IJACSA

WHERE WISDOM SHARES

International Journal of Advanced Computer Science and Applications

Special Issue

Special Issue on
Advances in Vehicular Ad Hoc Networking and Applications

ISSN 2156-5570(Online)
ISSN 2158-107X(Print)

SAI

www.ijacsa.thesai.org

IJACSA

WHERE WISDOM SHARES

INTERNATIONAL JOURNAL OF ADVANCED COMPUTER SCIENCE AND APPLICATIONS

SAI

THE SCIENCE AND INFORMATION ORGANIZATION

www.thesai.org | info@thesai.org

Editorial Preface

From the Desk of Managing Editor…

In the last decade, we have witnessed a big increase in the use of wireless technologies in different type of network, such as ad hoc sensor networks and vehicular ad hoc networks. Ubiquitous access to the information anywhere, anytime, from any device by end-users continue to conduct the need to develop innovative architectures and protocols for ad hoc networks with capabilities that can help achieve this goal. The birth of mobile networks like the VANETs has opened up many research challenges that need to be treating to enable end-to-end communication transparently over a highly heterogeneous network. In particular VANET networks (Vehicular Ad hoc NETworks) are a new form of mobile ad hoc networks used to establish communications between vehicles or with infrastructure located at roadsides. These networks are used to meet the needs of communication applied to transportation networks to improve the driving and road safety for road users.

Applications related to road safety is an important part of VANET applications and include the dissemination of messages on the state of traffic , the road conditions , accidents or messages reminding limitations speed and safety distances. Services deployed in VANETs are not limited only to road safety applications but other types of applications that allow the dissemination of practical information by providers of services to drivers of cars like the location of available parking places.

The VANETs are characterized by high mobility, related to the speed of the cars, which is important on highways. Therefore, a car can quickly join or leave the network in a very short time, which makes topology changes very frequently. Nevertheless, the absence of a central management of network functionality creates other constraints such as channel access, routing and data dissemination, self- organization, addressing or security of the network.

This special issue is dedicated to original results and achievements by active researchers, designers, and developers working on various issues and challenges related to wireless networks. After a rigorous peer review process we accepted four papers that cover different topics of the special issue.

In the paper entitled “The impact of Black-Hole Attack on AODV protocol” the authors presented the black hole attack and its impact on a network that uses the AODV protocol followed by simulation compared with AODV in normal situation.

In the paper entitled “Traffic Signs Detection and Recognition Using Neural Network on OpenCV library” a system of Traffic Signs Detection and Recognition based on image processing using OpenCV library and Multilayer Perceptron (MLP) algorithm was presented.

The authors describe in the first section the detection module which is based on image processing such as color segmentation, threshold technique, Gaussian filter, canny edge detection, Contour and contours detection based on OpenCV library.

The authors propose in the second section a Recognition Module which is based on neural network, in this part a pre-processing stage was presented in order to reduce the amount of information to process, and reduce the computing time cost of the system.

The entitled paper “Investigation of Time Slots corresponds to a node dies in LEACH protocol on Wireless Sensor Network“ present some idea to resolve the life time problem, that’s due to the fact that when a battery node dies in WSN, it becomes useless. The authors propose the use of Hierarchical routing protocols to maximize the network life time combined with the LEACH protocol that is one of the fundamental protocols that can be used for decreasing the energy consumed in aggregating and sending the data.

In the paper entitled "A comparative study of decision tree ID3 and C4.5" The authors have started with a presentation of the two algorithms ID3 and C4.5 were they focus on the key elements of the construction of these decision tree and finally they have completed their work by a comparison of these two algorithms according to several criteria.

We would like to take this opportunity to thank the Editor-in-Chief of the International Journal of Advanced Computer Science and Applications (IJACSA), Professor Dr Kohei Arai, for his invaluable support and encouragements throughout the preparation of this special issue. We thank the staff at IJACSA for their kind help.

We express our deepest gratitude to all reviewers who devoted much of their precious time reviewing all the papers submitted to this special issue. Their timely reviews greatly helped us select the best papers included in this issue. We also thank all authors who contributed to this special issue.

Finally, we hope you will enjoy reading this selection of papers as we did and you will find this issue informative and helpful in keeping yourselves up-to-date in the fast changing fields of wireless network.

Thank you for Sharing Wisdom!

Mohammed ERRITALI
TIAD Laboratory, Computer Sciences Department, Faculty of Sciences and Technologies
Sultan Moulay Slimane University. Beni-Mellal, BP: 523, Morocco
E-mail: m.erritali@usms.ma

Guest Editor

IJACSA Special Issue on Advances in Vehicular Ad Hoc Networking and Applications
ISSN 2156-5570 (Online)
ISSN 2158-107X (Print)

Editorial Board

CONTENTS

Robust Automatic Traffic Signs Recognition Using Fast Polygonal Approximation of Digital Curves and Neural Network

Abderrahim SALHI [1], Brahim MINAOUI [2], Mohamed FAKIR [3]
Information Processing and Decision Laboratory, Sultan Moulay Slimane University,

***Abstract*—Traffic Sign Detection and Recognition (TSDR) has many features help the driver in improving the safety and comfort, today it is widely used in the automotive manufacturing sector, a robust detection and recognition system a good solution for driver assistance systems, it can warn the driver and control or prohibit certain actions which significantly increase driving safety and comfort. This paper presents a study to design, implement and test a method of detection and recognition of road signs based on computer vision. The approach adopted in this work consists of two main modules: a detection module which is based on color segmentation and edge detection to identify areas of the scene may contain road signs and a recognition module based on the multilayer perceptrons whose role is to match the patterns detected with road signs corresponding visual information. The development of these two modules is performed using the C/C++ language and the OpenCV library. The tests are performed on a set of real images of traffic to show the performance of the system being developed.**

Keywords—Traffic sign; recognition; detection; pattern matching; image processing; Polygonal Approximation of digital curves

I. Introduction

A driver assistance system is designed to help the driver to better control the vehicle in difficult circumstances (tired, stress, carelessness, poor vision ...) to help increase safety and the safety of other drivers, pedestrians etc. involved in the traffic on the roads.

Computer vision is a promising approach for addressing this problem. Automatic control of the braking system, automatic speed control, generation of alerts and notifications, corresponding to the various events encountered on the road etc., are some examples among many installations of driver assistance that could be developed, based on this approach.

The aim of this work is to design, implement and test a system of Traffic signs Recognition.

Traffic signs Recognition system developed in this work like [1-3] composed of two main Modules. The first Module is the detection Module which is designed to detect and extract zones may contain traffic signs in the image, based on the particular color and geometry of these panels; and using image processing techniques such as color segmentation, threshold technique, Gaussian filter, canny edge detection, it allows us to greatly reduce the amount of information to be processed. This allows faster processing and facilitating the realization of a system operating in real time.

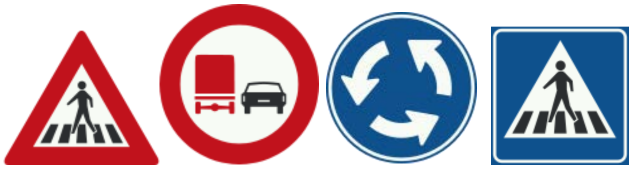


Fig. 1. Examples of the four types of panels considered in this work

The second Module is the recognition Module; it helps identify road signs by comparing information provided by the detection Module with models of road signs that have been learned beforehand, based multilayer perceptrons.

The algorithms are designed to detect and recognize red triangular and circular panels, and blue circular and rectangular panels. Making this choice of traffic signs to recognize, it covers a vast majority of common traffic signs and also the most important signals that advertise a danger on the road, a ban (speed limit) or require a driver to certain behavior (blue signals) figure (FIG 1). Shows some examples of signs used in this work.

II. System Description

The first section in our system (FIG 2) is the detection module, in this part we read the RGB image from real images sequence and convert color to HSV space, to minimize the effects of changing brightness, followed by a color segmentation using threshold [5], two masks are generated, the first one, is designed to find zones which may be a red circles or a red triangles shapes, using shape recognition [8], the second mask, is to find zones which may be a bleu circles or a bleu rectangles shapes by the same way.

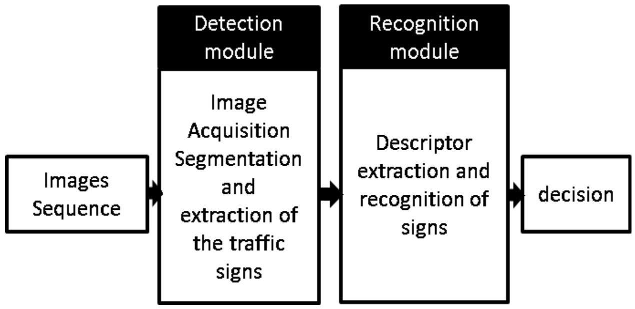
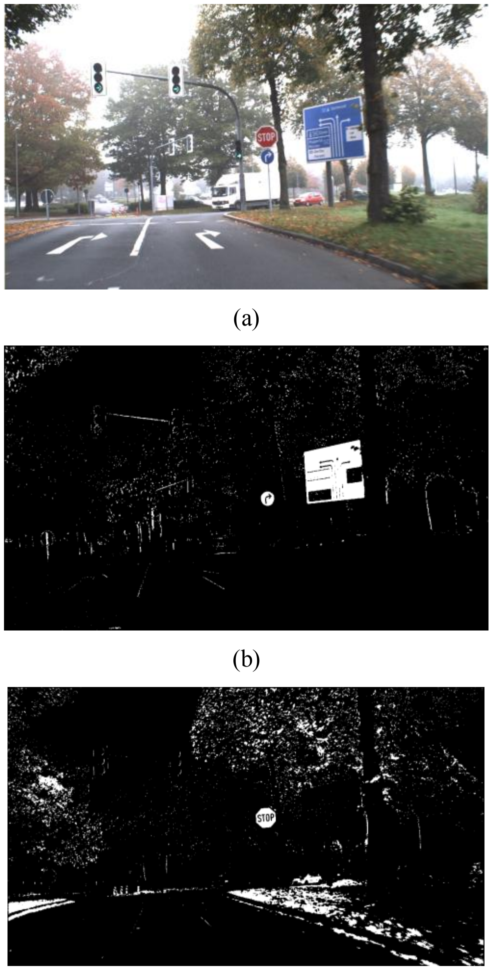

Detection module
Image Acquisition Segmentation and extraction of the traffic signs
Recognition module
Descriptor extraction and recognition of signs
Images Sequence
decision

Fig. 2. Block diagram of the system

The second section is the Recognition module, in this part we extract the descriptors of zones (blobs) provided by the detection module, to be matched with templates in database [9] using Multilayer Perceptron (MLP) algorithm, and take the appropriate decision.

