885	6/26/2015 12:
366364143	253	4	CO	45.329	6/26/2015 12:
366364143	253	4	CO2	334.003	6/26/2015 12:
366364143	253	5	CO	41.682	6/26/2015 12:
366364143	253	5	CO2	334.634	6/26/2015 12:
366364143	253	7	CO	40.006	6/26/2015 12:
366364143	253	7	CO2	334.279	6/26/2015 12:
366364143	253	8	CO	39.764	6/26/2015 12:

Fig. 11 Desktop Application Record Table

We show the data on the table while displaying the graphics so that the user can easily observing. Graphic display, as shown in Fig. 12, can be viewed by moving to Graph sidebar.

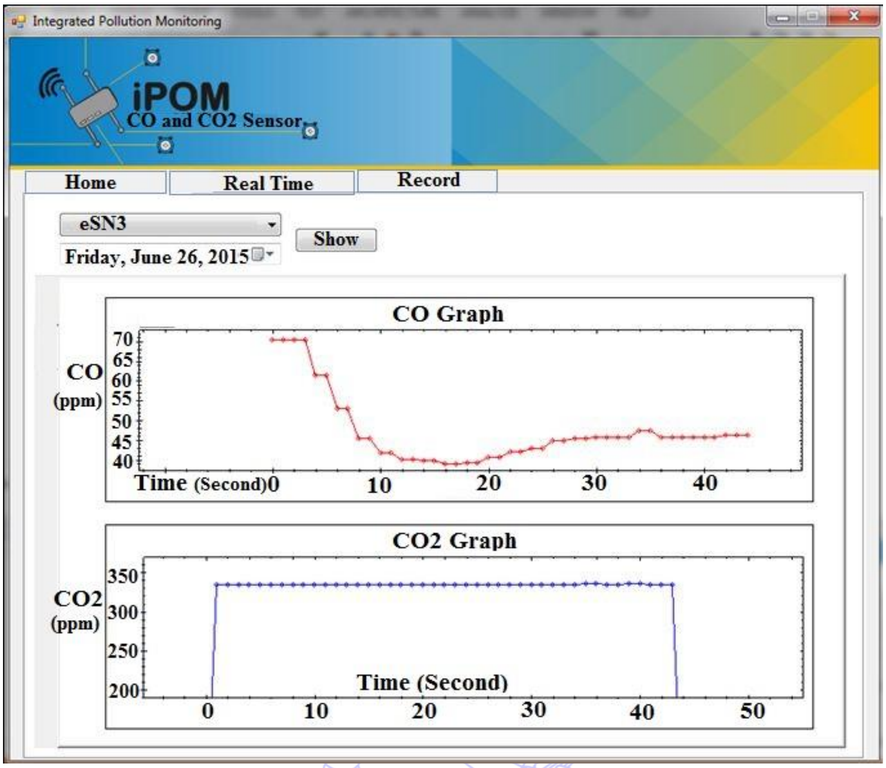


Fig. 12 Desktop Application Record Graph

For public accesses, our system provides a website for displaying the stored data. When user accesses the website, its Dashboard will be shown. We can see the results of the data readout from the environment in graph and table. As seen in Fig. 13, when we choose Summary Chart tab on the sidebar, the page to display the data in graphs will appear. On this page, we can display a summary of CO and CO_2 values within a certain time frame. First of all, we need to select the device which we want to see. We can see a summary value of CO and CO_2 concentration for today, this month, or this year, by clicking the button according to the desired time frame.

