

LIFE TIME SENSITIVE WEIGHTED CLUSTERING ON WIRELESS SENSOR
NETWORKS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF INFORMATICS
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

Elnaz Alizadeh Jarchlo

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
THE DEPARTMENT OF INFORMATION SYSTEMS

SEPTEMBER 2013

Approval of the thesis:

**LIFE TIME SENSITIVE WEIGHTED CLUSTERING ON WIRELESS SENSOR
NETWORKS**

Submitted by **Elnaz Alizadeh Jarchlo** in partial fulfillment of the requirements for the degree of **Master of Science in Information Systems, Middle East Technical University** by,

Prof.Dr. Nazife Baykal -----

Director, **Informatics Institute, METU**

Prof.Dr. Yasemin Yardımcı Çetin -----

Head of department, **Information Systems, METU**

Assoc.Prof. Dr. Cüneyt F. Bazlamaçcı -----

Supervisor, **Electrical and Electronics Engineering, METU**

Examining Committee Members:

Prof.Dr. Nazife Baykal -----

Information Systems, Inf. Inst., METU

Assoc.Prof.Dr. Cüneyt F. Bazlamaçcı -----

Electrical and Electronics Engineering Dept., METU

Assoc.Prof.Dr. Altan Koçyiğit -----

Information Systems, Inf. Inst., METU

Asst.Prof.Dr. Erhan Eren -----

Information Systems, Inf. Inst., METU

Prof.Dr. Semih Bilgen -----

Electrical and Electronics Engineering Dept., METU

Date: -----

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name: Elnaz Alizadeh Jarchlo

Signature:

iv

ABSTRACT

LIFE TIME SENSITIVE WEIGHTED CLUSTERING ON WIRELESS SENSOR NETWORKS

Elnaz Alizadeh Jarchlo

M.Sc., Department of Information Systems

Supervisor: Assoc.Prof.Dr. Cüneyt F. Bazlamaçcı

September 2013,52 pages

Wireless Sensor Networks typically include wireless sensor nodes with limited energy. Network lifetime and scalability are considered as two significant requirements for sensor network applications. In order to decrease energy consumption and increase network lifetime one can apply an efficient clustering method. The application of the clustering method proposed in this thesis (LTS-WCA) leads to reducing the energy cost and the transmission distance of each node by grouping the nodes in several clusters and assigning specific cluster heads for each group. This study aims to compare existing clustering algorithms and make modifications to enhance their performance. The study includes the implementation of LTS-WCA, which is proposed within the scope of this work, in a simulation environment and its evaluation in order to demonstrate its effectiveness.

Keywords: Wireless sensor networks, wireless sensor nodes, network lifetime, clustering, clusterhead.

ÖZ

KABLOSUZ DUYARGA AĞLARINDA YAŞAM SÜRESİNE DUYARLI AĞIRLIKLI KÜMELEME

Elnaz Alizadeh Jarchlo

M.Sc, Bilişim Sistemleri

Danışman: Doç.Dr. Cüneyt F. Bazlamaçcı

Eylül 2013, 52sayfa

Kablosuz duyarga ağları genel olarak sınırlı enerji kaynakları olan kablosuz algılayıcı düğümlerinden oluşmaktadır. Ağ yaşam süresi ve ağ ölçeklenebilirliği, duyarga ağ uygulamaları için çok önemli iki gereksinim olarak nitelendirilmektedir. Enerji tüketimini düşürmek ve ağ yaşam süresini artırmak için verimli bir kümeleme yöntemi kullanılabilir. Bu tez çalışmasında önerilen kümeleme yönteminin uygulanması, düğümlerin bir kaç kümede gruplanması ve her grup için belirli bir küme başının tespit edilmesiyle her bir düğümün iletim uzaklığının düşmesini sağlar. Bu tez çalışmasının amacı, duyarga ağları için var olan kümeleme algoritmalarının karşılaştırılması ve yapılacak değişikliklerle başarımlarının iyileştirilmesidir. Çalışma, tez kapsamında önerilen kümeleme algoritması LTS-WCA'nın bir benzetim ortamında gerçekleşmesi ve etkinliğinin değerlendirilmesini de içermektedir.

Keywords: Kablosuz duyarga ağları, kablosuz duyarga düğümü, ağ yaşam süresi, kümeleme, küme başı

To my family and all my friends
who supported me at the very stage of my work.
Thank you for your belief in me...

ACKNOWLEDGEMENTS

First of all, I would like to express my greatest appreciation to my thesis supervisor, Assoc.Prof. Dr. Cüneyt F. Bazlamaçcı, for his sincere help, genuine guidance, and strong everlasting encouragement throughout my thesis work.

I would like to thank Assistant Prof. Mubashir Husain Rehmani from the department of Electrical Engineering in COMSATS Institute of Information Technology (Wah Cantt, Pakistan) for his thoughtful help throughout my studies as well as the writing of the present thesis.

My sincere regards are conveyed to my closest friends, Alena Iriskulova, Rozita Kian, Saiedeh Nazirzadeh and Hamed Khosravi, who supported me at every stage of my work.

Finally, I wish to thank my parents for their support and encouragement throughout my study, because but this acknowledgement would be incomplete if I did not convey my deepest gratitude to my loving mother, father, and brother. Thank you for being so understanding and patient. Without your strong support this thesis would never be possible.

Thank you all...

TABLE OF CONTENTS

ABSTRACT.....	v
ÖZ.....	vi
ACKNOWLEDGEMENTS.....	viii
LIST OF ABBREVIATIONS.....	xiii
CHAPTER 1.....	1
INTRODUCTION.....	1
1.1. Overview of the Wireless Sensor Networks and Clustering Algorithms.....	1
1.2. Problem Definition.....	1
CHAPTER 2.....	5
OVERVIEW OF EXISTING WEIGHTED CLUSTERING ALGORITHMS.....	5
2.1 Mobile Adhoc Networks vs Wireless Sensor Networks.....	5
2.2 Weighted Clustering Algorithms for MANET.....	6
2.3. On-Demand Weighted Clustering Algorithm (WCA) for Ad Hoc Networks.....	8
2.4. Weight Based Adaptive Clustering in Wireless Ad Hoc Networks (WBACA).....	9
2.5. Weighted Clustering Algorithm Using Local Cluster-heads Election for QoS in MANETs.....	1
2.6. Entropy-Based Weighted Clustering Algorithm and Its Optimization for Ad Hoc Networks.....	12
2.7. Advanced Efficiency and Stability Combined Weight Based Distributed Clustering Algorithm in MANET.....	12
2.8. Weight Based Adaptive Clustering for Large Scale Heterogeneous MANET.....	13
2.9. Dynamic Energy Efficient Clustering Algorithm for MANETs.....	14
2.10. Improved Weight-Based Clustering Algorithm in MANETs.....	14
2.11. Energy Efficient and Stable Weight Based Clustering for Mobile Ad hoc Networks	15
2.12. Flexible Weighted Clustering Algorithm Based on Battery Power for Mobile Ad Hoc Networks.....	16
2.13. Enhanced Weighted Clustering Algorithm for Mobile Networks.....	16
2.14. Maximal Weight Topology Discovery in Ad hoc Wireless Sensor Networks.....	17

2.15. Efficient Cluster-head Election Algorithm Based on Maximum Weight for MANET.....	18
2.16. Comparison of Weighted Clustering Algorithms for MANET in terms of Their Objectives and Weight Functions.....	18
2.17. Weighted Clustering Algorithm for WSN.....	22
2.18. Clustering Algorithm for Localization in Wireless Sensor Networks.....	22
2.19. Improved Weighted Clustering Algorithm for Determination of Application Nodes in Heterogeneous Sensor Networks.....	23
2.20. Distributed Energy-Efficient Hierarchical Clustering for Wireless Sensor Networks.....	24
2.21. Comparison of Weighted Clustering Algorithms for WSN in Terms of Their Objectives and the Weight Functions.....	24
CHAPTER 3.....	27
LIFE TIME SENSITIVE WEIGHTED CLUSTERING ALGORITHM.....	27
3.1. Differences with the Original WCA.....	27
3.2. Assumptions on the Implementation of the Proposed Algorithm.....	28
3.3. The Proposed Algorithm – Life-Time Sensitive Weighted Clustering Algorithm (LTS-WCA).....	29
3.4. Enhancements parameters proposed on LTS-WCA.....	32
CHAPTER 4.....	35
SIMULATION WORK AND EVALUATION RESULTS.....	35
4.1. Nodes categorizations.....	35
4.2. Mobility Model.....	35
4.3. Energy Model.....	36
4.4. Traffic Model.....	37
4.5. Implementation Phase of LTS-WCA.....	37
4.6. Simulation Scenario.....	39
4.7. Implementation of LTS-WCA.....	39
CHAPTER 5.....	47
CONCLUSIONS.....	47
REFERENCES.....	49

LIST OF TABLES

Table 1: Weighted Clustering Algorithms for MANET	7
Table 2: Comparison of weighted clustering algorithms for MANET in terms of their objectives and weight functions.....	19
Table 3: weight combinations of weighted clustering algorithms	21
Table 4: Weighted Clustering Algorithms for WSN	22
Table 5: weighted clustering algorithm for WSN.....	25
Table 6: the ‘combined weight’ calculations specifically for each algorithm	25
Table 7: an adjacent table	31

LIST OF FIGURES

Figure 1: A Weighted Clustering Algorithm Using Local Cluster-heads Election for QoS in MANETs	11
Figure 2: Nch vs Nn	41
Figure 3: Tr vs Nch.....	41
Figure 4: Nc vs Tr.....	42
Figure 5: Time for clustering vs Nn	42
Figure 6: LT vs Nn	44
Figure 7: LT vs En.....	44
Figure 8: LT vs Tr	45
Figure 9: LT vs Nn	45

LIST OF ABBREVIATIONS

CBR	Constant Bit Rate
CFL	A Clustering Algorithm For Localization in Wireless Sensor Networks
CH	Cluster Head
CWBDCA	Advanced Efficiency and Stability Combined Weight Based Distributed Clustering Algorithm in MANET
DEECA	A Dynamic Energy Efficient Clustering Algorithm for MANETs
DWEHC	Distributed Energy-Efficient Hierarchical Clustering for Wireless Sensor Networks
EBWCA	An Entropy-Based Weighted Clustering Algorithm and Its Optimization for Ad Hoc Networks
ECAM	Energy Efficient clustering algorithm
EE-SWBC	Energy Efficient and Stable Weight Based Clustering for Mobile Ad hoc Networks
EWCAA	Novel Enhanced Weighted Clustering Algorithm for Mobile Networks
FWCABP	A Flexible Weighted Clustering Algorithm Based on Battery Power for Mobile Ad Hoc Networks
HP	Hop Count
ID	Identification Number
IWCA(1)	An Improved Weight-Based Clustering Algorithm in MANETs

IWCA(2)	An Improved Weighted Clustering Algorithm for Determination of Application Nodes in Heterogeneous Sensor Networks
LTS-WCA	Life-Time Sensitive Weighted Clustering Algorithm
MANET	Mobile Adhoc Network
Max weight	Maximal Weight Topology Discovery in Ad hoc Wireless Sensor Networks
TR	Transmission Range
WACHM	Weight Based Adaptive Clustering for Large Scale Heterogeneous MANET
WBACA	Weight Based Adaptive Clustering in Wireless Ad Hoc Networks
WCA	Weighted Clustering Algorithm
WCA-L	A Weighted Clustering Algorithm Using Local Cluster-heads Election for QoS in MANETs
WSN	Wireless Sensor Network

CHAPTER 1

INTRODUCTION

1.1. Overview of the Wireless Sensor Networks and Clustering Algorithms

Sensor networks include a large number of sensors which are able to sense the environment and process the data in order to transfer the gathered information through the sink. Wireless sensor nodes play a critical role in humans' lives, therefore many researchers have attempted to utilize the existing theoretical studies in order to apply them on practical applications in various areas such as for medical purposes (e.g. health monitoring [1]), environmental purposes (e.g., disaster monitoring), military purposes (e.g., target surveillance and battlefield mapping [2]) and agricultural purposes [3], etc.

Clustering a wireless sensor network leads to a hierarchical network by partitioning the existing flat network into several groups of nodes where each group has a leader node called "cluster-head" and the remaining local nodes, which are connected to their associated cluster-heads, are called "cluster members".

The main motivation behind running a clustering algorithm on a wireless sensor network is transmitting aggregated data to the sink, reducing number of nodes taking part in transmission, saving energy consumption, scaling for large number of nodes and reducing communication overhead.