III. Trafic Signs Detection Module



(a)

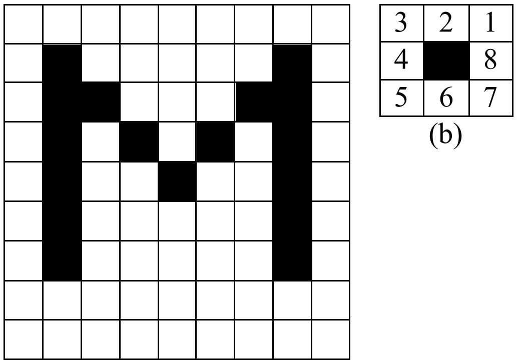
(b)

(c)

Fig. 3. a) BGR color image; b) thresholded Blue mask, c) thresholded Red mask.

The image is read in RGB (Red Green Blue) color mode (FIG 3: a), and then converted to HSV (Hue Saturation Value) color space, then we use the thresholding technique to generate two red mask and blue (FIG 3: a & FIG 3: b), these masks will be used later to find the contours, and before that, we apply the technique of smoothing by using a Gaussian filter, and the technique of Canny edge detection to improve the image and easily get the desired region.

The obtained binary image for each mask is processed to retrieve contours by `findContours` function for binary image and the retrieved contours are returned and stocked in chain format.



(b)

(a) Chain code:
2222255533366666

Fig. 4. Coding of Connected Components (Courtesy [10])

To represent contours in the OpenCV library, we use the Freeman method or the chain code. For any pixel all its neighbors with numbers from 0 to 7 can be enumerated as Fig. 4(a). The 0-neighbor denotes the pixel on the right side, etc.

As a sequence of 8-connected points, the border can be stored as the coordinates of the initial point, followed by codes (from 0 to 7) that specify the location of the next point relative to the current one. (Fig. 4 (b) illustrates example of Freeman coding of connected components.)

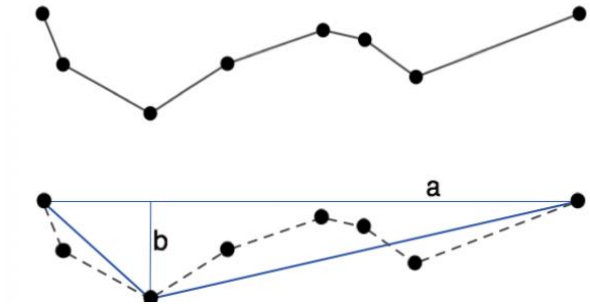


Fig. 5. Simplifying a piecewise linear curve with the Ramer-Douglas–Peucker algorithm [11].

The extracted contours are used to find the edges that can match the shapes of traffic signs, like triangular shapes, circular shapes and rectangular shapes. In the contours returned from the red mask, we will look for the triangular and circular shapes, and in those returned from blue mask, we will search for rectangular and circular shapes.

To find these shapes will be based on the algorithm of Ramer -Douglas – Peucker [11]. The idea is to simplify a polyline (n nodes) and replace it with a single line (two-node) if the distance of the farthest from the line formed by the ends of the polyline node is below a threshold, as shown in (FIG 5) The algorithm works recursively by the method of "divide and rule", to initialize the algorithm we select the first and last node (for a polyline), or any node (such as a polygon). These are the terminals. At each step, through all the nodes between the terminals and the farthest segment formed by the terminal node is selected.

- If there is no node between terminals algorithm ends.
- If this distance is less than a certain threshold are removed all the nodes between terminals.
- If it exceeds the polyline is cannot be directly simplified. Called recursively the algorithm on two sub-parts of the polyline: the first terminal to the remote node and the remote terminal to the final node.

The approxpolyDP function provided by the OpenCV library [10] is used to implement this algorithm, and takes as input the contours found by findContours function, and the threshold, it return a table of points forming the new polygon approached to the real contour.

Our algorithm tests the size of the returned table of points, and if it contains tree elements it means that the founded shape is a triangle, if it contains four elements, it means that the shape founded corresponding to a rectangle.

For the circular shapes we empirically take the size of the table from beyond six elements.

To close that founded shapes to those of traffic signs, we impose certain criteria, such as: triangles should be equilateral with an error near; their surfaces must be included in a well-defined interval, and the something for rectangles.

For circular shapes, we use the ellipse detection technique [1] provided by OpenCV library [10] and then the sizes of its two axes are taken as criteria. The shapes that satisfy the criteria imposed will be cropped and extracted from the color natural image; these blobs will be resized to 32x32 pixels to be used in the recognition phase.

This allows us to significantly reduce the number of shapes processed and computing time cost of the algorithm later; Figure (FIG 6) shows the flowchart of the detection module.

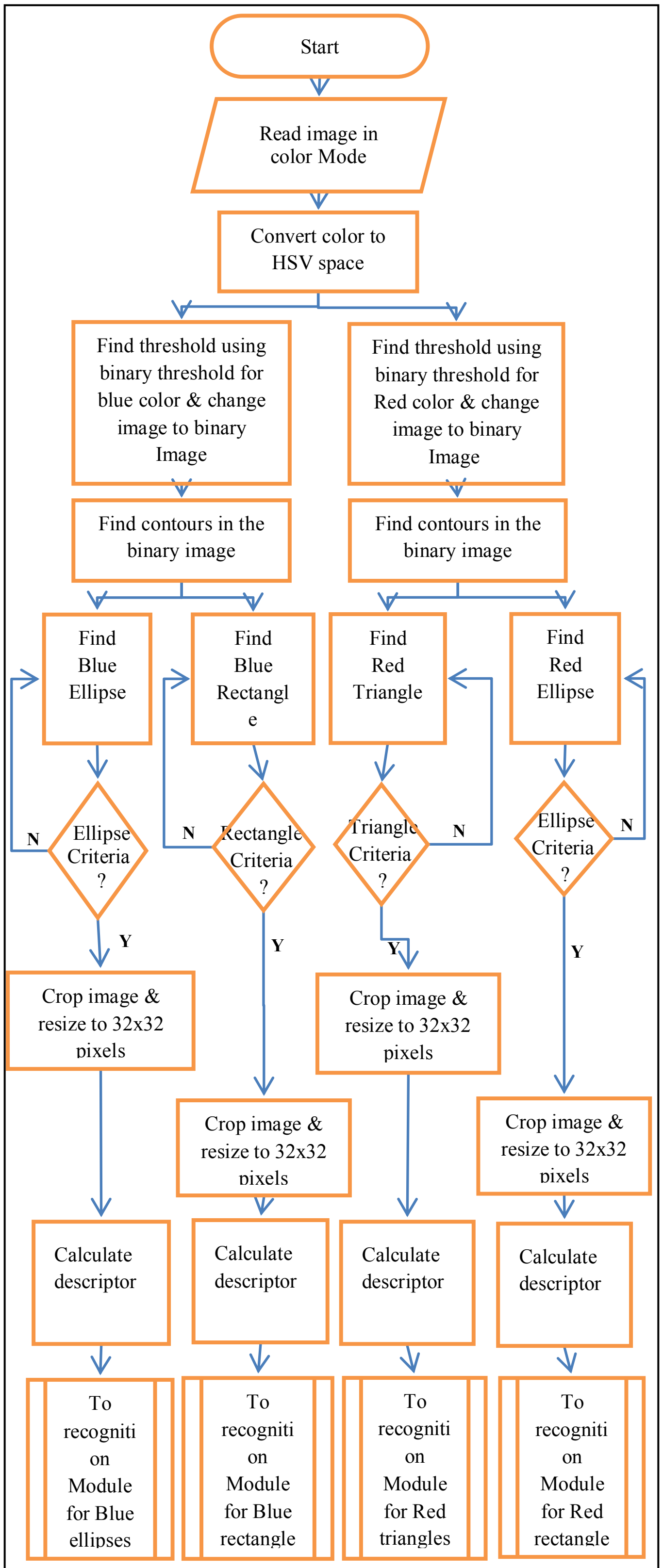
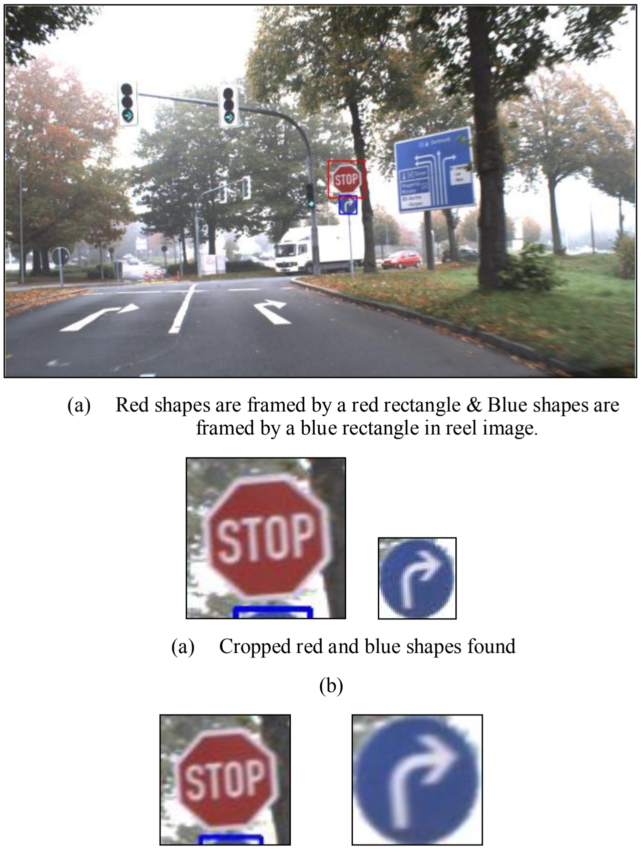