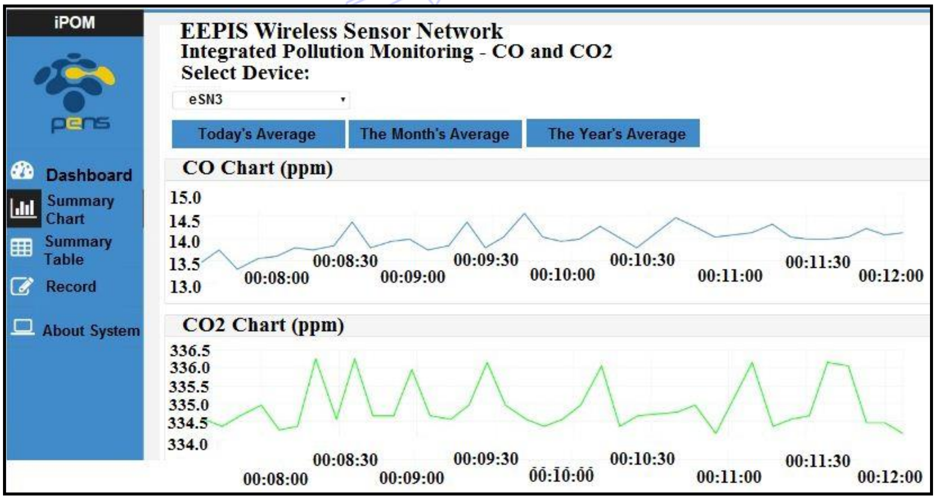


Fig. 13 Website Summary Chart

To view a summary of the value in table, we can open the Summary Table in the sidebar tab. The step that must be done is almost the same as displaying data in the graphic. We need to select the device which we want to see. We can see a summary value of CO and CO_2 concentration for today, this month, or this year, by clicking the button according to the desired time frame as shown in Fig. 14.

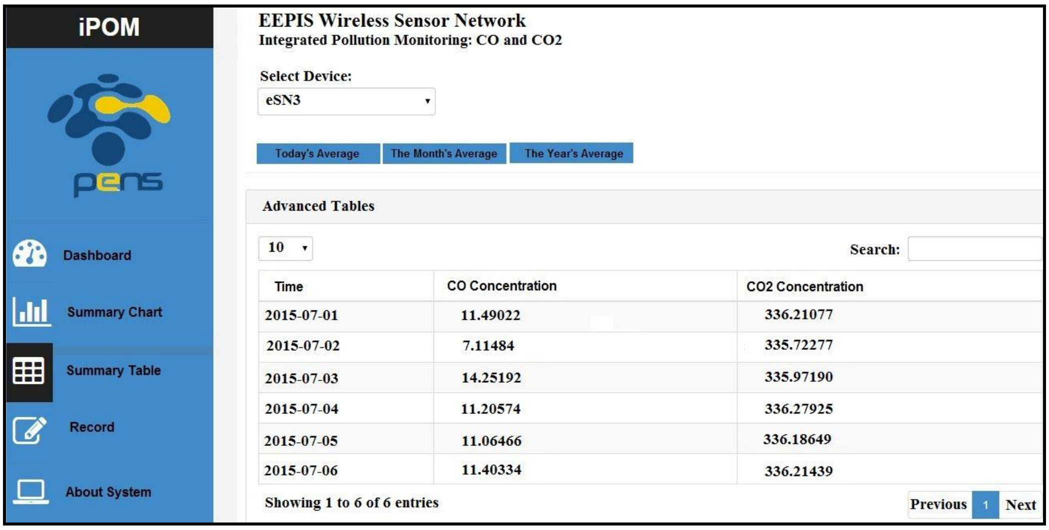


Fig. 14 Website Summary Table

In addition, we also can see a graph of data recorded according to a certain date through the Record tab. When we choose the tab, the page to see the recorded values in a certain day will appear. Just like before, we need to choose the device that we want to see. We can also choose All Devices to see the value of the entire devices being used. Then we also need to choose a specific date. The result is shown in Fig. 15 (one device selected) and Fig. 16 (all devices selected).