The cluster-heads then are usually connected in the form of a backbone of the corresponding wireless sensor network.

1.2. Problem Definition

One of the main resource restrictions in WSN is the energy available in the nodes. Therefore, it is important to modify weighted clustering algorithms in a more energy efficient manner to help increase network life time. The present thesis work proposes certain

modifications on an original weighted clustering algorithm in order to reach the aforementioned WSN-related purpose. Our approach includes three different stages of the WCA work including the application of the clustering algorithm on the network, network operation and online cluster maintenance. Each of these stages is explained separately in the following chapters.

There are various characteristics of WSNs, such as mobility and heterogeneity of nodes, large scalability requirement and power restrictions, which are all considered in our model. It is expected that during the lifetime of a sensor network, the network may face several problems, such as: (1) lack of energy in sensor nodes that fail in their attempt to cover the environment properly; (2) overloaded cluster-heads; (3) existence of some isolated sensor nodes in the network; (4) fast energy depletion rate used for both transmission and processing purposes.

Cluster heads play a critical role in wireless sensor networks. Their energy consumption is considerably higher than that of the ordinary sensing nodes in the network. Therefore, the selection of cluster heads in a wireless sensor network environment is a significant issue for efficiency, which, in its turn, is the main problem considered within the scope of the present thesis. This thesis work tries to contribute to the solution of this problem by modifying the objective function of the clustering algorithm and considering other extra parameters that are believed to be effective.

Cluster heads perform both intra- and inter-cluster communication and therefore consume larger amounts of energy compared to ordinary nodes. Sensor nodes are generally assumed to be cheap, with the limited amount of non-renewable energy, therefore controlling their energy consumption is considered to be a vital issue. This is necessary for keeping the nodes alive as long as possible in order to cover the sensing environment properly. The present work considers several solutions for overcoming the aforementioned problems by modifying the original WCA. Multi-hop routing is used within and between clusters. Gateway nodes are also employed to help cluster heads communicate with considerably less amount of energy regarding intra- and inter-cluster communications.

Several problems, such as the presence of energy holes and existence of some isolated sensor nodes in the network, affect the network coverage. The present thesis study considers additional helpful parameters within clustering algorithm such as hop count (HP) and

transmission range (TR). It defines several thresholds for the size of clusters, the residual energy of nodes, and the number of hops within a cluster in order to manage the coverage issue in the network. Although in cluster maintenance part there exists several considerations due to node mobility, it is assumed that the network is adjusted in an efficient way to keep the network coverage level as high as possible.

The present thesis aims at modifying and enhancing the weighted clustering algorithm (WCA), which was frequently cited in other works of the field for the purpose of forming cluster-heads in mobile ad hoc networks. The main argument for its use is its effectiveness when used in a wireless sensor network domain.

The thesis concentrates on the issue of energy efficiency in clustering Wireless Sensor Networks (WSN) and especially on increasing the network life time by considering several critical parameters in this new algorithm called as the Life-Time Sensitive Weighted Clustering Algorithm (LTS-WCA).

The first two chapters of the thesis describe two different networks, namely Mobile Adhoc Networks and Wireless Sensor Networks and the present Weighted Clustering Algorithms proposed for these. Chapter 3 presents our Life-Time Sensitive Clustering Algorithm (LTS-WCA) and the differences between it and the original Weighted Clustering Algorithm. Chapter 4 includes the simulation study carried out on LTS-WCA using NS2 simulator and the results and evaluation of LTS-WCA. Finally, Chapter 5 concludes the thesis.

CHAPTER 2

OVERVIEW OF EXISTING WEIGHTED CLUSTERING ALGORITHMS

This chapter presents mobile ad hoc networks as opposed to wireless sensor networks. It concentrates on various weighted clustering algorithms discussing their features and their associated weight formulas. Special emphasis is put on the issues of life time and energy efficiency as these are the main areas of interest in the present research work.

2.1 Mobile Adhoc Networks vs Wireless Sensor Networks

One of the main issues in a clustering algorithm is the process of categorizing the nodes into disjoint groups and choosing the most appropriate nodes as cluster-heads for having an efficient network. There are different types of clustering algorithms running on both Wireless Sensor Networks (WSN) and Mobile Adhoc Networks (MANET). However, there are similarities and differences between these two types of networks.

The main similarities between MANET and WSN are described as follows:

- Both are distributed networks without any central infrastructure.
- Communication and routing among nodes may be in multi hop fashion.
- Both use wireless links for communication purposes.
- Both are self-organized due to their distributed nature of networks.

The main differences between MANET and WSN can be listed as follows [21]:

- MANET users are generally appliances designed for human beings (e.g., laptop, computers, mobile radio terminals, etc.) but WSN focuses mostly on interactions with the environment (monitoring and sensing the environment or maybe tracking objects and/or activities in this sensing environment).

- MANET has less network density in comparison to WSN and the number of sensor nodes in WSN is higher.
- The number of active users in the deployment area specifies the network size in MANET, however the extension of the observed area, characteristics of the nodes and the required redundancy level define the number of nodes in WSN.
- Data traffic in MANET is higher than WSN because of using services like web, mail, video, etc.
- Sensor nodes in MANET are computation devices, whereas in WSN they are cheap tiny nodes.
- Energy issues in MANET have less importance in comparison to WSN. Stored energy in tiny sensor nodes in WSN is very restricted and the quick loss of energy is considered as a main problem there.

The purposes of running clustering algorithms on these two types of networks are different. Wireless sensor networks mostly concentrate on how to sense the environment and send the gathered data to the sink aiming at the increase of the network life time and the energy efficiency but in mobile adhoc networks the main goal is to distribute the generated traffic among nodes by routing it in a multi hop fashion.

Weighted Clustering Algorithm (WCA) is one clustering algorithms, which was originally and initially applied on mobile adhoc networks[5]. It considers several different parameters in order to select a cluster-head (CH), which is believed to be more beneficial compared to other clustering algorithms that take one or two parameters into account at most. The main objectives of applying WCA on MANET are choosing an optimal number of CHs, decreasing latency as much as possible and decreasing the number of re-affiliations. Since there exist quite a large number of similarities between MANET and WSN, the application of WCA may be helpful and beneficial for WSN as well. However, it surely needs certain modifications in order to be adapted to wireless sensor network environments.

2.2 Weighted Clustering Algorithms for MANET

Table 1 below shows that there is a vast amount of research work, which followed WCA and each of these weighted clustering studies has its own specific objectives observed in the form of various terms and different parameters in the algorithm.

Table 1: Weighted Clustering Algorithms for MANET

Year	Abbreviation	Full Name
2000	WCA	An on-demand weighted clustering algorithm (WCA) for ad hoc networks
2005	WBACA	Weight Based Adaptive Clustering in Wireless Ad Hoc Networks
2006	WCA-L	A Weighted Clustering Algorithm Using Local Cluster-heads Election for QoS in MANETs
2007	EBWCA	An Entropy-Based Weighted Clustering Algorithm and Its Optimization for Ad Hoc Networks
2007	CWBDCA	Advanced Efficiency and Stability Combined Weight Based Distributed Clustering Algorithm in MANET
2007	WACHM	Weight Based Adaptive Clustering for Large Scale Heterogeneous MANET
2008	DEECA	A Dynamic Energy Efficient Clustering Algorithm for MANETs
2008	IWCA(1)	An Improved Weight-Based Clustering Algorithm in MANETs
2008	EE-SWBC	Energy Efficient and Stable Weight Based Clustering for Mobile Ad hoc Networks
2008	FWCABP	A Flexible Weighted Clustering Algorithm Based on Battery Power for Mobile Ad Hoc Networks
2009	EWCA	A Novel Enhanced Weighted Clustering Algorithm for Mobile Networks
2010	Max weight	Maximal Weight Topology Discovery in Ad hoc Wireless Sensor Networks
2011	ECAM	Energy Efficient clustering algorithm

2.3. On-Demand Weighted Clustering Algorithm (WCA) for Ad Hoc Networks

On-demand Weighted Clustering Algorithm (WCA) is the original weighted clustering algorithm, which is cited by many other studies in the field [5]. WCA is an on-demand algorithm, which aims at optimizing the operation of the medium access control (MAC) protocol. Since it is an on-demand algorithm it attempts to decrease the cost spent for both computations and communications. In this algorithm, cluster heads use ‘dual power’ mode in order to transfer a message among clusters in a higher transmission range. WCA procedure is as follows:

- Each node finds its neighbors and the node degree
 - $d_v = |N[v]| = \sum_{v' \in V, v' \neq v} \{dist(v, v') < T_x\}$.
where $N[v]$ denotes the set of neighbors of node v , $dist(u, v)$ denotes the distance between nodes u and v , and T_x denotes the transmission range of a node.
- Each node calculates its degree difference
 - $\Delta_v = |d_v - \delta|$
- where d_v defines the number of node degree and δ is a predefined ideal number of nodes placed within a cluster.
- Each node calculates the sum of the distances to all its neighbors
 - $D_v = \sum_{v' \in N(v)} \{dist(v, v')\}$
- Each node defines the running average of its speed until current time T
 - $M_v = \frac{1}{T} \sum_{t=1}^T \sqrt{(x_t - x_{t-1})^2 + (y_t - y_{t-1})^2}$
where x and y denote the rectangular coordinates of the node in time.
- Each node computes its cumulative time, P_v , which is the time a node can act as a cluster-head.
- Each node then calculates its total weight as follows:
 - $W_v = w_1 \Delta_v + w_2 D_v + w_3 M_v + w_4 P_v; \quad w_1 + w_2 + w_3 + w_4 = 1$

- Considering global minima, the node with the smallest weight value is selected as a cluster-head and the remaining nodes repeat the aforementioned steps until the moment when each node starts acting as a cluster member or a cluster-head. Nodes that belong to the chosen clusters are not allowed to participate in the clustering election again and steps 2 to 7 repeats for rest of the nodes that are not signed as a cluster head or cluster member.

2.4. Weight Based Adaptive Clustering in Wireless Ad Hoc Networks (WBACA)

Weight Based Adaptive Clustering in Wireless Ad Hoc Networks (WBACA) [6] uses the local minima of weight instead of global minima in order to decrease the number of re-affiliations and the time delay within the network; it specifies several parameters such as transmission power, transmission rate, mobility, battery power and the degree of a node to make clusters within a mobile adhoc network domain [6]. An initial phase finds neighborhoods. Each node periodically broadcasts ‘Hello packet’ messages including its identification number. Nodes lying within the same transmission range receive the ‘Hello packet’ messages and broadcast it to their neighbors in response. The second phase includes calculating weight values. Each node calculates its weight and sends it to all its neighbors. The clustering formula is as follows:

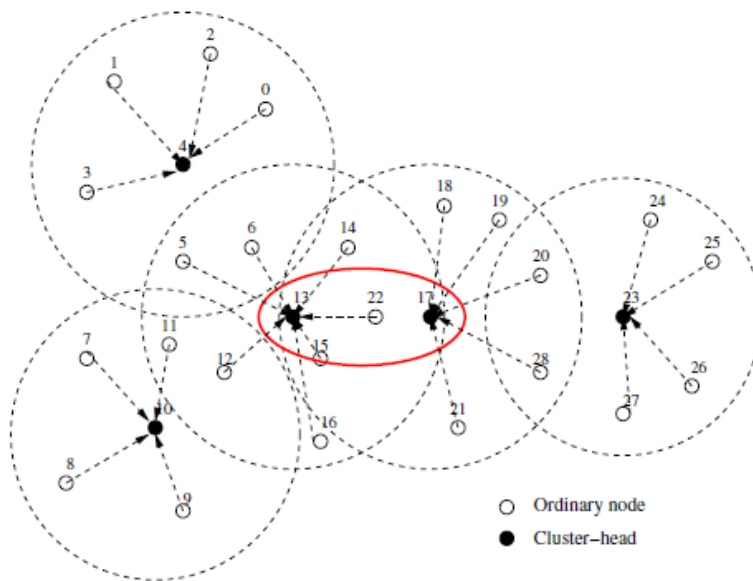
- $$W_n = w_1 * M + w_2 * B + w_3 * T_x + w_4 * D + w_5 / T_r; w_1 + w_2 + w_3 + w_4 = 1$$

where M is node mobility, B is battery power, T_x is transmission range, D is degree difference, and T_r is transmission rate.