Start
Read image in color Mode
Convert color to HSV space
Find threshold using binary threshold for blue color & change image to binary Image
Find threshold using binary threshold for Red color & change image to binary Image
Find contours in the binary image
Find contours in the binary image
Find Blue Ellipse
Find Blue Rectangl e
Find Red Triangle
Find Red Ellipse
Ellipse Criteria ?
Rectangle Criteria ?
Triangle Criteria ?
Ellipse Criteria ?
N
N
N
N
Y
Y
Y
Y
Crop image & resize to 32x32 pixels
Crop image & resize to 32x32 pixels
Crop image & resize to 32x32 pixels
Crop image & resize to 32x32 pixels
Calculate descriptor
Calculate descriptor
Calculate descriptor
Calculate descriptor
To recogniti on Module for Blue ellipses
To recogniti on Module for Blue rectangle
To recogniti on Module for Red triangles
To recogniti on Module for Red rectangle

Fig. 6. flowchart of the detection module



(a) Red shapes are framed by a red rectangle & Blue shapes are framed by a blue rectangle in reel image.

(a) Cropped red and blue shapes found

(b)

(c) Resized shapes to 32x32 pixels.

Fig. 7. Result of traffic signs Detection Module.

The traffic signs images database [9] used in this paper contains 300 color images with natural background, and with 1300x800 pixels size. Our program is tested on all these images, the figure (FIG 7) shows one result of traffic signs Detection Module.

The program may extract also some blobs which are not a traffic signs, but all these blobs will be presented as input to Traffic Signs Recognition Module to decide which the right traffic sign is, and which the bad is.

Figure (FIG 8 & 9) shows respectively the number of blobs extracted from the first 100 images and the time spent in processing each image by Traffic Signs Recognition Module.

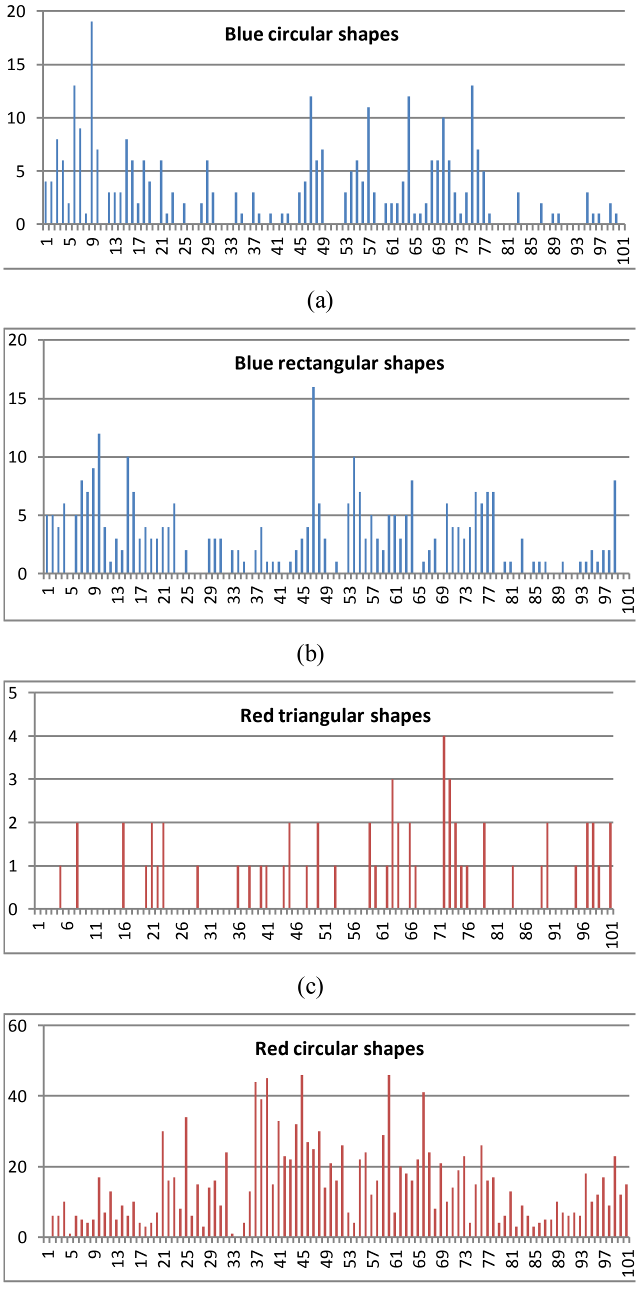
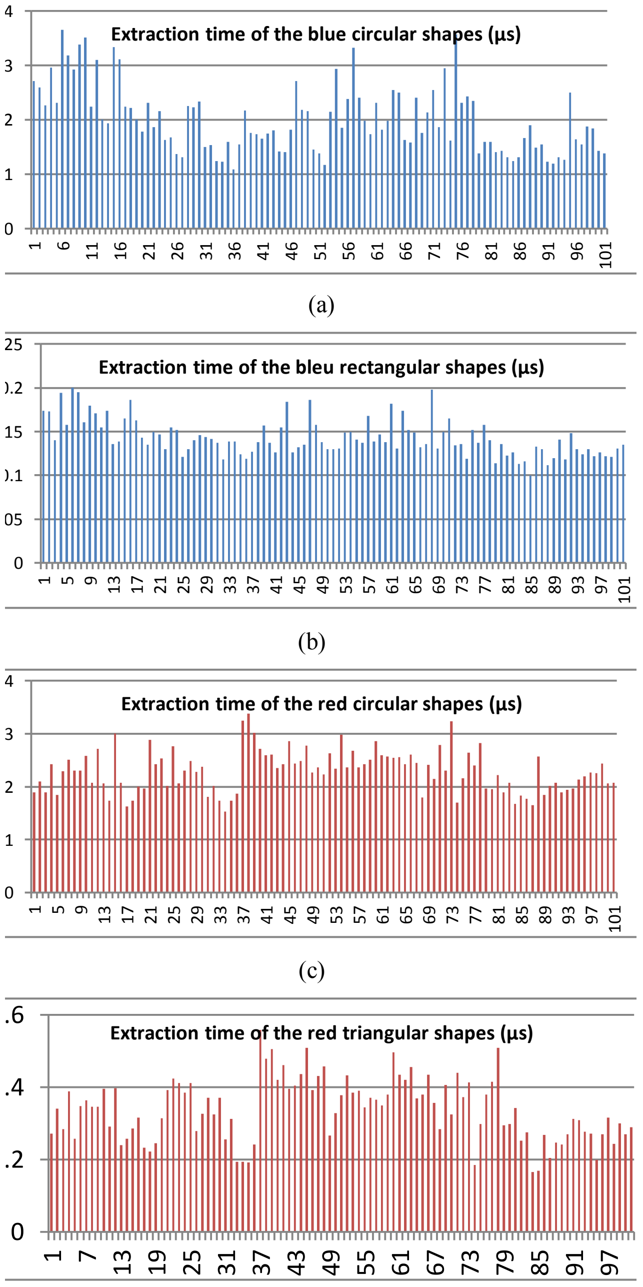


Fig. 8. Number of blobs extracted from the first 100 images



(a)

(b)

(c)

(d)

Fig. 9. Time spent in processing each image by Traffic Signs Detection and Recognition Module

IV. Traffic Signs Recogniation Module

A. *Preprocessing Stage.*

Before passing the extracted blobs to the Traffic Signs Recognition Module which is based on neural network, a preprocessing stage was required to reduce the amount of information to process, and reduce the computing time cost of the system thereafter. Through this stage, we proceed by extracting the descriptor of each blob.

Blobs are presented in the 32x32 size, which is 1024 pixels, and for the three layers R, G and B, the size of neural network input vector will be 3072, and this will delay the system. For each channel, a projection of the pixels is done on both vertical and horizontal axes.

So for each point CX_i on the horizontal axis:

$$CX_i = \frac{1}{255} * \frac{1}{32} \sum_{j=1}^{32} C_{i,j} \quad \textit{i=1, 2... 32} \qquad (1)$$

And for each point CY_i on the vertical axis:

$$CY_j = \frac{1}{255} * \frac{1}{32} \sum_{i=1}^{32} C_{i,j} \quad \textit{j=1, 2... 32} \qquad (2)$$

- $C_{i,j}$ Is the intensity of the pixel whose coordinates are (i, j) of the layer C.
- C Is the red layer R, green layer G or blue layer B.
- CX_i And CY_i values are between 0 and 1.

The new descriptor is composed of 192 elements, the 32 elements ofCX_i, and 32 others of CY_i for the three layers RGB.