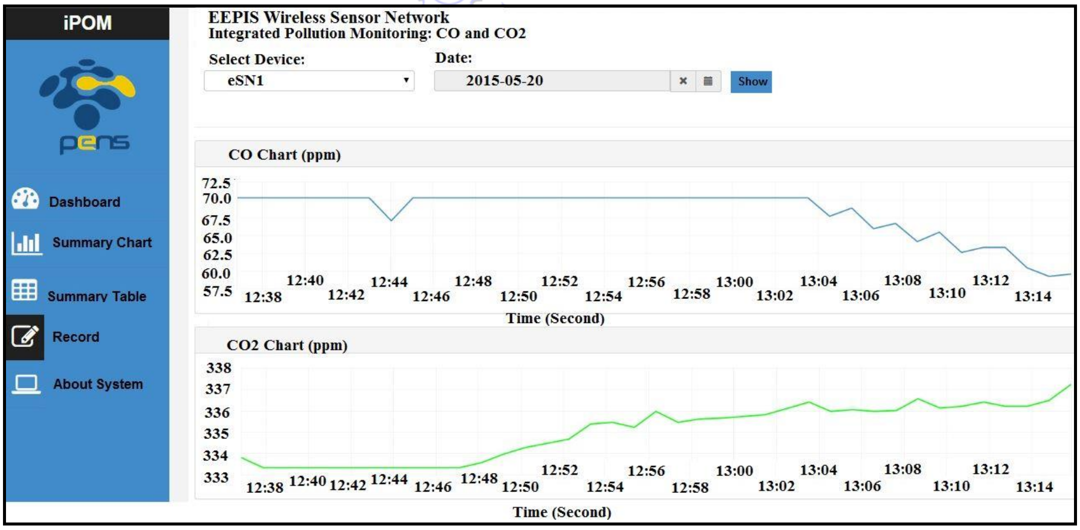


Fig. 15 Website Record One Device Selected

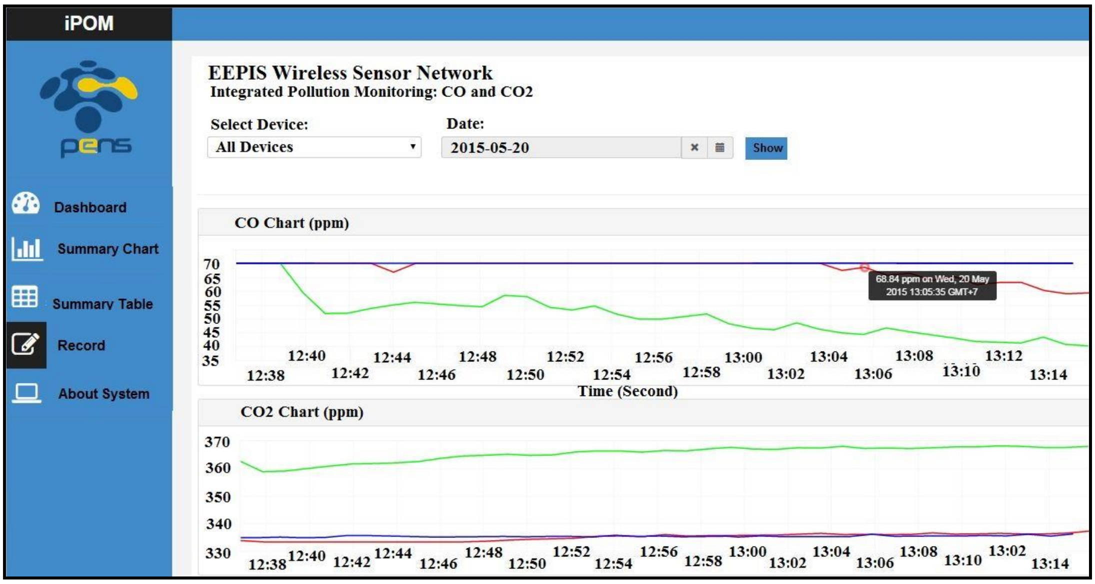


Fig. 16 Website Record All Devices Selected

Based on the above results from IPOM design and related works, we have considered six parameters to show the contribution of IPOM design to implement gas monitoring wireless sensor network such as sensor type, wireless transmission module, environment (indoor or outdoor), microcontroller, gateway, and backup service (i.e., database backup and synchronization) as shown on Table 2.