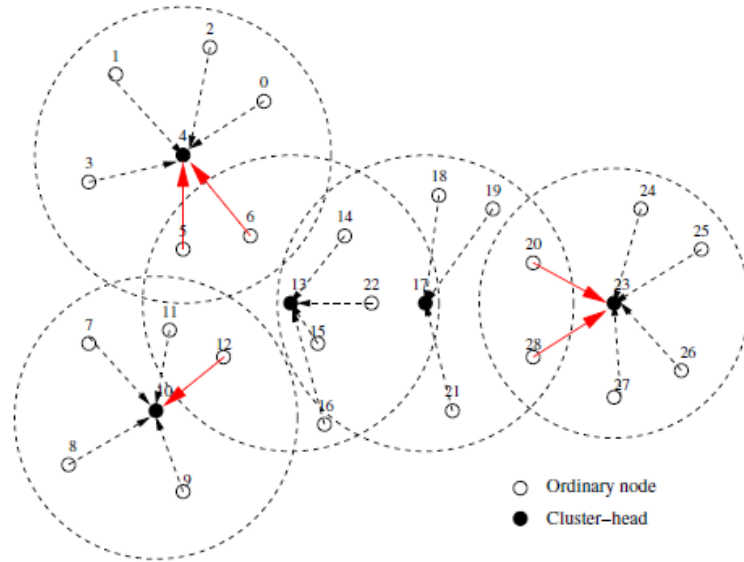
While running WBACA, in a case when there is no node with the smallest weight, the node itself becomes a cluster-head. Otherwise, the node sends a ‘Join-Req’ message to the neighboring cluster-head with the smallest weight. The cluster-head sends a ‘Join-Ack’ message to reply ‘Join-Req’ messages. The cluster-head accepts nodes as its cluster members until its degree becomes equal to the defined threshold value. In a case when a node cannot belong to any cluster, it identifies itself as a cluster-head. Restarting the algorithm takes place at the moment when a link between a node and its cluster-head gets broken.

2.5. Weighted Clustering Algorithm Using Local Cluster-heads Election for QoS in MANETs

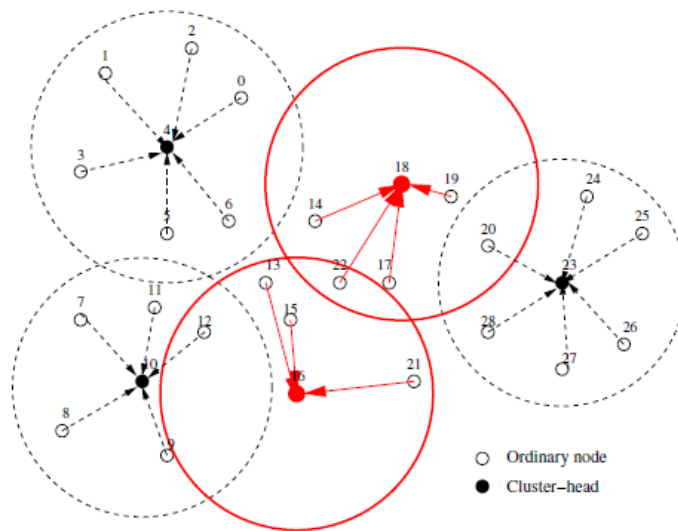
Weighted Clustering Algorithm Using Local Cluster-heads Election (WCA-L) aims at turning isolated nodes into cluster-heads and forms their clusters by invoking an election immediately at the moment when two cluster-heads are one-hop neighbors [7]. Therefore, as the figures below show, the ordinary nodes attempt to affiliate to another cluster and only the gateway nodes that lie within the transmission ranges of two different cluster-heads are successful. Lastly, the remaining nodes and the cluster-heads keep electing clustering process, which is the same as the one defined in WCA.



(a)



(b)



(c)

Figure 1: A Weighted Clustering Algorithm Using Local Cluster-heads Election for QoS in MANETs

2.6. Entropy-Based Weighted Clustering Algorithm and Its Optimization for Ad Hoc Networks

Entropy-Based Weighted Clustering Algorithm(EBWCA) aims at demonstrating the uncertainty concerning the amount of disorder within a network [8]. One of the parameters taken into account in EBWCA is the relative position between two nodes, which is set as below:

$$a_{m,n} = \sum_{i=1}^N |p(m, n, t_i)|$$

where m and n are node IDs and t_i refers to the time instant of the i -th calculation and N is the number of separate times, t_i . And the entropy of the node m is $H_m(t, \Delta t)$ that is defined as follows:

$$H_m(t, \Delta t) = \frac{-\sum_{k \in F_m} P_k(t, \Delta t) \log P_k(t, \Delta t)}{\log C(F_m)}$$

where

$$P_k(t, \Delta t) = \frac{a_{m,k}}{\sum_{i \in F_m} a_{m,i}}$$

F_m defines the set of the node m 's neighbors and $C(F_m)$ is the cardinality of set F_m . The proposed algorithm EBWCA then calculates overall weight for the node v as follows:

$$W_v = c_1 \Delta_v + c_2 D_v + c_3 (-H_v) + c_4 P_v$$

where c_1, c_2, c_3, c_4 are the corresponding weighing factors.

2.7. Advanced Efficiency and Stability Combined Weight Based Distributed Clustering Algorithm in MANET

This algorithm has a hierarchical structure, which aims to optimize network performance by minimizing energy consumption and improving probable problems of efficiency and stability that a network may face [9]. The initial phase of this algorithm is the identification of a neighbor. A set $N[v]$, consists not only of ID numbers of neighbors their locations also as follows:

$$N[v] = \{(k, (x_k, y_k)) : k \in \{\text{ID of neighbors}\}\}$$

In the next phase each node calculates its parameters as given below:

$$w_v = w_1 D_v + w_2 M_v + w_3 E_v + w_4 V; \quad w_1 + w_2 + w_3 + w_4 = 1$$

Where D_v is the sum of distances of node v to all its neighbors, M_v is average running speed of the node v , E_v is the remaining amount of the node v 's battery and ν is the degree difference of the node v , which is calculated as below:

$$d_v = |N[v]| = \sum_{v' \in V, v' \neq v} \{dist(v, v') < T_x\}; \quad \Delta_v = |d_v - \delta|$$

d_v is the number of the node degree within transmission range (T_x) and δ is an ideal number of nodes within a cluster. And finally the node with smallest amount of W acts as a cluster-head. This algorithm uses local minima instead of global minima and moreover there is a maintenance stage, which includes cluster maintenance at resigned cluster-head role and cluster maintenance at moving cluster member node.

2.8. Weight Based Adaptive Clustering for Large Scale Heterogeneous MANET

Weight Based Adaptive Clustering Algorithm (WACHM) attempts to make a large scale hierarchical MANET by utilizing heterogeneous nodes to build various layers [10]. The objectives of this algorithm are considered to be choosing the optimal number of cluster-heads, using optimal hop counts and defined dependency probability and link stability within the network domain.

The first step of the algorithm is to find neighbors, which are categorized as two different types: type-1 (cluster-heads) and type-0 (cluster members). After that each node calculates the following parameters:

$$W_i = w_2 AP_s + w_3 AP_d + w_4 E_r - w_1 d_i; \quad w_1 + w_2 + w_3 + w_4 = 1$$

where

AP_s is the average of link stability P_s over all neighbors:

$$AP_s = \frac{1}{K_{N(i)}} \sum_{K=1}^{K_{N(i)}} P_s(i, j); \quad (j \in N(i))$$

AP_d is the average dependency probability P_d over all neighbors:

$$AP_d = \frac{1}{K_{N(i)}} \sum_{K=1}^{K_{N(i)}} P_d$$

Finally the node with the highest W_i is chosen as a cluster-head this time and all the neighbors of the selected cluster-head are not permitted to participate in the election process.

2.9. Dynamic Energy Efficient Clustering Algorithm for MANETs

Dynamic Energy Efficient Clustering Algorithm (DEECA) contains two important stages: cluster formation and network maintenance [11]. Two defined energy thresholds are taken into account in order to equilibrate the amount of load among cluster-heads within the network. At the beginning all nodes have an undecided status which means they do not belong to any cluster. After that each node broadcasts ‘HELLO’ messages during ‘BROADCAST-PERIOD’ interval. Then each node calculates its own weight as follows:

$$w_v = w_1V + w_2E_v + w_3M_v; w_1 + w_2 + w_3 + w_4 = 1$$

After the weight calculation process each node sends ‘HELLO’ messages including the amount of its weight. Therefore, each node is aware of its immediate neighbor’s weight. Lastly the node with the minimum weight acts as a cluster-head and the nodes in its immediate neighborhood are its cluster-members.

2.10. Improved Weight-Based Clustering Algorithm in MANETs

Improved Weight-Based Clustering Algorithm (IWCA) has been proposed to overcome the problem of high rate of re-affiliations that leads to an increase in the network overhead [12]. The algorithm starts finding the neighbors of each node as follows:

$$D_v = |N(v)| = \sum_{v' \in V, v' \neq v} \{dist(v, v') < T_x\}$$

Following this, each node calculates the parameters below:

- The degree-difference

$\Delta_v = |d_v - \delta|$; where d_v is the number of nodes included in node v ’s transmission range and δ is an ideal number of nodes that v ’s cluster-head can handle.

- Average speed of node v and its neighbors

$$avg_speed = \frac{1}{|N'(v)|} \sum_{v' \in N'(v)} M(v'); \quad N'(v) = N(v) \cup \{v\}.$$

- Speed difference

$$\Delta_{speedv} = |M(v) - avg_speed|$$

- The neighbor change ratio NCR

$$NCR_v = \frac{|N(v)t1 \cap N(v)t2|}{|N(v)t1 \cup N(v)t2|}$$

- Stability factor S_v and improved parameters \overline{M} , where ‘ a ’ and ‘ b ’ are constant numbers.

$$S_v = (a \times NCR_v + 1)^{\frac{1}{\Delta_{speedv+1}}}$$

$$\overline{M}_v = b / S_v$$

- The average relative distances

$$\overline{D}_v = \sum_{v' \in N(v)} \left\{ \frac{dist(v, v')}{|N(v)|} \right\}$$

- The cumulative time P , which shows how long a node acts as a cluster-head in a network

Finally it calculates the combined weight w for each node v

$$w = w_1 \overline{M}_v + w_2 P_v + w_3 \overline{D}_v + w_4 \Delta_v; \quad w_1 + w_2 + w_3 + w_4 = 1$$

and each node chooses the node with minimum amount of weight as a cluster-head; and moreover the isolated nodes also act as cluster-heads.

2.11. Energy Efficient and Stable Weight Based Clustering for Mobile Ad hoc Networks

Energy Efficient and Stable Weight Based Clustering Algorithm (EE-SWBC) tries not to send any additional weight messages and select the most potential node as a cluster-head and decrease the amount of general overhead within the network [13]. Moreover it uses Grey Decision Method (GDM) in order to calculate the amount of the weight values of the nodes. Firstly, each node sends a ‘Hello message’ which includes ID, status, $\sigma I(u)$, $Mt(u)$, $Pt(u)$, $D(u)$ and coordinates. Each node calculates the following parameters:

- Node Degree of node u [$\sigma I(u)$],
- Average Speed of node u [$Mt(u)$]
- Residual Battery Power of node u [$Pt(u)$]
- Average Distance of node u [$D(u)$]

EE-SWBC assumes the node degree and the residual battery power parameters to be the positive criteria and the average speed and distance parameters to be the negative criteria.

Finally, the maximum combined weight of neighbor nodes is calculated by multiplying the weight factors ' w_i ' by the relations coefficient matrix utilized in GDM.

2.12. Flexible Weighted Clustering Algorithm Based on Battery Power for Mobile Ad Hoc Networks

Flexible Weighted Clustering Algorithm (FWCA) aims to make a network stable by decreasing the number of clusters and an amount of overhead due to forming clusters [14]. Moreover it tries to prevent nodes with less battery power act as cluster-heads. FWCA includes two stages: clustering algorithm and clustering maintenance. The moment clustering algorithm starts, each node broadcasts a "beacon" message to recognize its neighbors. All the sensor nodes create a neighbor list considering the received beacons. Then each node calculates the parameters below:

Spreading degree means the difference between cluster's size (the threshold for the cluster members) and the real number of neighbors $R(N)$

$$SP = 1 - (|Size - R(N)|) / size$$

Local clustering coefficient (connectivity) where $L(N)$ is the number of links in the neighborhood and $N(N-1)/2$ is the number of all possible links, where N is the number of mobile sensor nodes.

$$LLC = (L(N)) / ((N(N-1))/2)$$

The remaining battery power BP of the node

The average speed S_i of the node i

$$w_i = w_1 * SP + w_2 * LLC + w_3 * BP + w_4 * S_i; \quad w_1 + w_2 + w_3 + w_4 = 1$$

Finally a node with the minimum amount of weight is selected as a cluster head. Clustering maintenance is applied on the network in the case when battery power threshold is reached and a node's separation from its cluster is large.