B. *tarffic signs recognition CORE*

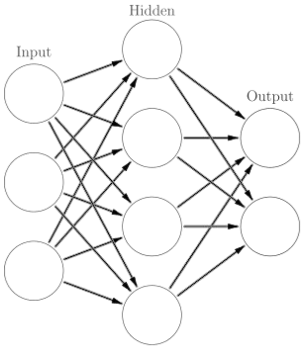


Fig. 10. MLP Structure

The Traffic Signs Recognition Module is implemented by using feed-forward artificial neural networks or, more particularly, multi-layer perceptrons (MLP), the most used type of Artificial neural networks [12]. These are a mathematical model, inspired by the brain that is often used in machine learning. It was initially proposed in the '40s and there was some interest initially, but it waned soon due to the inefficient training algorithms used and the lack of computing power. More recently however they have started to be used again, especially since the introduction of autoencoders, convolutional nets, dropout regularization and other techniques that improve their performance significantly.

The MLP includes at least 3 layers. The first one is the input layer; the last one is the output layer, and one or more hidden layers. Each layer of MLP contains one or more neurons directionally linked with the neurons from the previous and the next layer. The figure (FIG 10) represents an example of a 3-layer perceptron with three inputs, two outputs, and the hidden layer including four neurons. All the neurons in MLP are similar. Each of them has several input links (it takes the output values from several neurons in the previous layer as input) and several output links (it passes the response to several neurons in the next layer).

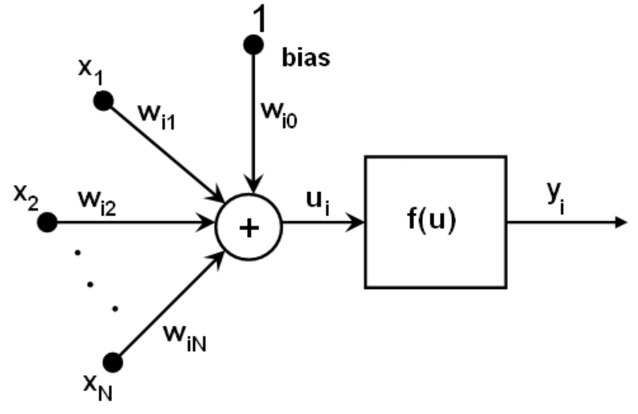


Fig. 11. Computing process in a neuron

The values retrieved from the previous layer are summed up with certain weights, individual for each neuron, plus the bias term. The sum is transformed using the activation function f that may be also different for different neurons (FIG 11) In other words, given the outputs x_j of the layer n, the outputs y_i of the layer $n+1$ are computed as:

$$u_i = \sum_j \left(w_{i,j}^{n+1} * x_j\right) + w_{i,bias}^{n+1} \qquad (3)$$

$$y_j = f(u_i) \qquad (4)$$

The activation function that used in this paper is binary sigmoid function, which is defined as:

$$f(x) = \beta * \frac{1-e^{-\alpha x}}{1+e^{-\alpha x}} \qquad (5)$$

With $\alpha = 1\ et\ \beta = 1$, it is shown like illustrate (FIG 12); the sigmoidal function basically has some very useful mathematical properties, monotonicity and continuity. Monotonicity means that the function f(x) either always increases or always decreases as x increases. Moreover, continuity means there are no breaks in the function, it is smooth.

These parameters are intrinsic properties eventually assist networks power to approximate and generalize on functions by learning from data.

The Traffic Signs module used in this paper is separated to four MLP functions; each one is used for one type of blobs, (blue circular blobs, blue rectangular blobs, red circular blobs and red triangular blobs).

The results of Traffic Signs Recognition Module are presented in table I, II and III below.

TABLE I. RESULTS OF RECOGNIZED RED TRIANGULAR TRAFFIC SIGNS

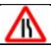
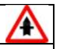
					
Number of tested images	30	5	3	7	12
Number of Recognized Signs	26	4	2	6	10
Recognition rate (%)	87	80	67	85	83

TABLE II. RESULTS OF RECOGNIZED RED CIRCULAR TRAFFIC SIGNS

Number of tested images	33	18	36	14	10	9	23	11
Number of recognized signs	30	15	32	10	10	8	20	11
Recognition rate (%)	90	83	89	71	100	89	87	100

TABLE III. RESULTS OF RECOGNIZED BLUE CIRCULAR TRAFFIC SIGNS

Number of images	29	5	6	5	7	11	18	5
Number of recognized sign	25	4	4	4	6	9	16	3
Recognition rate (%)	86	80	67	80	86	82	89	60

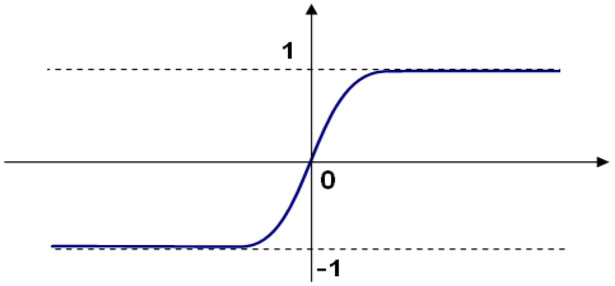


Fig. 12. sigmoid function representation with $\alpha = 1\, et\, \beta = 1$

V. Perspectives

Tests show that the Detection Module currently implemented is very fast and good enough (detection rate 99%), even if it's depending on size of images captured. by cons, the Recognition Module is much less efficient (71%). To improve performance of the proposed system, an improvement of the Recognition Module is required, by using or combining other recognition techniques.

We can also use the techniques of multithreading, which allows parallel execution of several programs at the same time, this will significantly improve the execution time of the system which will be more robust and efficient for a real time implementation.

VI. Conclusion

Traffic Sign Recognition is discussed in this study, by using Neural Networks technique. In the first time the Traffic signs Module has processed the input images using image processing techniques, such as, threshold technique, Gaussian filter, canny edge detection, Contours, algorithm of Ramer -Douglas - Peucker and Fit Ellipse. Next the Traffic Signs Recognition based one the Neural Networks, were performed to recognize the traffic sign patterns. The main reason to select this method is to reduce the computational cost inorder to facilitate the real time implementation. The strategy is to reduce number of MLP inputs by pre-processing blobs before giving them to Traffic Sign Recognition Module

The system was ended with a Traffic Sign Recognition Module based on Neural Networks, which were trained, and validated with a database containing more than 300 real images, with complex natural background, to find the best network architecture.

References

[1] A. Lorsakul and J. Suthakorn, "Traffic Sign Recognition for Intelligent Vehicle/Driver Assistance System Using Neural Network on OpenCV", The 4th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI 2007)

[2] R. Belaroussi — J-P. Tarel, " Détection des panneaux de signalisationroutière par accumulation bivariée",Traitement du Signal 27, 3 (2010) pp 265-296- DOI : 10.3166/ts.27.265-296 .

[3] F.A. Aly and A.E. Alaa. Detection, categorization andrecognition of road signs for autonomous navigation.In Proceeding of Advanced Concepts for Intelligent Vision System, Brussels, Belgium, Aug 2004.

[4] F.A. Aly and A.E. Alaa. Detection, categorization and recognition of road signs for autonomous navigation.In Proceeding of Advanced Concepts for IntelligentVision System, Brussels, Belgium, Aug 2004.

[5] H. X. Liu, and B. Ran, "Vision-Based Stop Sign Detection and Recognition System for Intelligent Vehicle", Transportation Research Board (TRB) Annual Meeting 2001, Washington, D.C., USA, January 11, 2001.

[6] H. Fleyeh, and M. Dougherty, "Road And Traffic Sign Detection And Recognition", Proceedings of the 16th Mini - EURO Conference and 10th Meeting of EWGT, pp. 644-653.

[7] [4] Arturo de la Escalera, Luis E. Moreno, Miguel Angel Salichs, and José Maria Armingol. Road traffic sign detection and classification. IEEE Transactions on Industrial Electronics, 44(6) :848–859, Dec 1997.

[8] S. Belongie, "Shalpe Matching and Object Recognition Using Shape Contexts",IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL 24, NO 24, APRIL 2002.

[9] Dataset "The German Traffic Sign Detection Benchmark", http://benchmark.ini.rub.de/?section=gtsdb&subsection=dataset 2013

[10] Intel Corporation, "Open Source Computer Vision Library," Reference Manual, Copyright © 1999-2001, Available: www.developer.intel.com

[11] Urs Ramer, "An iterative procedure for the polygonal approximation of plane curves", Computer Graphics and Image Processing, 1(3), 244–256 (1972)

[12] Laurence Fausett, "Fundamentals of Neural Networks Architectures, Algorithms, and Applications", Prentice Hall Upper Saddle River, New Jersey 1994.