Table 2 Comparison between design of IPOM and related works

Model	**Sensor**	**Wireless Transmission**	**Environment**	**Microcontroller**	**Hardware (Gateway)**	**Backup Service**
Yang [4]	CO_2	GPRS	Outdoor	*ARM Cortex-M3*	No Gateway	No
Sivaraman [5]	CO_2, NO_2, O_3	GPRS	Outdoor	NA	No Gateway	No
Peng [6]	Temperatur, Humidity	ZigBee	Indoor	Arduino Atmel 8-bit AVR	Arduino Atmel 8-bit AVR	No
Kadri [7]	O_3, CO, NO_2, H_2S	GPRS/3G	Outdoor	ATmega2560	No Gateway	No
Jelicic [8]	Metal Oxide Semiconductor (MOS), Pyroelectric InfraRed (PIR)	ZigBee	Indoor	Jennic JN5148	No Gateway	No
Dian [9]	CO_2	WLAN	Indoor	NA	No Gateway	No
Proposed Model	CO, CO_2	ZigBee	Outdoor	ATmega1281	Meshlium	Yes

4. Conclusions

This paper explains development of an integrated pollution monitoring (IPOM) system to monitor the concentration of air pollution using gas sensor and Meshlium gateway. We analyze the data received from sensor nodes to server receiver can be converted by Meshlium using its sensor parser. We can join our local area network and our wireless sensor network. By joining our network, we can use a server computer to manage the data sent by sensor nodes. Meshlium also has synchronization system to synchronize its local database and our external database in server computer. The experiment results show that the sensed data from environment can be displayed through wireless sensor network in desktop application and website to monitor the environment condition from distant. For the future work, we will add more meshlium gateway and sensor node to increase the coverage area, and use WiFi/3G/GPRS to increase scale of pollution monitoring system.

Acknowledgement

This research was supported by Ministry of Research, Technology and Higher Education of Indonesia, INSINAS Scheme.

References

[1] Y. Xiang, R. Piedrahita, R. P. Dick, M. Hannigan, Qin Lv, and L. Shang, "A hybrid sensor system for indoor air quality monitoring," Proc. IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS), IEEE Press, May 2013, pp. 96-104.

[2] B. H. Lee, M. U. H. Al Rasyid, and H. K. Wu, "Analysis of superframe adjustment and beacon transmission for IEEE 802.15.4 cluster tree networks, " EURASIP Journal on Wireless Communications and Networking, vol. 2012, July 2012.

[3] M. U. H. Al Rasyid, B. H. Lee, A. Sudarsono, and Taufiqurrahman, "Implementation of body temperature and pulseoximeter sensors for wireless body area network," Sensors and Materials, vol. 27, pp. 727–732, September 2015.

[4] H. Yang, Y. Qin, G. Feng, and H. Ci, "Online monitoring of geological CO2 storage and leakage based on wireless sensor networks," IEEE Sensors Journal, vol. 13, pp. 556-562, October 2012.

[5] V. Sivaraman, J. Carrapetta, K. Hu, and B. G. Luxan, "HazeWatch: A participatory sensor system for monitoring air pollution in Sydney," Proc. IEEE 38th Conference on Local Computer Networks Workshops (LCN Workshops), IEEE Press, Oct. 2013, pp. 56-64.

[6] I. H. Peng, Y. Y. Chu, C. Y. Kong, and Y. S. Su, "Implementation of indoor VOC air pollution monitoring system with sensor network," Proc. Seventh International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS), IEEE Press, July 2013, pp. 639-643.

[7] A. Kadri, E. Yaacoub, M. Mushtaha, and A. A. Dayya, "Wireless sensor network for real-time air pollution monitoring," Proc. International Conference on Communications, Signal Processing, and their Applications (ICCSPA), Feb. 2013.

[8] V. Jelicic, M. Magno, G. Paci, D. Brunelli, and L. Benini, "Design, characterization and management of a wireless sensor network for smart gas monitoring," Proc. 4th IEEE International Workshop on Advances in Sensors and Interfaces (IWASI), IEEE Press, June 2011, pp. 115-120.

[9] F. Dian, "Development of novel gas detection wireless sensor node," Proc. Spring Congress on Engineering and Technology (S-CET), May 2012, pp. 1-3.

[10] G. Pau, "Power consumption reduction for wireless sensor networks using a fuzzy approach," International Journal of Engineering and Technology Innovation (IJETI), vol. 6, no. 1, pp. 55-67, January 2016.

[11] "GitHub Repository Arduino Code," https://github.com/fablabbcn/ASK-Shield/blob/master/ArduinoCode%20-%20ASK%20Shield/SensorTest/SensorsTest_ASK/co2.ino, September 11, 2015.