2.13. Enhanced Weighted Clustering Algorithm for Mobile Networks

The objectives in Enhanced Weighted Clustering Algorithm (EWCA) are achieving the minimum number of affiliations and the general amount of overhead during the

formation process of clusters and increasing the stability of clusters, system performance and nodes' life time through the adhoc network domain [15]. The moment the clustering algorithm starts, each node calculates the following parameters:

- Set of neighbors, which includes the nodes in a same transmission range.
- Degree-difference: $\Delta_i = |d_i - \delta|$

where d_v is the number of nodes included in node v 's transmission range and δ is the ideal number of nodes that v 's cluster-head can handle.

- The sum of the distances with all its [the node's] neighbors, which is defined based on the received strength of the signal as

$$D_i = \sum_{j=1}^n \text{dist}_{(i,j)} ; \quad \text{dist}_{(i,j)} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

where $\text{dist}_{(i,j)}$ is the distance between two nodes i and j .

- Mobility: $M_i = \frac{1}{\Delta t} |D_i(t) - D_i(t + \Delta t)|$ $D_i(t) = \frac{1}{n} \sum_{j=1}^n \text{dist}_{(i,j)}(t)$

where M_i is the relative speed of a node, $D_i(t)$ is the average distance between the node and its set of neighbors at time t and n is the number of the node's neighbors.

- The consumed energy of a node : $E_i = \sum_{k=1}^q d_{ik} e$

where q is the time duration when a node plays the role of a cluster-head and d_{ik} is the degree of the node which acts as a cluster-head at a time k . E_i defines the amount of battery power consumed during the time that the node acts as a cluster-head.

- A combined weight: $w_i = w_1 \Delta_i + w_2 M_i + w_3 D_i + w_4 E_i$; $w_1 + w_2 + w_3 + w_4 = 1$

Finally nodes with lesser weight are elected as cluster-heads and the defined cluster-heads and members are not allowed to participate in the clustering elections. Moreover at the network maintenance stage there is a feature of controlling the battery power consumption of mobile sensor nodes.

2.14. Maximal Weight Topology Discovery in Ad hoc Wireless Sensor Networks

This algorithm specifies two phases through the adhoc network domain. The first phase is 'information exchange' and the second phase is 'cluster discovery' [16]. The main purpose of the algorithm is minimizing the number of reconfigurations and the number of

cluster-heads within the network domain in order to achieve the optimal topology for the network. After the neighbor recognition step, each node calculates the parameters as follows:

- Node Energy (E)
- Node Mobility (M)
- Node Degree (Δ)
- Neighboring Nodes positions (Π_n)
- Data rate ($\Delta\rho$)
- Target Revisit Rate (T)
- $\Omega_n = w_1E + w_2M + w_3\Delta + w_4\Pi_n + w_5\Delta\rho + w_6T$; $w_1 + w_2 + w_3 + w_4 + w_5 + w_6 = 1$

The node with the maximum amount of Ω is selected as a cluster-head and following the second phase (clustering discovery), it creates its own cluster based on ‘color’ algorithm[16].

2.15. Efficient Cluster-head Election Algorithm Based on Maximum Weight for MANET

Efficient Cluster-head Election Algorithm (ECAM) includes two stages of clustering and maintenance. There are two parameters that are checked during maintenance stage [17]. The first one is ‘the mobility of a cluster’ and the second one is ‘cluster maintenance’. The moment ECAM starts running, each node sends its identification number to its neighbors in order to build a neighbor's table. After that each node calculates its weight as shown below:

$$\bullet w_n = w_1 * n + w_2 * S_n + w_3 * T(n) + w_4 * T_C + w_5 * C_n; \quad w_1 + w_2 + w_3 + w_4 = 1$$

where n is the set of a node’s neighbors, S is the sum of the distances between the node with all its neighbors, $T(n)$ is the speed of the node, T_C is the cumulative time that the node acts as a cluster-head and C is the characteristics of the node.

Finally the node which has the maximum weight value acts as a cluster-head and sends a ‘CH-MSG’ to its neighbors in order to build its cluster.

2.16. Comparison of Weighted Clustering Algorithms for MANET in terms of Their Objectives and Weight Functions

Each of the mentioned weighted clustering algorithms has its own specific objectives and goals. Table 2 has been compiled to represent the aims of weighted clustering algorithms which are applicable in the form of specific parameters listed below:

Table 2: Comparison of weighted clustering algorithms for MANET in terms of their objectives and weight functions

Weighted Clustering Algorithms for MANET	Compared with	The objectives	Parameters considered
WCA [5]	Lowest Id, Highest degree, Node-Weight heuristic	Decreasing the number of re-affiliation, Choosing an optimal number of CHs, decrease latency as possible	Comparison of re-affiliations, Average number of clusters, Re-affiliations per unit time, Dominant set updates,
WBACA [6]	WCA, Lowest-ID	Number of cluster formed in network, Re-affiliation, Starting Delay	Number of Clusters vs number of Nodes and Txrange, Number of Re-affiliations vs number of Nodes and Txrange, Starting Delay vs. No. of Nodes and Tx range
WCA-L [7]	WCA	Restraining the range of an election to an area where at least two cluster-heads are one-hop neighbors. Limiting the overhead	Number of updates of the dominant set, Number of handovers between two clusters, Average packet delivery rate, Overhead
EBWCA [8]	WCA	Increasing network stability Decreasing number of re-affiliations Making a stable network.	Number of re-affiliation
CWBDCA [9]	WCA	Decreasing time to perform clustering, Using "local minima" instead	Average Number of Cluster, Network Overhead, Average Number of CH Change,

		of global minima	Number of Re-affiliations
WACHM [10]	-	Using heterogeneous nodes	Number of cluster-heads Optimal hops Dependency probability Link stability
DEECA [11]	WCA SEECA	Providing energy efficiency and an acceptable level of load balancing among adjacent cluster-heads	Network Lifetime: When 20% of the nodes runs out of energy. Average Dissipated Energy: The average energy consumed by each node.
IWCA(1) [12]	WCA	Reducing the amount of re-affiliation	Average re-affiliation count vs. node number, max speed, Tx range
EE-SWBC [13]	DWCA	Selecting the most potential node as a cluster-head Decreasing the amount of general overhead	Control overhead consumed energy Average LBF
FWCABP [14]	WCA	Controlling battery power consumption	Number of nodes vs. min lifespan of nodes Number of nodes vs. average number of clusters Transmission range vs. stability of clusters.
EWCA [15]	WCA	Maintaining stable clustering structure Minimizing the overhead for the clustering set up, Maximizing lifespan of mobile nodes in the system Achieving good end to-end performance. Minimizing the number of clusters	Number of clusters formed Number of re-affiliations, Lifespan of the network
Max weight [16]	WCA Highest Degree and Topology	Using Local minima Reducing communication cost	Average number of Cluster-heads Number of Re-affiliations per

	Discovery.		unit time Network lifetime Stability of Clusters. Ratio of Cluster heads to total number of nodes
ECAM [17]	WCA EWBCA EWCSA	Electing of efficient cluster-heads in a cluster Increasing the lifetime of the network Improve the stability of the node	Speed with number of cluster head changes Number of node with consumed energy Number of nodes with Life time Number of nodes with End to End delay

Table 3 shows the weight combinations of each specific algorithm mentioned above:

Table 3: weight combinations of weighted clustering algorithms

Name	Calculation of combined weight
WCA	$w_v = w_1 \Delta_v + w_2 D_v + w_3 M_v + w_4 T_v$
WBACA	$w_n = w_1 * M + w_2 * B + w_3 * T_x + w_4 * D + w_5 / T_r$
WCA-L	$w_v = w_1 \Delta_v + w_2 D_v + w_3 M_v + w_4 T_v$
EBWCA	$w_v = c_1 \Delta_v + c_2 D_v + c_3 (-H_v) + c_4 P_v$
CWBDCA	$w_v = W_1 D_v + W_2 M_v + W_3 E_v + W_4 \Delta_v$
WACHM	$w_i = w_2 AP_s + w_3 AP_d + w_4 E_r - w_1 d_i$
DEECA	$w_v = w_1 \Delta_v + w_2 E_v + w_3 M_v$
IWCA(1)	$w = w_1 M_v + w_2 P_v + w_3 D_v + w_4 \Delta_v$
EE-SWBC	GDM matrix of $\sigma_1(u)$, $Mt(u)$, $Pt(u)$, $D(u)$ [13]

FWCABP	$w_i = w_1 * S_p + w_2 * LCC + w_3 * B_p + w_4 * S_i$
EWCA	$w_i = w_1 \Delta_i + w_2 M_i + w_3 D_i + w_4 E_i$
Max-w	$\Omega_{\sigma} = w_1 E + w_2 M + w_3 \Delta + w_4 \Pi_{\sigma} + w_5 \Delta \rho + w_6 T$
ECAM	$w_n = w_1 * n + w_2 * S_n + w_3 * T(n) + w_4 * T_C + w_5 * C_n$

2.17. Weighted Clustering Algorithm for WSN

It is necessary to mention that since MANET and WSN are relatively similar networks, some of the aforementioned weighted clustering algorithms for MANET are applicable to WSN as well. However Table 4 defines the weighted clustering algorithms, which are built specifically for WSN.

Table 4: Weighted Clustering Algorithms for WSN

Published Year	Name	Weighted Algorithms Full Name
2005	DWEHC	Distributed Energy-Efficient Hierarchical Clustering for Wireless Sensor Networks
2008	CFL	A Clustering Algorithm For Localization in Wireless Sensor Networks
2010	IWCA(2)	An Improved Weighted Clustering Algorithm for Determination of Application Nodes in Heterogeneous Sensor Networks

2.18. Clustering Algorithm for Localization in Wireless Sensor Networks

Clustering Algorithm for Localization in Wireless Sensor Networks (CFL) aims at achieving the minimum number of clusters with maximum number of nodes inside in order to improve the general performance of weighted clustering algorithm [18]. At the moment the CFL algorithm starts by all nodes broadcasting the ‘Hello’ message through the network and building their ‘neighbor table’ based on received messages, which includes estimation of distances also. Then each node calculates the parameters below:

- the remaining energy (E_i)

$$E_i = \frac{E_r}{E_{max}} \quad \text{where } E_r \text{ is the available energy of a node and } E_{max} \text{ is the initial amount of}$$

energy of a node.

- the number of node’s neighbors (N_i)
- the transmission power (P_i)
- the combined weight (w_i)

$$w_i = aN_i + bE_i + c\frac{1}{P_i}; \quad (a+b+c \leq 1)$$

Lastly, the node with the maximum weight value acts as a cluster-head and sends a ‘CH-msg’ to its neighbors and the nodes that receive this message change their states to cluster members. The remaining nodes, which do not belong to any clusters, change to cluster-heads.

2.19. Improved Weighted Clustering Algorithm for Determination of Application Nodes in Heterogeneous Sensor Networks

Improved Weighted Clustering Algorithm (IWCA) attempts to increase the network life time and in addition to forming clusters it includes network maintenance. Network maintenance checks two thresholds for energy amount of nodes, which triggers the recalculation of the clustering algorithm [19]. There is an additional parameter that is added to WCA formula; i.e., the characteristics C of each node, which is calculated as:

$$C_v = \frac{C * r_v}{E_v}$$

where C is a constant number for amplification, r_v is the transmission rate and E_v is the initial energy of a node.

The combined weight is calculated as given below:

$$w_v = w_1 \Delta_v + w_2 D_v + w_3 M_v + w_4 T_v + w_5 C_v; \quad w_1 + w_2 + w_3 + w_4 = 1.$$

The node with the minimum weight acts as a cluster-head and forms its cluster.