Time Slots Investment of a dead node in LEACH protocol on Wireless Sensor Network

Abedelhalim HNINI
LAVETE laboratory Mathematics and
Computer Science Dept Sciences and Tech Fac
Settat, 26000, Morocco

Abdellah EZZATI
LAVETE laboratory Mathematics and
Computer Science Dept Sciences and Tech Fac
Settat, 26000, Morocco

FIHRI Mohammed
LAVETE laboratory Mathematics and
Computer Science Dept Sciences and Tech Fac
Settat, 26000, Morocco

Abdelmajid HAJAMI
LAVETE laboratory Mathematics and
Computer Science Dept Sciences and Tech Fac
Settat, 26000, Morocco

***Abstract*—Wireless sensor network (MSN) is a wirelessly interconnected network. WSN promises a wide range of potential such as surveillance, military and civilian, to name just a few, applications. A sensor node senses the environment and delivers data to the sink. Energy saving is one of the keys to challenge network life time. LEACH protocol has been incorporated to extend network life time. This protocol forms clusters of the sensor nodes and elect one of them to become a Cluster Head (CH) to route data cluster to sink. In cluster, communication uses TDMA technique. This latter organizes transmission time (Time Slot) which corresponds to each node member of cluster. When a node dies, time slot corresponding to this node will be free. In this paper, we will present LEACH comportments after a node dies, then we will propose some ideas to invest its time slots by alive node members to maximize data reception time parameter so throughput end-to-end. This parameter is very important for real-time data. Finally, we will simulate this idea with Network Simulator (NS2) to argue for our propositions.**

Keywords—Network Clustering; routing protocol; Ad hoc Network; LEACH; WSN; Node dies

I. INTRODUCTION

Wireless Sensor Networks (WSNs) have gained booming interest in recent years. They are used in various fields: military, agriculture, meteorology and medicine… Wireless Sensor Network consists of a huge number of nodes called sensor nodes. They are deployed in a spacious area. A typical sensor node is made of 4 building blocks: power unit, communication unit, processing unit and sensing unit. Limited network life is one of the most critical inconvenience and limitations of WSNs. It's arduous to recharge or to change batteries in the battery-powered sensor nodes. That is the reason why many researchers have been incorporated to increase the network life time.

The clustered WSN compromises three types of entities: the base station, cluster head sensor node and non-cluster head sensors. Non-cluster head sensor nodes take the information and send it to the respective cluster head. This latter sends the data to the base station directly in order to resolve the life time problem, which's due to the fact that when a battery node dies, it becomes useless. Several protocols have been used for a long time. Hierarchical routing protocols are the best used protocols to maximize the network life time. LEACH protocol is one of the fundamental protocols that can be used to decrease the energy which is consumed to aggregate and send the data. Each node has a time slot of data transmission, even the dead nodes or the nodes outside the area. In our contribution, we seek to invest the free slots of the dead nodes.

Our work is organized in the following way: section 1 introduces the WSNs. Section 2 presents related works on the hierarchical routing protocol and the existing LEACH descendent. Section 3 includes the presentation of our contribution. Section 4 shows the simulation steps and demonstrates the simulation results. The last section conclues our work.

II. RELATED WORKS

LEACH protocol is a basic hierarchical protocol for the WSN. It's made up of nodes cluster. Each cluster has a special node as cluster head. This latter collects data from nodes that belong to the respective cluster and transmits it to the base station [1][2].

LEACH protocol has two phases, the set up phase and the steady state phase. In set up phase, the clusters are formed and the cluster heads are chosen. Each node decides if it will become a cluster head. The decision is made by the node selecting a stochastic number between 0 and 1. If the number is less than a threshold T(n), the node becomes a cluster head for the current round [1].

The threshold is set as:

$$\mathrm{T(n)} = \begin{cases} \dfrac{\mathrm{p}}{1 - \mathrm{p} * \left[\mathrm{r} \bmod \left(\frac{1}{\mathrm{p}}\right)\right]}, \mathrm{n} \in \mathrm{G} \\ 0, \qquad \text{otherwise} \end{cases} \tag{1}$$

Where:

- p is the probability of needed being selected as a cluster head node;

- r is the number of rounds passed;
- G is the collection of ordinary nodes mod denotes modulo operator.

In the steady state phase the cluster head is maintained and the data is transmitted between nodes. The cluster head sends all the data to the base station after receiving and aggregating it. Each cluster communicates using the TDMA technique, which divides time of communication by slots, each time slot corresponds to a node of the cluster that can deliver data in their time slot.

A node is free only when a node is out of area or dead.

A modified version of the LEACH protocol is known as the LEACH-C [2], we call it also LEACH Centralized. This protocol uses the same steady-state phase as the LEACH. However, in set-up phase each node omits information concerning its current position, which is determined using position finding system, and residual energy level to the base station.

Another extension of LEACH is the TEEN (threshold-sensitive Energy Efficient sensor Network), in which two thresholds are added for the sensed attribute: the hard threshold and the soft one. The hard threshold is the minimum value of the sensed attribute required to oblige a sensor node to activate its transmitter and transmit to the cluster head. The soft threshold emitted by the cluster head to its nodes member and the cluster configuration. All this is done to increase the life span of network [3].

Most studies have shares on LEACH protocol and its descendants interesting by life time protocol but most of these works are not interested in the freedom of time slots which corresponds to the dead node.

LEACH-V [4] is introduced to overcome the problem when a CH dies or does not have sufficient energy to transmit the received data arriving from cluster members to the base station. In this case, another member node will become a CH of the cluster. This node is called vice cluster head. This one is interested in the death of CH but it doesn't also use the free time slots corresponding to the dead node member.

The exception studies are interested in a phenomenon which is close to free time slots is presented in [5][6][7][8], all these studies meet the influence mobility of nodes in term of packet loss. But there added a lot of flow to control packet loss. Therefore they consume a lot of energy. . In addition, they took the packet loss even if a node that does not have any data to transmit, it changed the cluster or its battery is sold out.

All this protocol, when a member node (not CH) dies, the time slot reserved of this node is free or it is using with more energy . That's what we did to develop this free time to invest it and use it by other node, in a manner minimizing power consumption.

III. CONTRIBUTION

LEACH protocol divides sensor network to form clusters and randomly selects a CH (Cluster Head) in each cluster. In a cluster member node, which is not a CH, senses data and transmits it to the CH. This latter aggregates the received data and forwards these data to base station. As presented in schema.

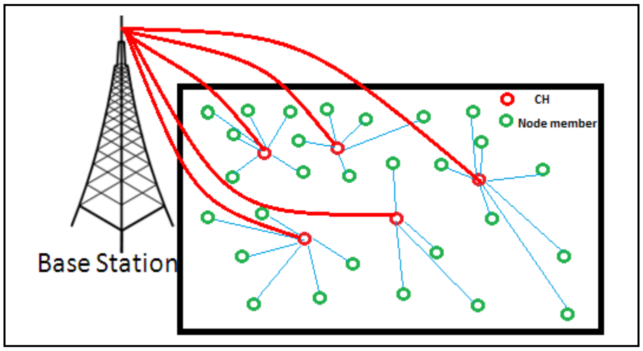


Fig. 1. Clustering in LEACH

In cluster, communication between CH and its member nodes use TDMA technique to synchronize data transmission time. TDMA divides time of communication to form frames. This latter is subdivided to form Time Slot: data transmission time corresponds to one member node. Every node has one time slot as presented in figure (figure 1).

A. The energy dissipation of the steady data transmission phase

A sensor uses its energy to realize three main activities: the acquisition, communication and data processing.

Acquisition: the energy consumed to complete the acquisition is not very important. However it depends on the phenomenon and the type of monitoring performance.

Communication: The communications consumes a lot of energy than other spots. They cover communication transmission and reception.

Treatments: the energy consumed in calculations is much lower than the energy of communications.

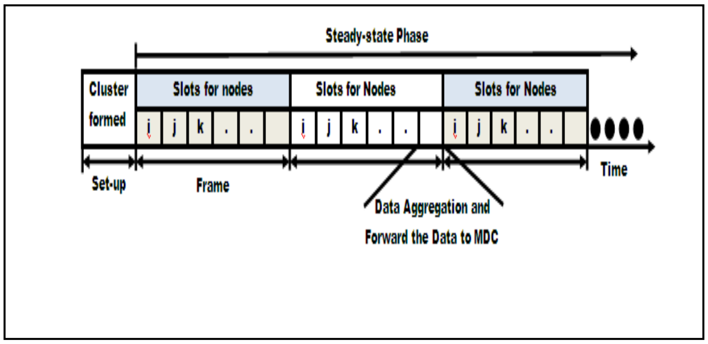


Fig. 2. chronologique communication in LEACH

In LEACH [2], to transmit a K-bit message for a distance d. the transmission consumes energy as showed below:

$ETx(K, d) = Eelec \times k + \varepsilon amp \times k \times d2$ ----------------(2)

To receive a K-bit message the reception consumes:

$ERx(K) = ERx\text{-}elec(k) = Eelec \times k$ -------------------(3)

Where Eelec: energy transmission/ reception power.

K: message size

d: the distance between transmitter and receiver.

εamp: amplification factor.

Each cluster head dissipates energy among the signals received from the nodes and transmits the aggregated signals to the BS since the BS is far from the nodes. Presumably, the energy dissipation follows (d4 power loss); therefore the energy dissipated in the cluster head node during a single frame is:

$$Eframe = N/k * (ETx(K, d) + ERx(K)) + Ech \quad \text{----- (4)}$$

$$Ech = N/k * ERx(K) + N/k * ETx(K, d2) \quad \text{------ (5)}$$

Where we assume that there are N nodes distributed uniformly in an M*M region and there are k clusters, there is an average of N/k nodes per cluster.

Each non-cluster head node needs less energy to transmit its data to the CH. Thus the energy used in each non cluster head node is:

$$EnotCH = ETx(K, d) + ERx(K) \quad \text{------------------- (6)}$$

Where d is the distance from the node to CH.

B. *The communication technique inside one cluster*

In this paper, we will do a restatement of slots. We distribute free time slot of the dead nodes to the other living nodes with regard to three different possible cases. In the first case, the free time slot will be given to the first node in the list. In the second, it will be given to the last node as CH. In the last case, in every turn, we reserve it for a member node in the cluster.

In the first case, the free time slot is reserved to the first node in cluster as showed in figure 3. We assume that node N3 is dead after Frame 1, so the organization of time slots will change according to the diagram in Figure 3. After the end of each frame CH informs the first node in cluster (N2) to lock this free time slot. Energy dissipation is added to the equation (3) as Espending to inform node N1 so (4) revert equation (7)

$$Eframe = N/k * (ETx(K, d) + ERx(K)) + Ech + Espending \quad \text{----- (7)}$$

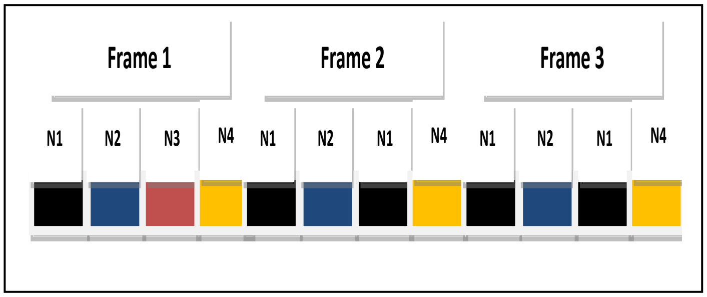


Fig. 3. The organization of the frame after the node N3 dies

In the second case, the free time slot is reserved to the last node in cluster that is the CH. So after node N3 dies, we obtain the following organization (fig 4). We get the energy Espending dissipated in (7) but CH always remains active to receive the data sent by their member nodes, so the CH dies earlier than the other nodes in the cluster.

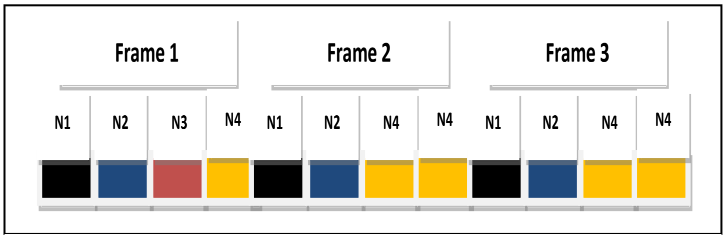


Fig. 4. The organization of free time slot to CH

In the last case, in each frame we reserve the free time slot to one node in the cluster. As presented in fig 5. In this case, all the cluster nodes will be compelled, at the end of each frame, to be active to receive the broadcast message by the CH which reorganizes the time slots again. Energy dissipation is added to the equation (3) as Espending + k/N*Eactive. So (4) change to (8)

$$Eframe = N/k * (ETx(K, d) + ERx(K)) + Ech + Espending + k/N*Eactive \quad \text{----- (8).}$$

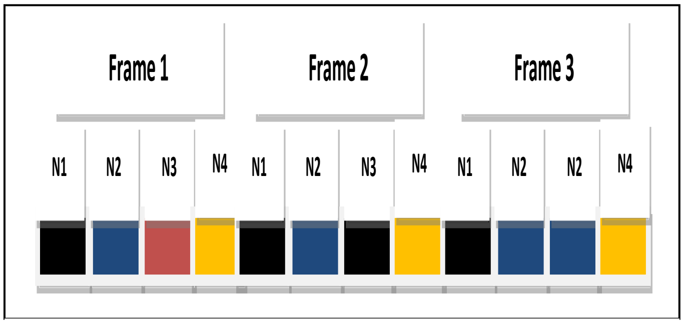


Fig. 5. In each frame free time slot reserved to one node in cluster

In all cases, we spend slightly more energy but we minimize the data time arrival, we maximize data aggregation and throughput. As a result, we maximize QoS parameter. These parameters will be presented clearly in the simulation result. WSN generates different traffic. This requires the classification of different type of services. In addition, real-time data normally demands larger bandwidth and entails higher network throughput to transport large volume of data to remote data sink rapidly and reliably. However, in this simulation we will focus on data reception rates, jitter and end-to-end reception. These parameters have a lot of benefits despite consuming a little bit more energy.

IV. SIMULATION RESULTS

We run our technique using NS-2 simulator (version 2.34) [13] with LEACH protocol [14] to determine its benefits. Then we will compare it to LEACH in terms of Throughput relative to the number of the alive nodes (data received to the BS in terms of number of the alive nodes per time), energy dissipation per time. For the experiment, we choose the following parameters: a network of 100 homogeneous nodes randomly deployed in a zone with dimensions of 1000*1000 and a base station which is located at (X=70, Y=200). For our evaluations, we use the average of 10 runs for each set of tested parameters and we neglect the case of the rotation of free time slots by living cluster nodes.

- Energy dissipation.

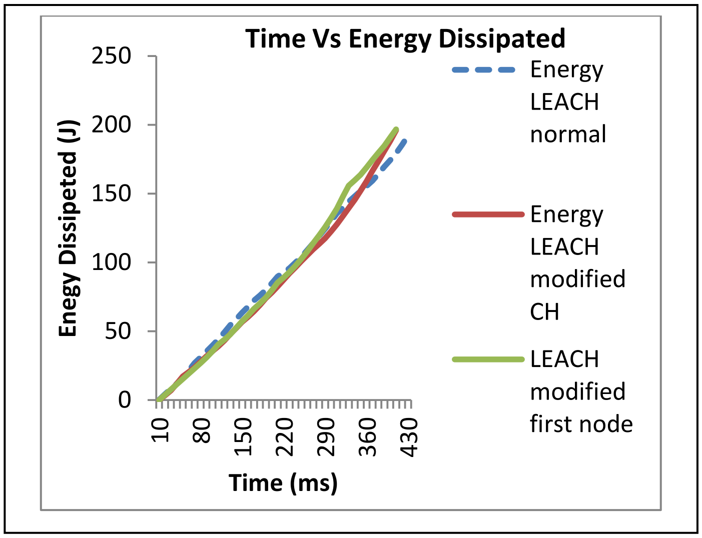


Fig. 6. Comparison of Energy Dissipated with Time

The first graph (Fig 6) in this figure depicts the energy dissipation of normal LEACH protocol and LEACH protocol with reformation of time slot. We see that after the time exceeds almost 200, the two lines {LEACH is modified with free time slots reserved to CH (LEACH modified CH), LEACH is modified with free time slot reserved to the first node in the cluster (LEACH modified first node)} are divergent of LEACH classic (LEACH normal) because it is time that the nodes begin to be exhausted which generates the free time slots that are used again later in LEACH modified by other nodes. That is why the lines of LEACH modified are superior to LEACH normal with reformation of time slots. But that is normal because we added a traffic signal by CH soon as the death of node for LEACH modified CH and the first node of threshold for LEACH modified first node. The performance is obvious at the end-to-end data aggregation depends on the alive member nodes and throughput of data reception.

- Average Throughput.

The second graph (Fig 7) shows the average throughput depending on time. In both cases, normal LEACH and LEACH with integration of our contribution, after the time exceeds almost 200, the two last lines have more throughput than the first line. This shows that the performance increases at the reception data then throughput. It is very clear in graph (Fig 8), when the nodes begin to die; the lines of LEACH modified are above the line of normal LEACH.

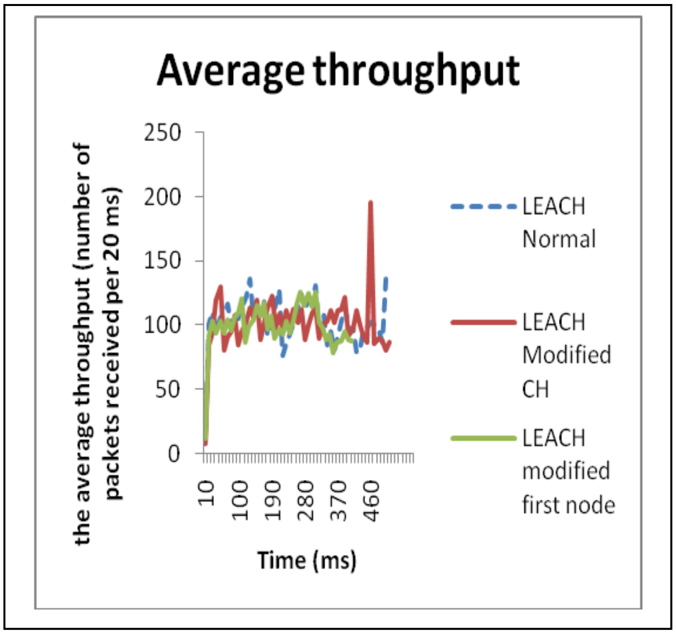


Fig. 7. The average throughput

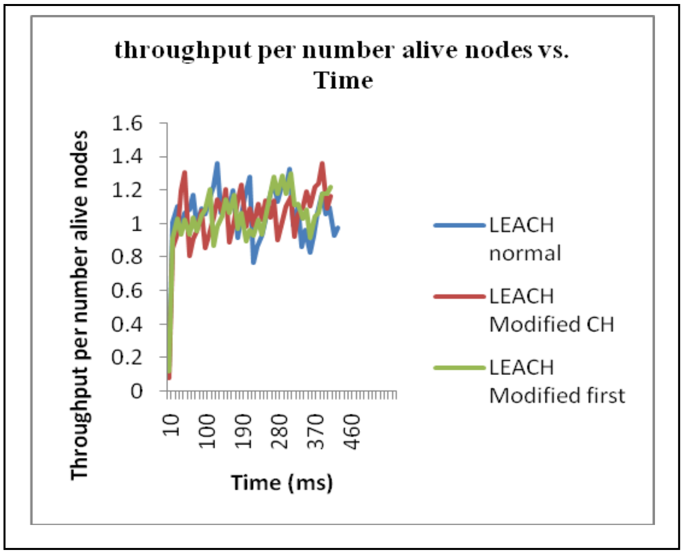


Fig. 8. . Throughput per number alive nodes

V. Conclusion

In this work, we presented the LEACH hierarchical protocol in wireless sensor network with the use of free time slots of the dead nodes. The same goal of achieving a long lifetime of wireless sensor network and a flow rate of reception elevated relative to live node.