2.20. Distributed Energy-Efficient Hierarchical Clustering for Wireless Sensor Networks

This algorithm utilizes two parameters – residual energy and distance with neighbors – in order to calculate the combined weight of each node [20].

$$W_{\text{weight}}(s) = \left(\sum_{u \in N_{\alpha,c}(s)} \frac{(R-d)}{6R} \right) \times \frac{E_{\text{residual}}(s)}{E_{\text{initial}}(s)}$$

where R defines the cluster range, d specifies the distance from node s to the neighboring node u , E_{residual} is the residual energy in node s , E_{initial} is the initial energy in node s which is the same for all the nodes and $N_{\alpha,c}$ is the set of the neighbors of the node s where α is equal to 2 or 4 and c is a constant. It is assumed that the number of neighboring nodes of a cluster is at most 6.

2.21. Comparison of Weighted Clustering Algorithms for WSN in Terms of Their Objectives and the Weight Functions

Table 5 has been compiled to represent weighted clustering algorithms for WSN with their specific objectives as well as the parameters that are checked through the algorithms.

Table 5: weighted clustering algorithm for WSN

Weighted Clustering Algorithms for WSN	Compared with	The objectives	Parameters considered
CFL	ACE	Achieving good distribution of cluster heads on the network	The time required to create all clusters vs the number of the cluster head over all the nodes in the network.
IWCA(2)	WCA	Increasing lifetime (longer battery life)	The lifetimes and the execution time of the two algorithms for different numbers of base stations
DWEHC	HEED-AMRP	Achieving optimal energy consumption	Average energy used for intra-cluster Average energy used for intra-cluster Energy used for inter-cluster Energy used for inter-cluster Number of rounds until the first node dies Number of rounds until the first node dies

Table 6 summarizes the ‘combined weight’ calculations specifically for each algorithm mentioned above:

Table 6: the ‘combined weight’ calculations specifically for each algorithm

Name	Calculation of combined weight
CFL	$W_i = aN_i + bE_i + c\frac{1}{P_i}$
IWCA	$W_v = w_1\Delta v + w_2Dv + w_3Mv + w_4Tv + w_5Cv$
DWEHC	$W_{weight}(s) = \left(\sum_{u \in N_{a,c}(s)} \frac{(R-d)}{6R}\right) \times \frac{E_{residual}(s)}{E_{initial}(s)}$

CHAPTER 3

LIFE TIME SENSITIVE WEIGHTED CLUSTERING ALGORITHM

The present chapter first emphasizes the main differences between the original WCA and LTS-WCA proposed in this chapter; it discusses the assumptions concerning the implementation of the conducted algorithm and explains the proposed algorithm in details.

3.1. Differences with the Original WCA

Life-Time Sensitive Weighted Clustering Algorithm (LTS-WCA) is a fully distributed algorithm, which is applicable for heterogenous Mobile Wireless Sensor Networks. It aims to solve the predefined problems included within the reviewed Weighted Clustering Algorithms and modifies these previous works efficiently in order to apply them on WSN domain. Although LTS-WCA is designed for a heterogenous Wireless Sensor Network, it can be used on homogeneous WSN, too. There are several significant modifications on the original version, which can be listed as follows:

- LTS-WCA uses local minimum instead of global minima, which means each decision can be made by a group of nodes in a local manner and there is no need for a node to get aware of the other nodes' decisions and specifications.
- There is a specific life time assigned on each protocol packet of a node in terms of hop count in order to limit packet retransmissions.
- Sensor nodes communicate in a multi-hop fashion both for inter- and intra-communications.
- Several additional parameters are included in the clustering algorithm to be used to calculate node weights and to find efficient cluster heads. The additional parameters used are:
 - E_r = remaining energy amount of a node
 - T_R = transmission range of a node

- S = Size of the cluster that a node can support as a cluster head
- dv = Number of 1-hop neighbors of a node
- There is an enhancement part which is proposed and considered in order to improve LTS-WCA performance in terms of energy efficiency and network life time. It considers four parameters, i.e., the ‘energy of nodes’, ‘node mobility’, ‘load put on a cluster head’ and ‘ D_{\min} ’ (the minimum number of hops that a node is far from its cluster neighbor's cluster-head) . There are two thresholds that will be defined later in the thesis regarding the enhancement in terms of energy of nodes: the minimum energy amount needed for each node to act as a cluster head and the minimum energy amount needed for a node to act as relay node.

3.2. Assumptions on the Implementation of the Proposed Algorithm

There are several assumptions that should be taken into consideration in order to develop the proposed algorithm:

- The network is assumed to carry data traffic and control traffic (which is generated by WCA and the related mechanisms).
- All sensor nodes in the network environment are assumed to be mobile nodes but they may have different speeds.
- All sensor nodes move randomly in different directions and their moves cannot be controlled.
- All nodes can move only in two dimensions on a flat environment.
- The network is heterogeneous.

There are two different types of wireless sensor networks: heterogeneous and homogenous. Heterogeneity may have different definitions on various research works. However, what we mean by heterogeneity is that each sensor node has its own special properties such as transmission power, transmission range, initial power, computation power, speed (mobility) and memory. Moreover, for each single node there is a possibility to act as a “cluster-head” or a “cluster member”. Using heterogeneous nodes in wireless sensor networks helps us in decreasing

response time, improving battery life time and making the network balanced in terms of general energy usage.

- The transmission power for each node is not adjustable. It means each node has its specific transmission range considering its initial energy amount and it doesn't change until the network dies.
- The predefined life time is assigned to each packet of a node in order to control packets in terms of hop count and each node defines its own packet life time depending on its residual energy.
- Each cluster has predefined limit on the number of nodes that it can accommodate as cluster members.
- Gateways, which are nodes that lie within transmission ranges of two different cluster heads, are employed. Some of the cluster members can act as gateways.
- There is a single source of energy, which is used for all purposes.
- Initial distribution of nodes goes beyond our research area and we assume that all nodes are distributed in such a way that they cover the whole network environment.
- At the initial step of our proposed algorithm, we assume that all nodes have undecided state and at the end of clustering, nodes are assigned 'cluster-head' or 'cluster member' as their states.
- Immediate neighbors defined for each node is the set of nodes that are 1-hop away from it.
- A neighborhood defined for a node i is the set of nodes that are able to receive packets from i in a multi-hop manner.
- A relay node is a node which transfers packets generated within its cluster to its "cluster head".

3.3. The Proposed Algorithm – Life-Time Sensitive Weighted Clustering Algorithm (LTS-WCA)

At the exact moment when clustering timer is triggered through the network, each mobile wireless sensor node gets prepared to follow the steps below:

- Step 1: each node recognizes the nodes in its neighborhood.

- Step 2: each node calculates its parameters as below :
 - S= Ideal number of cluster members in a cluster

$$S = \frac{N K^2 T_r^2 \pi}{A}$$

Where

K = the number of hops that a node can support inside of its cluster in a case it becomes a cluster head.

N= number of nodes that can receive transmitted packets from a source node.

A= area of the network

T_r = Transmission range of a node

- Speed of a node (direction-independent)
- Remaining Energy:
 - E_r = amount of residual energy of a node
- Degree of a node (d_v), which means the number of 1-hop neighbors of a node
- Weight of a node:
 - $W = (w_1 T_r + w_2 M_v) / (w_3 d_v + w_4 E_r + w_5 S)$
 - The amount of parameters w_1, w_2, w_3, w_4 and w_5 are adjustable such that $w_1 + w_2 = 1, w_3 + w_4 + w_5 = 1$.
- Step 3: Each node prepares its data structures considering the above calculations and broadcasts it.
- Step4: Each node compares its calculated weight with other neighbors' weights.
- Step 5: Each node checks whether it has the least weight.
 - If a node has the least weight, it sends a 'ch-msg' including its ID and weight to all its neighbors and changes its state as a cluster head. Moreover after receiving each 'join-req' it starts to accept nodes until the number of nodes within its cluster doesn't exceed the defined threshold.
 - If a node does not have the least weight amount, it checks whether it receives any 'ch-msg'. If so it sends a 'join-req' to selected cluster head and

after receiving acceptance from the cluster head it changes its state as cluster member.

- Isolated nodes, which cannot join to any cluster, change their state as cluster heads.

At the end of the clustering algorithm each node has a cluster head or cluster member state within the defined clusters. It is worth noting that during exchanging of information among sensor nodes, each node checks whether the received data is new or not. If it is new, it propagates it to its 1-hop neighbors considering the packet's life time field, if it is not, it simply drops it. Moreover, each node updates the packet life time field once in every run of the clustering algorithm considering its remaining energy.

During each run of the clustering algorithm, each node forms a table containing the following items:

Table 7: an adjacent table

ID	CID	W	state	Er
----	-----	---	-------	----

The fields are explained below:

- ID: Identification number of each node
- CID: Cluster identification number
- W: Weight of a node calculated by the clustering algorithm, which shows the goodness of a node in order to get elected as a cluster head
- State: it can be one of these three items: cluster member (CM), cluster head(CH)
- Er: Amount of residual energy of a node

There are two different kinds of messages being transferred within the network domain, which play critical roles in the algorithm.

- 'ch-msg' sent by cluster heads to make neighbors aware that they have become cluster heads
- 'join-req' sent by neighborhood nodes to cluster heads to ask if they can get accepted as cluster members

3.4. Enhancements parameters proposed on LTS-WCA

The present subsection presents ideas that may be used to enhance the performance of the proposed approach.

- D_{min}:

Nodes from different clusters are called gateways if they lie within the same transmission range with other nodes in a different cluster. In order to calculate the number of hops that a node is away from cluster head of a neighboring cluster (D_{min}), the cluster gateway nodes exchange information about their distances in hop counts with the gateway nodes of the neighboring cluster. As long as the gateway nodes can hear each other, they keep exchanging such information among clusters. So the rest of the nodes can get aware of the number of hops that they are away from neighboring cluster heads. The D_{min} parameter can be included in weight calculation as follows:

$$W = (w_1 D_{min} + w_2 T_r + w_3 M_v) / (w_4 d_v + w_5 E_r + w_6 S)$$

By decreasing the distance among the cluster heads, the amount of energy consumption for transmitting data to sink and the number of isolated nodes in the network can be decreased considerably.

- Energy Thresholds:

One of the most significant parameters that need to be checked periodically is the amount of residual energy of mobile wireless sensor nodes. There are two thresholds specified for two different positions of energy amount of both relay and cluster head nodes.

- TE1 is the minimum energy amount needed for each node to act as a cluster head. The moment that a cluster head node's energy reaches TE1, it sends a signal through the members identifying that it is not going to act as a cluster head anymore and triggering the cluster algorithm run through the cluster.
- TE2 is the minimum energy amount needed for a relay node to act efficiently in a network domain. The moment a relay node's energy amount reaches TE2, it sends a signal and transfers all the packets it has; and finally it breaks its links to and from other sensor nodes. Therefore the information it has already is not dropped suddenly.

- Node mobility:

The third parameter, which is considered as another important issue to check is a node's mobility. Considering that the wireless sensor nodes are mobile and considering their heterogeneous, the speed of each node may be different from other nodes. Therefore mobility of cluster heads and members both affect the running frequency of the clustering algorithm within the wireless network domain.

While a cluster member is preparing to move and affiliate to other possible clusters within the network domain, it sends a signal through the current cluster to inform nodes of its re-affiliation, which means it separates from its cluster and joins to another one. After its separation from the cluster, it tries to join neighbor clusters, which it gets to know beforehand. Then the node follows the steps presented below to complete its re-affiliation:

- 1) It checks the possible nodes within a neighbor cluster to join them, either cluster members or cluster head node.

- 2) Considering the received information from the previous cluster gateway nodes, it chooses the least hop count possible to join a neighbor cluster.

- 3) It sends a 'join-req' and its own table to the neighbor cluster's cluster head node.

- 4) The cluster head node checks the received information and compares its weight if it is still the least one. After this comparison, the cluster head decides whether it still acts as a cluster head or it should leave the competition and change its state to cluster member.

- 5) If there is any change regarding the cluster heads, it is announced through the cluster and each node updates its table considering recent modifications.

Since in LTS-WCA, one of the considered parameters for calculating the amount of a node's weight is the node speed and due to the fact that a possibility of a node to be elected as a cluster head with the least amount of speed is higher than other nodes because the least amount of speed leads to having the minimum amount of weight, the act of separation of a cluster head node from its cluster can rarely happen. When a cluster head is going to separate from its cluster, it sends a signal to inform cluster members about its separation and triggers the clustering algorithm to run one more time within a cluster. The new cluster head gets chosen and each node updates its table considering recent modifications. Then the separated cluster head acts as a normal node which tries to affiliate to other neighbor clusters.

It is possible that a competition among cluster heads takes place during the clustering process. This may happen when two cluster heads act as rivals to each other, which happens when a cluster head receives a 'ch-msg' from another cluster head. The node checks the received 'ch-msg' to see whether the weight of a rival cluster head is less than its own weight value. In such a case, it exits the competition and sends a 'join-req' to its rival and waits until it receives acknowledge. Otherwise it keeps its state as a cluster head.

- The amount of load put on a cluster head:

The last option to check in maintenance part is the load that a cluster head handles it. As was mentioned before, parameter S defines the maximum number of nodes that a node can have within its cluster in a case that the node becomes a cluster head. Since the wireless sensor nodes are mobile, they are attaching and de-attaching to/from clusters and the number of cluster members is dynamic. There is a need for a cluster head to check its cluster size in a defined period of time. Therefore in a case that it becomes overloaded, it can de-attach the extra cluster members by informing them beforehand and then breaking the links between.

CHAPTER 4

SIMULATION WORK AND EVALUATION RESULTS

In the present chapter the simulation process, the required properties, the topology of the network environment and, finally, the result of our LTS-WCA implementation are described.

4.1. Nodes categorizations

Since the proposed WSN is a heterogeneous network, all nodes within the network are put into three categories and each category has its specific amount of initial energy, transmission range, speed and direction.

There are two constants for each node, which are fixed from the beginning of the network initialization until the end of the network's life time. These are node transmission range and speed. The others, i.e., node energy and direction, are variable within the network environment.

4.2. Mobility Model

The mobility model considered in the present proposed algorithm contains three parts: speed of a node, speed limitation and direction.

Initially a random number between two thresholds 5(m/s) and 10(m/s) is assigned to each sensor node in the network environment. It is considered that there are no additional factors within the network environment to make changes in the speed of the sensors nodes. Therefore they keep the initial pre-set speed until the very end. In order to provide higher

network coverage, it is considered that all nodes can move through the predefined directions, which are assigned to each node randomly in the initial phase of algorithm.

Moreover, the nodes may change their directions during the predefined time periods. In case a node reaches to the network edge, it turns automatically and keeps moving in other directions within the network area.

4.3. Energy Model

All nodes are divided into three different categories and three predefined amounts of energy are assigned initially to each category of nodes. A single node is expected to lose a certain amount of energy while various transactions take place, such as packet transmission and computation and node movement.

Assigning three different initial amounts of energy to each node makes them distinct in their capacity to transmit and process data. In the initial phase of the our algorithm, the following energy amounts are assigned:

- 20 Joules to 10% of the sensor nodes within the network environment.
- 15 Joules to 60% of the sensor nodes within the network environment.
- 10 Joules to 30% of the sensor nodes within the network environment.

Following the assignment of the initial energy amounts to nodes, each node category assigns a predefined transmission range, which will not change until the end of the network's life time based on the following:

- The nodes which have the battery size of 20 Jules obtain 100m as their transmission range
- The nodes which have the battery size of 15 Jules obtain 70m as their transmission range
- The nodes which have the battery size of 10 Jules obtain 30m as their transmission range

Moreover there is a specific number of hops that a single node can support within its cluster while it acts as a cluster head in the network environment. This number is assigned based on the remaining energy of each node. These numbers are selected between 4 and 0,

which are initially assigned to three node categories in the network environment as shown below:

- The node with the highest initial amount of energy can support a neighborhood area of 4 hops.
- The node with less initial amount of energy can support a neighborhood area of 3 hops.
- The node with the least initial amount of energy can support a neighborhood area of 2 hops.

It is worth to mention, though, that during the running of the algorithm, these amounts may decrease on the basis of the remaining energy amount of a node.

4.4. Traffic Model

There are three node categorizations with three different transmission powers. It is considered that each node has the equal amount of T_x and R_x . The traffic model which is used among sensor nodes in the network environment is constant bit rate CBR . Each node broadcasts its gathered information every T seconds with broadcast rate R within its transmission range periodically (for instance 10 packets per second). Packets are sent to sink nodes through gateways and cluster-heads in a multi-hop fashion.

4.5. Implementation Phase of LTS-WCA

Life-Time Sensitive Weighted Clustering Algorithm was implemented on NS2.34 simulator with Mannasim patch file installed on it. Mannasim is a Wireless Sensor Networks simulation framework, which is used for designing, developing, analyzing and clustering Wireless Sensor Networks.

There are several Classes on mannasim specially designed for clustering purposes on WSN. They include: `SensorNode` class, `BatteryClass`, `DataGenerator` class, `TemperatureDataGenerator` class, `TemperatureAppData` class, `Processing` class, `SensedData`

class, *OnDemandData* class, *OnDemandParameter* class, *SensorBaseApp* class, *CommonNodeApp* class and *ClusterHeadApp* class.

Each Mannasim class serves the specific purpose described below:

- *SensorNode* class: This class specifies a wireless sensor node. It is an extended class of NS2 *MobileNode* class aimed at adding more specific sensor node characteristics (such as power consumption) in order to sense the environment and processing activities.
- *BatteryClass*: This class defines sensor node power supplies. It is an extension of NS2 *EnergyModel* class. The *Battery* class can support various types of existing battery models. The class includes the methods for turning sensor nodes on and off and putting them into sleep mode or waking them up.
- *DataGenerator* class: This class has a duty of simulating the sensing tasks of a wireless sensor node. It enables a node to adjust sensing interval, generate and collect data. It can be used to generate different sensed information such as temperature, light, magnetic fields, etc.
- *TemperatureDataGenerator* class: This class can be used for simulating temperature sensing task of a wireless sensor node. It makes an extension for *DataGenerator* class in order to add more characteristics concerning the temperature data.
- *TemperatureAppData* class: This class has the ability to encapsulate raw data that is generated by *TemperatureDataGenerator* class.
- *Processing* class: This class develops a base for every type of data processing on the sensed data in a sensor node.
- *SensedData* class: This class makes up the sensed data after processing.
- *OnDemandData* class: This class shows the request messages sent out of the WSN.
- *OnDemandParameter* class: This class makes up the queries that are sent within *OnDemandData* request messages.
- *SensorBaseApp* class: This class is a foundation for WSN applications. It defines node applications such as common node and cluster head ones.
- *CommonNodeApp* class: This class defines the provided implementation that *SensorBaseApp* class sets to its virtual methods.

- *ClusterHeadApp class*: The ClusterHeadApp class is able to simulate the Cluster Head's behavior within a hierarchical WSN.

4.6. Simulation Scenario

Since a WSN includes distributed wireless sensor nodes within the network environment and since most of the nodes used in the network are not stationary, it is considered as a great advantage to sense and cover the network environment in a wider range.

One WSN usage is applying it for the purpose of fire detection. In the proposed scenario, LTS-WCA is applied on a WSN in order to sense the temperature of environment and give an alarm in case there is a possibility of an unpredicted fire through the forest. It is assumed that wireless nodes that are used in the network are mobiles and they are disturbed through the forest in a random manner. They are able to sense and process the data and inject this into the Wireless Sensor Network. The Nodes are clustered via LTS-WCA in order to send the gathered data to a sink in an energy efficient way.

4.7. Implementation of LTS-WCA

It is assumed that there are mobile wireless nodes in a 1000*1000 square. They are categorized into three types of nodes with speeds of 7,5 and 3m/s. Node transmission ranges are 200,150 and 100meters. Initial energy of nodes are 20,15 and 10Joules.

It is worth mentioning that although LTS-WCA was initially designed especially for heterogeneous WSNs, it can be used for homogenous WSNs and it can also be applied on MANET, if the density of the mobile nodes within the network is not large. This is believed to be its biggest advantage.

Table 8: LTS_WCA parameters

Parameters	Value
Transmission range	100, 150,200
Area	1000*1000
Traffic type	CBR
Tx power	0.036 Jules
Rx power	0.024 Jules
Initial power	20, 15, 10 Jules
Number of nodes	50, 100, 200
Max packet in ifq	200 packets
w_1, w_2, w_3, w_4, w_5	0.5,0.5,0.4,0.4,0.2

The simulation results of CFL, WCA and WBACA are directly borrowed from the respective publications [6], [18]. The comparison results of LTS-WCA performance with the existing Weighted Clustering Algorithm on MANET are shown below:

In order to compare the result of LTS-WCA with WCA and WBACA, we applied the same MANET environment (1000*1000meters square space) with same values of speed for nodes (10m/s). The following LTS-WCA plots are obtained as averages of 50 consecutive runs of the simulation, each starting with a new randomly assigned node distribution.

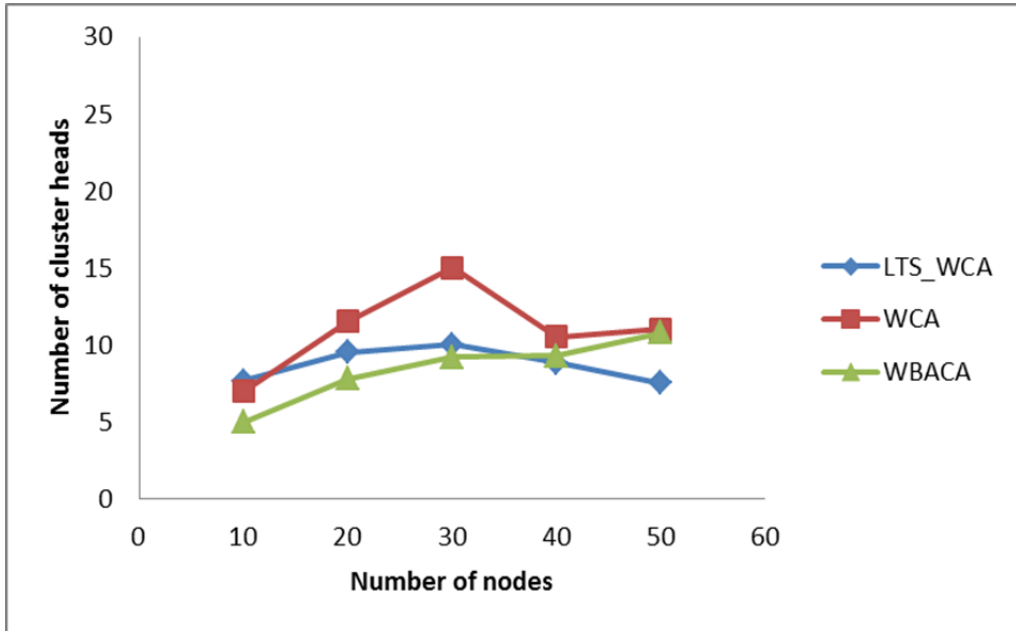


Figure 2: Nch vs Nn

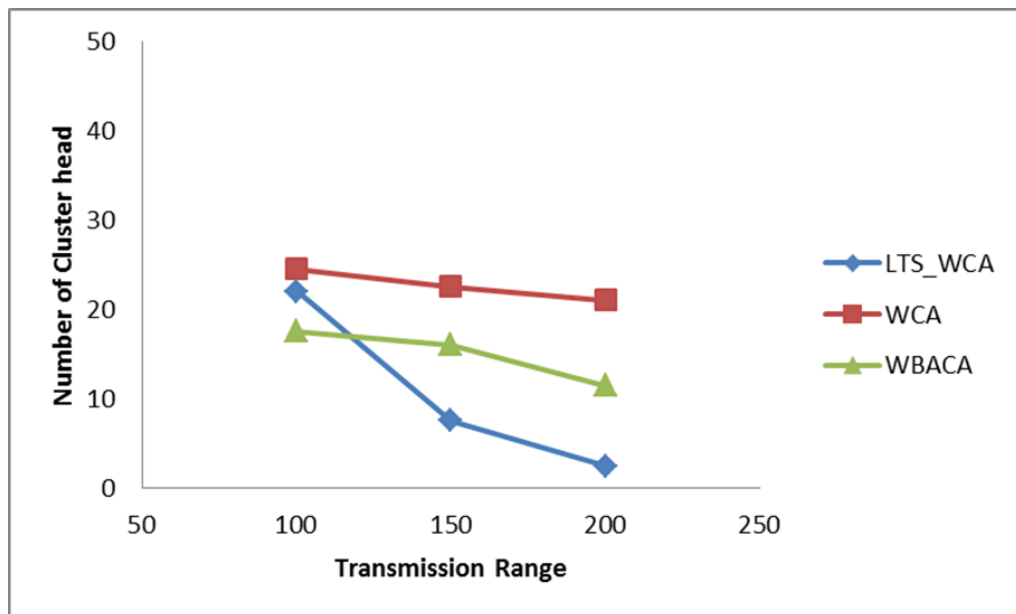


Figure 3: Tr vs Nch

Figure 2 shows the comparison of LTS-WCA with two different Weighted Clustering Algorithms (WCA and WBACA) by considering two parameters: *number of cluster heads* and *number of nodes*. As the figure, by increasing the number of nodes the number of clusters increases until the number of nodes reaches 30. After this point in both WCA and LTS-WCA the number of cluster heads decreases. The reason behind this is the fact that when the node density is small, each node takes a cluster head role in the environment. However, by increasing the number of nodes further, the possibility of nodes belonging to a cluster increases and this decreases the number of cluster heads, which in return decreases the energy consumption to transmit data to sink. As figure 2 shows the performance of LTS-WCA is much better than that one of WCA [5] and it also is better than WBACA [6] when the number of nodes gets larger. Figure 3 presents results considering two parameters: *number of cluster heads* and *transmission range*. By increasing the node transmission range within the network, the number of cluster head decreases. As it is seen in the figure the performance of LTS-WCA in decreasing the number of clusters and therefore transmission energy consumption is much better than that of two other algorithms.