We also saw, through our extensive simulations we tested the stability of the life of the network, however simulation results clearly show that the line corresponding to normal LEACH protocol has a less throughput than the line of LEACH protocol modified. Therefore our technique is able to extend the performance in terms of average throughput.

References

[1] R.SARAVANAKUMAR, S.G.SUSILA, J.RAJA “Energy Efficient Constant Cluster Node Scheduling Protocol for Wireless Sensor Networks”, WSEAS TRANSACTIONS on COMMUNICATIONS,pp119-128 Volume 10, April 2011.

[2] Mr. Santosh.Irappa.Shirol, Ashok Kumar, Mr. Kalmesh.M.Waderhatti, “Advanced-LEACH Protocol of Wireless Sensor Network” IJETT, Volume4 Issue6- June 2013.

[3] D. S. Kim and Y. Chung, “Self-organization routing protocol supporting mobile nodes for wireless sensor network,” in Proc. of 1st Int’l Multi-Symposium on Computer and Computational Sciences (IMSCCS’06), 2006.

[4] S. A. B. Awwad, C. K. Ng, N. K. Noordin, and M. F. A. Rasid, “Cluster based routing protocol for mobile nodes in wireless sensor network,” in Proc. of Int’l Symposium on Collaborative Technologies and Systems 2009, CTS’09, pp. 18-22, May 2009.

[5] G. Santhosh Kumar, V. Paul M V , and K. Poulose Jacob, “Mobility Metric based LEACH-Mobile Protocol” 16th International Conference on Advanced Computing and Communications, ADCOM pp. 248 – 253, 2008.

[6] Ravneet Kaur1 , Deepika Sharma2 and Navdeep Kaur3, “Comparative Analysis Of Leach And Its Descendant Protocols In Wireless Sensor Network”, IJP2PNTT- Volume3Issue1- 2013.

[7] Karim, L. ; Nasser, N. "Energy Efficient and Fault Tolerant Routing Protocol for Mobile Sensor Network", Communications (ICC), 2011 IEEE International Conference on, On page(s):1-5

[8] Hanady M. Abdulsalam, Layla K. Kamel: W-LEACH: Weighted Low Energy Adaptive Clustering Hierarchy Aggregation Algorithm for Data Streams in Wireless Sensor Networks. ICDM Workshops 2010: 1-8

[9] Gnanambigai,Dr.N.Rengarajan,K.Anbukkarasi “Leach and Its Descendant Protocols: A Survey”, International Journal of ommunication and Computer Technologies Volume 01 – No.3, Issue: 02 September 2012.

[10] Hong, J.; Kook, J.; Lee, S.; Kwon, D.; Yi, S. T-LEACH: The method of threshold-based cluster head replacement for wireless sensor networks. Inf. Syst. Front. 2009, 11, 513–521.

[11] G Haosong, L Gang, Y Yonghwan “A partition based centralized LEACH algorithm for wireless sensor netoworks using solar energy”, ICHIT, Korea, Aug, 2009.

[12] Mayura Bhandarkar, “Analysis of sLEACH for improvement of network lifetime in Wireless Sensor Networks”, ProQuest, 2008.

[13] Kevin Fall , Kannan Varadhan “The ns Manual” MIT, November 4, 2011.

[14] W. Heinzelman "The MIT uAMPS ns Code Extensions, Version 1.0," MIT, August 2000.

A comparative study of decision tree ID3 and C4.5

Badr HSSINA, Abdelkarim MERBOUHA,Hanane EZZIKOURI,Mohammed ERRITALI
TIAD laboratory, Computer Sciences Department, Faculty of sciences and techniques
Sultan Moulay Slimane University
Beni-Mellal, BP: 523, Morocco

Abstract—**Data mining is the useful tool to discovering the knowledge from large data. Different methods & algorithms are available in data mining. Classification is most common method used for finding the mine rule from the large database. Decision tree method generally used for the Classification, because it is the simple hierarchical structure for the user understanding & decision making. Various data mining algorithms available for classification based on Artificial Neural Network, Nearest Neighbour Rule & Baysen classifiers but decision tree mining is simple one. ID3 and C4.5 algorithms have been introduced by J.R Quinlan which produce reasonable decision trees. The objective of this paper is to present these algorithms. At first we present the classical algorithm that is ID3, then highlights of this study we will discuss in more detail C4.5 this one is a natural extension of the ID3 algorithm. And we will make a comparison between these two algorithms and others algorithms such as C5.0 and CART.**

Keywords—Data mining; classification algorithm; decision tree; ID3 algorithme; C4.5 algorithme

I. INTRODUCTION

The construction of decision trees from data is a longstanding discipline. Statisticians attribute the paternity to Sonquist and Morgan (1963) [4] who used regression trees in the process of prediction and explanation (AID - Automatic Interaction Detection). It was followed by a whole family of method, extended to the problems of discrimination and classification, which were based on the same paradigm of representation trees (Thaid - Morgan and Messenger, 1973; CHAID - Kass, 1980). It is generally considered that this approach has culminated in the CART (Classification and Regression Tree) method of Breiman et al. (1984) described in detail in a monograph refers today. [4]

In machine learning, most studies are based on information theory. It is customary to quote the ID3 Quinlan method (Induction of Decision Tree - Quinlan 1979), which itself relates his work to that of Hunt (1962) [4]. Quinlan has been a very active player in the second half of the 80s with a large number of publications in which he proposes a heuristics to improve the behavior of the system. His approach has made a significant turning point in the 90s when he presented the C4.5 method which is the other essential reference when we want to include decision trees (1993). There are many other changes this algorithm, C5.0, but is implemented in a commercial software.

Classification methods aim to identify the classes that belong objects from some descriptive traits. They find utility in a wide range of human activities and particularly in automated decision making.

Decision trees are a very effective method of supervised learning. It aims is the partition of a dataset into groups as homogeneous as possible in terms of the variable to be predicted. It takes as input a set of classified data, and outputs a tree that resembles to an orientation diagram where each end node (leaf) is a decision (a class) and each non- final node (internal) represents a test. Each leaf represents the decision of belonging to a class of data verifying all tests path from the root to the leaf.

The tree is simpler, and technically it seems easy to use. In fact, it is more interesting to get a tree that is adapted to the probabilities of variables to be tested. Mostly balanced tree will be a good result. If a sub-tree can only lead to a unique solution, then all sub-tree can be reduced to the simple conclusion, this simplifies the process and does not change the final result. Ross Quinlan worked on this kind of decision trees.

II. INFORMATION THEORY

Theories of Shannon is at the base of the ID3 algorithm and thus C4.5. Entropy Shannon is the best known and most applied. It first defines the amount of information provided by an event: the higher the probability of an event is low (it is rare), the more information it provides is great. [2] (In the following all logarithms are base2).

A. Shannon Entropy

In general, if we are given a probability distribution P = (p_1, p_2,…, p_n) and a sample **S** then the **Information** carried by this distribution, also called the **entropy of P** is giving by:

$$\boldsymbol{Entropie(P) = -\sum_{i=1}^{n} p_i \times log(p_{i)}} \qquad (1)$$

B. The gain information G (p, T)

We have functions that allow us to measure the degree of mixing of classes for all sample and therefore any position of the tree in construction. It remains to define a function to select the test that must label the current node.

It defines the gain for a test **T** and a position **p**

$$\boldsymbol{Gain(p,T) = Entropie(p) - \sum_{j=1}^{n} (p_j \times Entropie(p_j))} \qquad (2)$$

where values (p_j) is the set of all possible values for attribute T. We can use this measure to rank attributes and build the decision tree where at each node is located the attribute with the highest information gain among the attributes not yet considered in the path from the root.

III. ID3 Algorithm

J. Ross Quinlan originally developed ID3 (Iterative DiChaudomiser 3) [21] at the University of Sydney. He first presented ID3 in 1975 in a book, Machine Learning [21], vol. 1, no. 1. ID3 is based off the Concept Learning System (CLS) algorithm. The basic CLS algorithm over a set of training instances C. ID3 is a supervised learning algorithm, [10] builds a decision tree from a fixed set of examples. The resulting tree is used to classify future samples. ID3 algorithm builds tree based on the information (information gain) obtained from the training instances and then uses the same to classify the test data. ID3 algorithm generally uses nominal attributes for classification with no missing values. [10]

The pseudo code of this algorithm is very simple. Given a set of attributes not target C_1, C_2, ..., C_n, C the target attribute, and a set S of recording learning. [7]

```
Inputs: R: a set of non- target attributes, C: the target
attribute, S: training data.
Output: returns a decision tree
Start
Initialize to empty tree;
        If S is empty then
            Return a single node failure value
        End If
        If S is made only for the values of the same target
then
            Return a single node of this value
        End if
        If R is empty then
            Return a single node with value as the most
        common value of the target attribute values found in
        S
        End if
D ← the attribute that has the largest Gain (D, S) among all
the attributes of R
{$d_j$ j = 1, 2, ..., m} ← Attribute values of D
{$S_j$ with j = 1, 2, ..., m} ←The subsets of S respectively
constituted of $d_j$ records attribute value D
        Return a tree whose root is D and the arcs are
labeled by d1, d2, ..., $d_m$ and going to sub-trees ID3 (R-{D},
C, S1), ID3 (R-{D} C, S2), .., ID3 (R-{D}, C, Sm)
End
```

Fig. 1. Pseudocode of ID3 algorithm

Example 1

Suppose we want to use the ID3 algorithm to decide if the time ready to play ball.
During two weeks, the data are collected to help build an ID3 decision tree (Table 1).