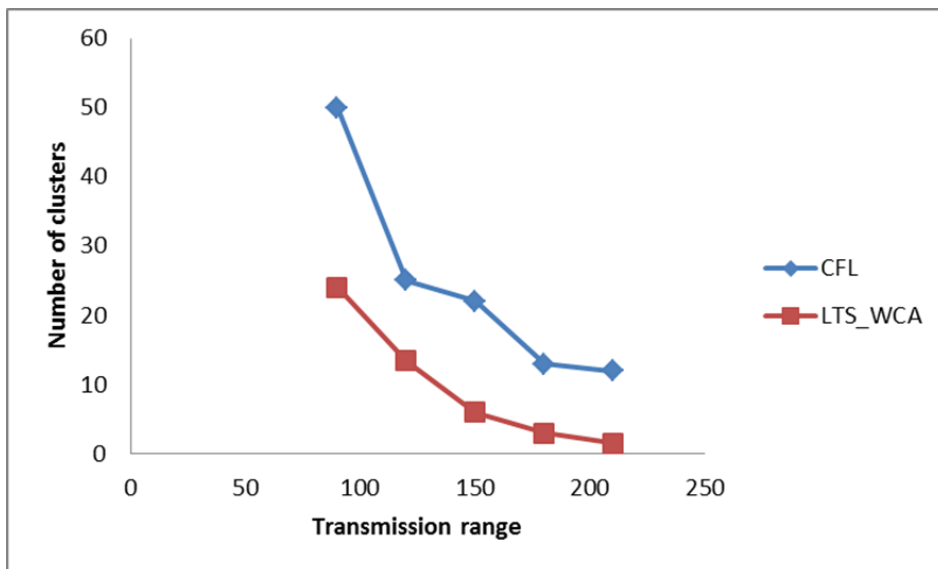


Figure 4: N_c vs T_r

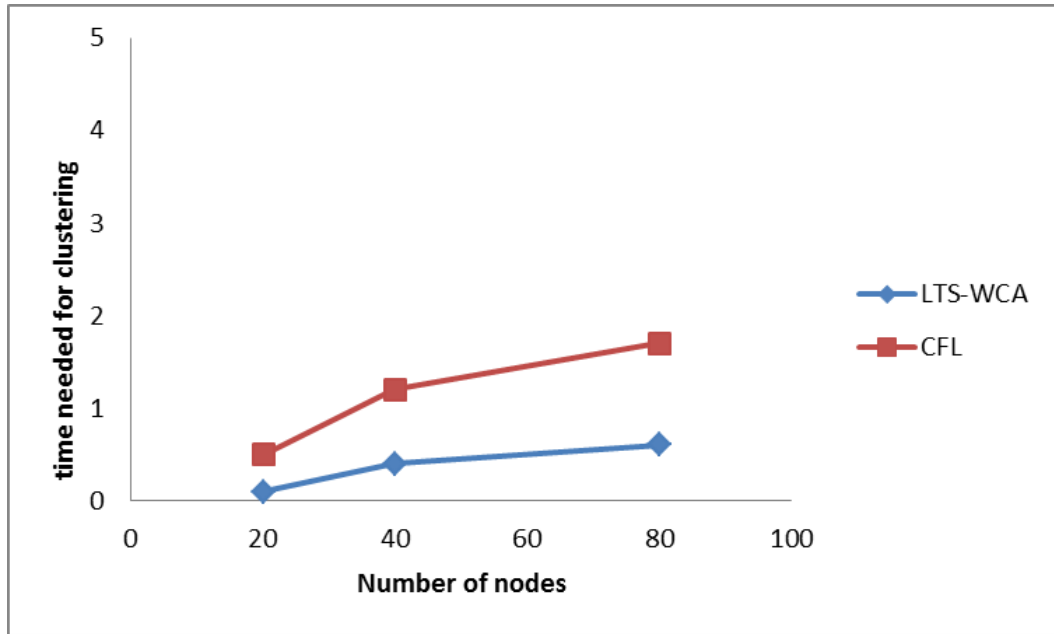


Figure5: Time for clustering vs Nn

Figure 4 shows the comparison between LTS-WCA and CFL [18] algorithm on WSN. By increasing the transmission range, the number of clusters within the network decreases and this leads to the decrease of the amount of energy needed to transmit data from the source to a sink. As the graph represents LTS-WCA has a considerably better performance in comparison with CFL [18] algorithm.

Figure 5 illustrates the change in the time needed for clustering vs the number of nodes in WSN. As seen in the figure, LTS-WCA is quite successful to decrease the clustering time in comparison with CFL [18].

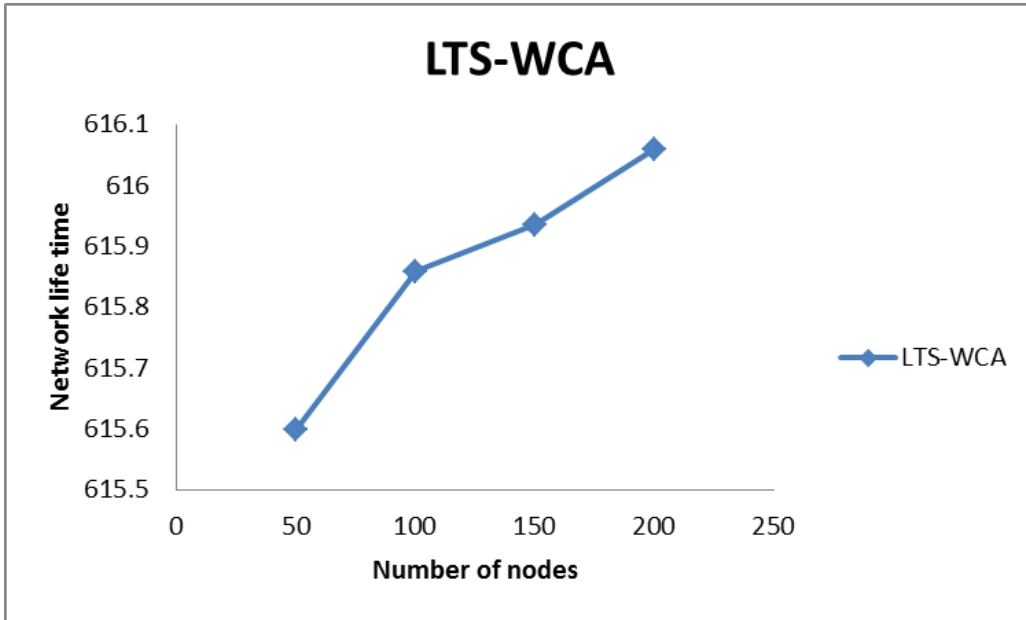


Figure 6: LT vs Nn

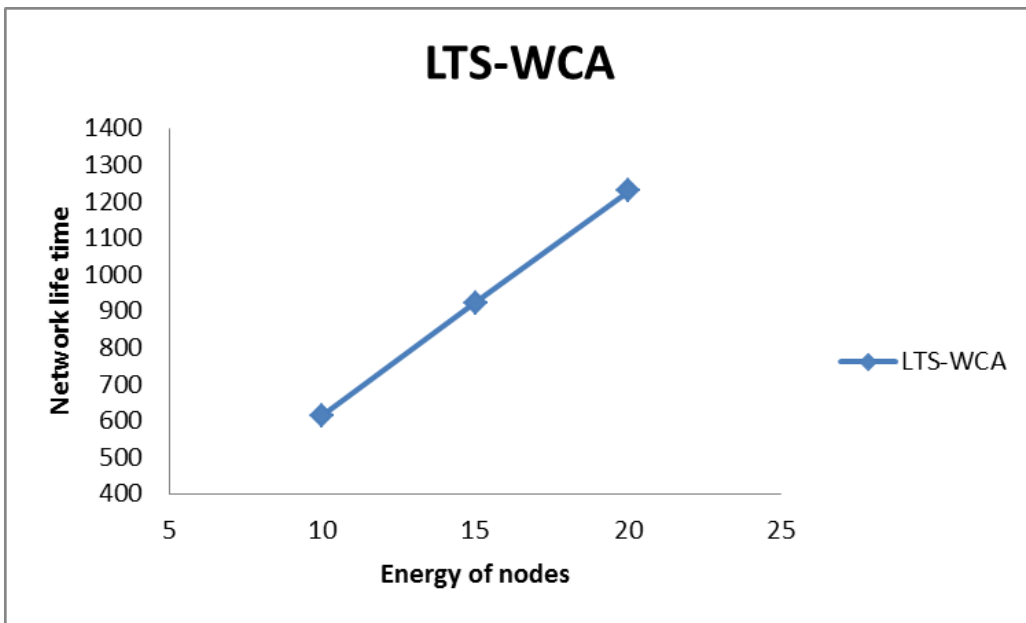


Figure 7: LT vs En

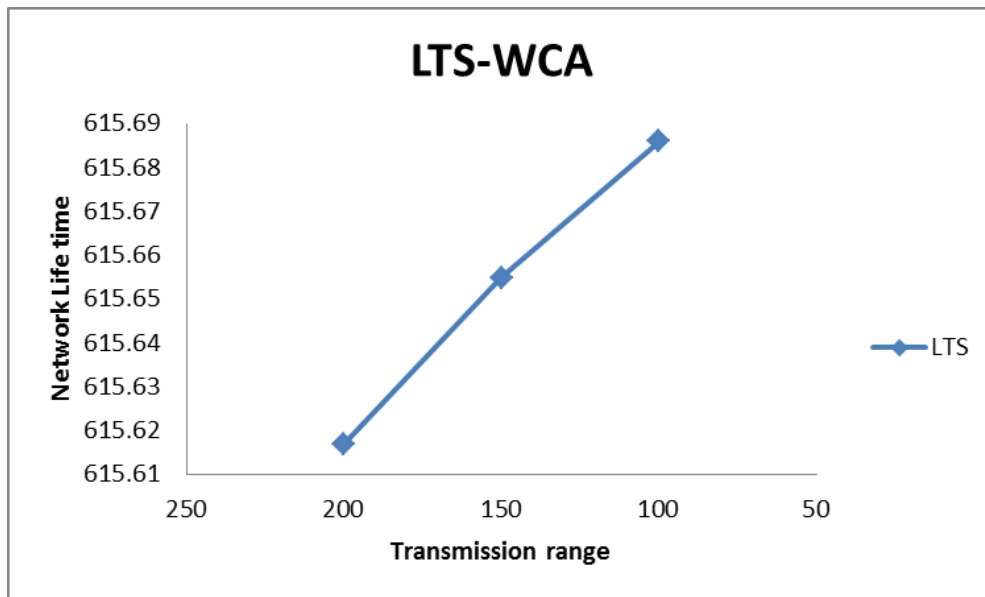


Figure 8: LT vs Tr

Figures 6, 7 and 8 present the performance of LTS-WCA in a homogenous WSN to show how energy efficient it is in terms of network life time. As was mentioned earlier, the moment the first node dies is considered as a network life time in LTS-WCA for simplicity. In Figure 6 by increasing the number of nodes, the network life time increases gradually. Figure 7 shows the considerable increase network life time if initial energy of nodes within the environment is increases. Figure 8 presents the decrease of the network life time of WSN by increasing the nodetransmission range. The reason behind this is the fact that by increasing the transmission range of nodes, the size of clusters increases, too. Finally cluster heads become overloaded and they start to lose their energy much faster. As a result, the network life time also decreases.

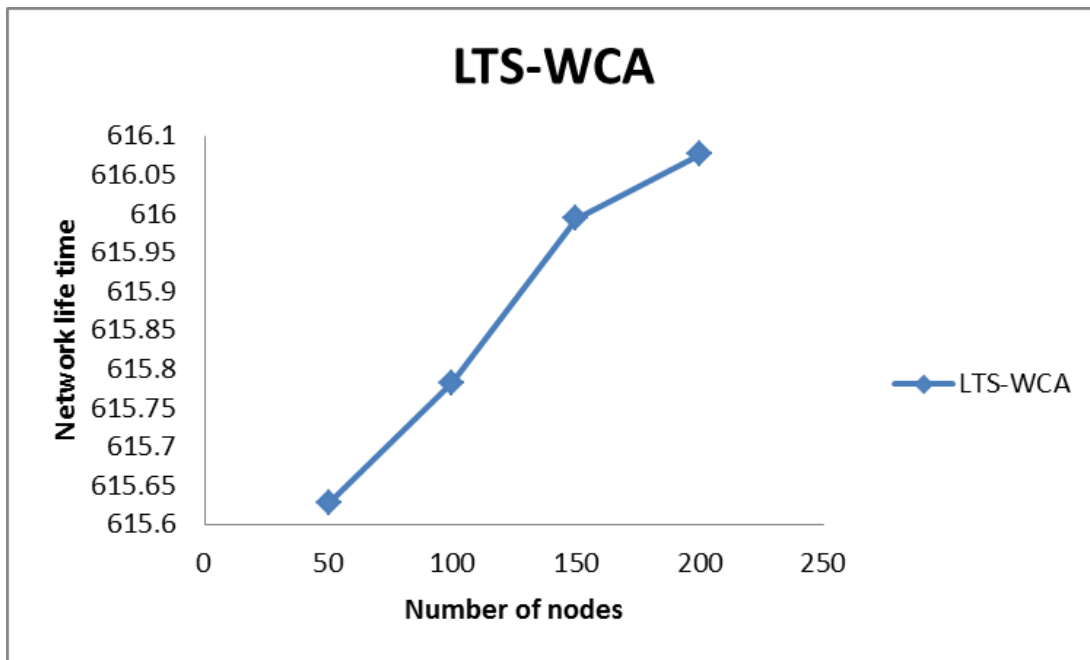


Figure 9: LT vs Nn

Figure 9 represents the performance of LTS-WCA on a heterogeneous WSN. By increasing the number of nodes, the network life time also increases.

CHAPTER 5

CONCLUSIONS

In the present thesis, we studied the vast amount of research done in the field of weighted clustering algorithm for two different networks: Mobile Adhoc Network and Wireless Sensor Network. We examined their main motivations concentrating mostly on the energy efficiency and network overhead. Since in WSN life time is considered to be a vital issue, researchers mostly take it as a significant parameter to be improved within their proposed clustering algorithms [19], [20]. However, along with life time, the issue of energy efficiency plays an equally important role. Therefore, it became the second emphasized area of the present study.

LTS-WCA is a weighted clustering algorithm which is designed in this work specifically for distributed heterogeneous Wireless Sensor Network. The algorithm includes two phases: *clustering* and *network maintenance*. It employs five key parameters in order to choose the best cluster head through the network environment. These parameters are transmission range of a node (T_r), minimum distance to a neighbor cluster's cluster head (D_{min}), speed of a node (M_v), degree of a node (d_v), remaining energy of a node (E_r) and number of nodes that a node can handle inside of its cluster in case it becomes a cluster head (S). After choosing cluster heads and grouping the network nodes in clusters, the maintenance phase starts. In the maintenance phase, three parameters are checked periodically within the network environment: the residual energy of mobile wireless sensor nodes, the mobility of sensor nodes and the amount of load put on a cluster head.

The main purpose of LTS-WCA is to overcome the problems which a Wireless Sensor Network faces. LTS-WCA increases network life time by decreasing the number of clusters within the network environment. Decreasing the number of clusters leads to less usage of transmission power and finally keeping the nodes alive for much longer within the network environment. Moreover decreasing the time needed to group the network into clusters also in

increasing the network life time and LTS-WCA acts successfully to increase the overall network life time on a Wireless Sensor Network.

One of the advantages of LTS-WCA is that it is applicable to MANET and homogenous networks also. As a result, as shown in our simulation study, it has a much better performance in terms of energy efficiency in comparison with existing Weighted Clustering Algorithms on both MANET and WSN such as WCA [5], WBACA [6] and CFL [18].

In terms of increasing energy efficiency and network life time, there is still a lot of work to be done. There are several parameters such as 'transmission range', 'number of neighbors', 'degree differences', and 'remaining battery power' and 'distances with neighbors' which play significant roles in the process of selecting cluster-heads and clustering formations, and these parameters should be thoroughly worked out and developed further. There is still lack of research done in this area and scant written materials covering the aforementioned issues.

Further improvements on weighted clustering algorithms should concentrate on clustering formation and cluster-heads election for creating a more stable network structure with less energy cost. In order to maintain the network, efficient thresholds should be used in terms of energy amount of nodes, mobility of nodes and cluster size; this should be done in order to decrease the number of re-affiliations as well as the number of re-clustering the network domain. Replacing some parameters for calculating the combined weight with some other parameters may help to keep the amount of load on the cluster-head balance and decrease the general overhead within the network.

REFERENCES

- [1] Na Li ,Nan Zhang, Sajal K. Das a, Bhavani Thuraisingham, “Privacy preservation in wireless sensor networks: A state-of-the-art survey”, Elsevier Journal, Ad Hoc Networks7 1501–1514, April 2009.
- [2] D. J. Dechene, A. El Jardali, M. Luccini, and A. Sauer, “Algorithms for Node Clustering in Wireless Sensor Networks: A Survey”, IEEE magazine, vol., no., pp.295-300,Dec 2008.
- [3] Burrell, J., Brooke, T., and Beckwith, R. “Vineyard computing: Sensor networks in agricultural production”, IEEE Pervasive Computing, vol.3, no.1, pp. 38- 45, Jan.-March 2004
- [4] Vandana Jindal, A.K.Verma, Seema Bawa, “How the two Adhoc networks can be different: MANET & WSNs”, International Journal of Computer Science & Technology, IJCST Vol. 2, Issue 4, Oct . - Dec. 2011.
- [5] Chatterjee, M.; Sas, S.K.; Turgut, D.; , "An on-demand weighted clustering algorithm (WCA) for ad hoc networks," Global Telecommunications Conference, 2000. GLOBECOM '00. IEEE , vol.3, no., pp.1697-1701 vol.3, 2000
- [6] Dhurandher, S.K.; Singh, G.V.; , "Weight based adaptive clustering in wireless ad hoc networks," Personal Wireless Communications, 2005. ICPWC 2005. 2005 IEEE International Conference on , vol., no., pp. 95- 100, 23-25 Jan. 2005
- [7] Bricard-Vieu, V.; Nasser, N.; , "WSN16-1: A Weighted Clustering Algorithm Using Local Cluster-heads Election for QoS in MANETs," Global Telecommunications Conference, 2006. GLOBECOM '06. IEEE , vol., no., pp.1-5, Nov. 27 2006-Dec. 1 2006

[8] Yu-Xuan Wang; Bao, F.S.; , "An Entropy-Based Weighted Clustering Algorithm and Its Optimization for Ad Hoc Networks," *Wireless and Mobile Computing, Networking and Communications*, 2007. WiMOB 2007. Third IEEE International Conference on , vol., no., pp.56, 8-10 Oct. 2007

[9] Yoon-cheol Hwang; Yoon-Su Jeong; Sang-Ho Lee; Jeong-Young Song; Jin-II Kim; , "Advanced Efficiency and Stability Combined Weight Based Distributed Clustering Algorithm in MANET," *Future Generation Communication and Networking (FGCN 2007)* , vol.2, no., pp.478-483, 6-8 Dec. 2007

[10] Yi Wang; Hairong Chen; Xinyu Yang; Deyun Zhang; , "WACHM: Weight based adaptive clustering for large scale heterogeneous MANET," *Communications and Information Technologies*, 2007. ISCIT '07. International Symposium on , vol., no., pp.936-941, 17-19 Oct. 2007

[11] Safa, H.; Mirza, O.; Artail, H.; , "A Dynamic Energy Efficient Clustering Algorithm for MANETs," *Networking and Communications*, 2008. WIMOB '08. IEEE International Conference on Wireless and Mobile Computing, , vol., no., pp.51-56, 12-14 Oct. 2008

[12] Likun Zou; Qishan Zhang; Jianwei Liu; , "An Improved Weight-Based Clustering Algorithm in MANETs," *Wireless Communications, Networking and Mobile Computing*, 2008. WiCOM '08. 4th International Conference on , vol., no., pp.1-4, 12-14 Oct. 2008

[13] Bouk, S.H.; Sasase, I.; , "Energy Efficient and Stable Weight Based Clustering for mobile ad hoc networks," *Signal Processing and Communication Systems*, 2008. ICSPCS 2008. 2nd International Conference on , vol., no., pp.1-10, 15-17 Dec. 2008

[14] Hussein, A.H.; Abu Salem, A.O.; Yousef, S.; , "A flexible weighted clustering algorithm based on battery power for Mobile Ad hoc Networks," *Industrial Electronics*, 2008. ISIE 2008. IEEE International Symposium on , vol., no., pp.2102-2107, June 30 2008-July 2 2008

[15] Chang Li; Yafeng Wang; Fan Huang; Dacheng Yang; , "A Novel Enhanced Weighted Clustering Algorithm for Mobile Networks," *Wireless Communications*,

Networking and Mobile Computing, 2009. WiCom '09. 5th International Conference on , vol., no., pp.1-4, 24-26 Sept. 2009

[16] Fayyaz, Y.; Nasim, M.; Javed, M.Y.; , "Maximal Weight Topology Discovery in Ad hoc Wireless Sensor Networks," Computer and Information Technology (CIT), 2010 IEEE 10th International Conference on , vol., no., pp.715-722, June 29 2010-July 1 2010

[17] Sivaprakasam, P.; Gunavathi, R.; , "An efficient clusterhead election algorithm based on maximum weight for MANET," Advanced Computing (ICoAC), 2011 Third International Conference on , vol., no., pp.315-320, 14-16 Dec. 2011

[18] Zainalie, S.; Yaghmaee, M.H.; , "CFL: A clustering algorithm for localization in Wireless Sensor Networks," Telecommunications, 2008. IST 2008. International Symposium on , vol., no., pp.435-439, 27-28 Aug. 2008

[19] Tzung-Pei Hong, "An Improved Weighted Clustering Algorithm for Determination of Application Nodes in Heterogeneous Sensor Networks", Journal of Information Hiding and Multimedia Signal Processing, Volume 2, Number 2, April 2011

[20] Ping Ding , Joanne Holliday , Aslihan Celik, "Distributed energy-efficient hierarchical clustering for wireless sensor networks", Proceeding DCOSS'05 Proceedings of the First IEEE international conference on Distributed Computing in Sensor Systems, Pages 322-339, 2005

[21] <http://www.scribd.com/doc/50634760/MANET-vs-WSN>

APPENDIX A



TEZ FOTOKOPİ İZİN FORMU

ENSTİTÜ

Fen Bilimleri Enstitüsü

Sosyal Bilimler Enstitüsü

Uygulamalı Matematik Enstitüsü

Enformatik Enstitüsü

Deniz Bilimleri Enstitüsü

YAZARIN

Soyadı :

Adı :

Bölümü :

TEZİN ADI (İngilizce).....

.....

TEZİN TÜRÜ : Yüksek Lisans

Doktora

1. Tezimin tamamı dünya çapında erişime açılsın ve kaynak gösterilmek şartıyla tezimin

bir kısmı veya tamamının fotokopisi alınsın.

2. Tezimin tamamı yalnızca Orta Doğu Teknik Üniversitesi kullanıcılarının erişimine açılsın.

(Bu seçenekle tezinizin fotokopisi ya da elektronik kopyası Kütüphane aracılığı ile ODTÜ

dışına dağıtılmayacaktır.)

3. Tezim bir (1) yıl süreyle erişime kapalı olsun. (Bu seçenekle tezinizin fotokopisi ya da

elektronik kopyası Kütüphane aracılığı ile ODTÜ dışına dağıtılmayacaktır.)

Yazarın imzası

Tarih