The classification of the target is "should we play ball?" which can be Yes or No.
Weather attributes outlook, temperature, humidity and wind speed. They can take the following values:

Outlook = {Sun, Overcast, Rain}
Temperature = {Hot, Sweet, Cold}
Humidity = {High, Normal}
Wind = {Low, High}

Examples of the set S are:

TABLE I. Data set S

Day	Outlook	Temperature	Humidity	Wind	Play
D1	Sun	Hot	High	Low	No
D2	Sun	Hot	High	High	No
D3	Overcast	Hot	High	Low	Yes
D4	Rain	Sweet	High	Low	Yes
D5	Rain	Cold	Normal	Low	Yes
D6	Rain	Cold	Normal	High	No
D7	Overcast	Cold	Normal	High	Yes
D8	Sun	Sweet	High	Low	No
D9	Sun	Cold	Normal	Low	Yes
D10	Rain	Sweet	Normal	Low	Yes
D11	Sun	Sweet	Normal	High	Yes
D12	Overcast	Sweet	High	High	Yes
D13	Overcast	Hot	Normal	Low	Yes
D14	Rain	Sweet	High	High	No

We need to find the attribute that will be the root node in our decision tree. The gain is calculated for the four attributes.

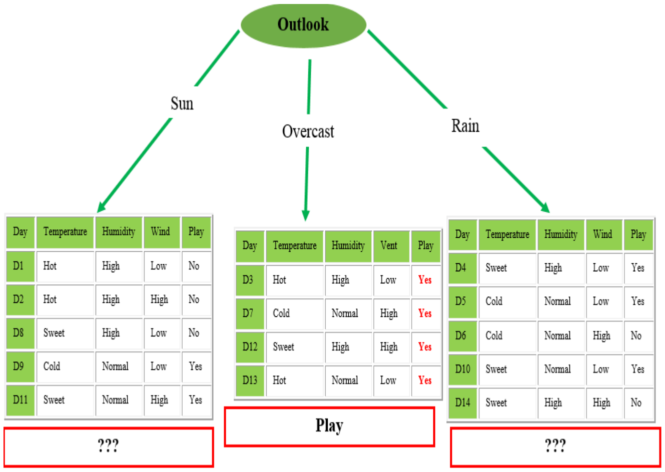
The entropy of the set S:

Entropy (S) =-9/14*$\log_2$ (9/14)-5/14*$\log_2$ (5/14) = 0.94

Calculation for the first attribute
Gain(S, Outlook) = Entropy (S)-5/14*Entropy (S_{Sun})
-4/14*Entropy (S_{Rain})
-5/14* Entropy ($S_{Overcast}$)
=0.94 – 5/14*0.9710-4/14*0 – 5/14*0.9710
Gain(S, Outlook) = 0 .246
Calculation of entropies:
Entropy (S_{Sunl}) = -2/5*$\log_2$ (2/5)-3/5* $\log_2$ (3/5) = 0.9710
Entropy (S_{Rain}) = -4/4*$\log_2$ (4/4)-0* $\log_2$ (0) =0
Entropy ($S_{overcast}$) = -3/5* $\log_2$ (3/5) -/5* $\log_2$ (2/5) =0.9710
As well we find for the other variables:
Gain(S, Wind) = 0.048
Gain(S, Temperature) = 0.0289
Gain(S, Humidity) = 0.1515

Outlook attribute has the highest gain, so it is used as a decision attribute in the root node of the tree (Figure 2).

Since Visibility has three possible values, the root node has three branches (Sun, Rain and Overcast).



Day	Temperature	Humidity	Wind	Play
D1	Hot	High	Low	No
D2	Hot	High	High	No
D8	Sweet	High	Low	No
D9	Cold	Normal	Low	Yes
D11	Sweet	Normal	High	Yes

Day	Temperature	Humidity	Vent	Play
D3	Hot	High	Low	Yes
D7	Cold	Normal	High	Yes
D12	Sweet	High	High	Yes
D13	Hot	Normal	Low	Yes

Day	Temperature	Humidity	Wind	Play
D4	Sweet	High	Low	Yes
D5	Cold	Normal	Low	Yes
D6	Cold	Normal	High	No
D10	Sweet	Normal	Low	Yes
D14	Sweet	High	High	No

Fig. 2. Root node of the ID3 decision tree

So by using the three new sets, the information gain is calculated for the temperature, humidity, until we obtain subsets Sample containing (almost) all belonging examples to the same class (Figure 3).

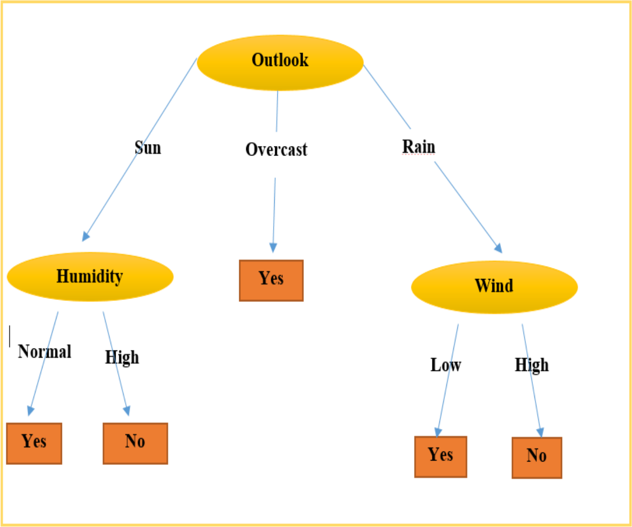


Fig. 3. ID3 Final tree

IV. C4.5 ALGORITHME

This algorithm was proposed in 1993, again by Ross Quinlan [28], to overcome the limitations of ID3 algorithm discussed earlier.

One limitation of ID3 is that it is overly sensitive to features with large numbers of values. This must be overcome if you are going to use ID3 as an Internet search agent. I address this difficulty by borrowing from the C4.5 algorithm, an ID3 extension. ID3's sensitivity to features with large numbers of values is illustrated by Social Security numbers. Since Social Security numbers are unique for every individual, testing on its value will always yield low conditional entropy values. However, this is not a useful test. To overcome this problem, C4.5 uses "Information gain," This computation does not, in itself, produce anything new. However, it allows to measure a gain ratio.

Gain ratio, is defined as follows:

$$GainRatio(p,T) = \frac{Gain(p,T)}{SplitInfo(p,T)} \qquad (3)$$

where **SplitInfo** is:

$$SplitInfo(p,test) = -\sum_{j=1}^{n} P'\left(\frac{j}{p}\right) \times log\left(P'\left(\frac{j}{p}\right)\right) \qquad (4)$$

P' (j/p) is the proportion of elements present at the position p, taking the value of j-th test. Note that, unlike the entropy, the foregoing definition is independent of the distribution of examples inside the different classes.

Like ID3 the data is sorted at every node of the tree in order to determine the best splitting attribute. It uses gain ratio impurity method to evaluate the splitting attribute (Quinlan, 1993). [10] Decision trees are built in C4.5 by using a set of training data or data sets as in ID3. At each node of the tree, C4.5 chooses one attribute of the data that most effectively splits its set of samples into subsets enriched in one class or the other. Its criterion is the normalized information gain (difference in entropy) that results from choosing an attribute for splitting the data. The attribute with the highest normalized information gain is chosen to make the decision.

A. Attributes of unknown value

During the construction of the decision tree, it is possible to manage data for which some attributes have an unknown value by evaluating the gain or the gain ratio for such an attribute considering only the records for which this attribute is defined. [2]

Using a decision tree, it is possible to classify the records that have unknown values by estimating the probabilities of different outcomes.

The new criterion gain will be of the form:

$$Gain\ (p) = F\ (Info\ (T) - Info\ (p,\ T)) \qquad (5)$$

where :

$$Info\ (p, T) = \sum_{j=1}^{n}(p_j \times Entropie(p_j)) \qquad (6)$$

Info (T) = Entropy (T)

F = number of samples in the database with the known value for a given / total number of samples in a set of attribute data.

B. Attributes value on continuous interval

C4.5 also manages the cases of attributes with values in continuous intervals as follows. Let us say that Ci attribute a continuous interval of values. Examines the values of this attribute in the training data. Let that these values are in ascending order, A_1, A_2, ..., A_m Then for each of these values, the partitioned between records those that have values of C, less than or equal to A_j and those which have a value larger then A_j values. For each of these partitions gain is calculated, or the gain ratio and the partition that maximizes the gain is selected.

C. Pruning

Generating a decision to function best with a given of training data set often creates a tree that over-fits the data and is too sensitive on the sample noise. Such decision trees do not perform well with new unseen samples.

We need to prune the tree in such a way to reduce the prediction error rate. Pruning [5] is a technique in machine learning that reduces the size of decision trees by removing sections of the tree that provide little power to classify instances. The dual goal of pruning is the reduction complexity of the final classifier as well as better predictive accuracy by the reduction of over-fitting and removal of sections of a classifier that may be based on noisy or erroneous data.

The pruning algorithm is based on a pessimistic estimate of the error rate associated with a set of N cases, E of which do not belong to the most frequent class. Instead of E/N, C4.5 determines the upper limit of the binomial probability when E events have been observed in N trials, using a user-specified confidence whose default value is 0.25.

Pruning is carried out from the leaves to the root. The estimated error at a leaf with N cases and E errors is N times the pessimistic error rate as above. For a sub-tree, C4.5 adds the estimated errors of the branches and compares this to the estimated error if the sub-tree is replaced by a leaf; if the latter is no higher than the former, the sub-tree is pruned.

D. Exemple 2

We will work with the same example used before but this time we will take continuous values for humidity attribute.

TABLE II. DATA SET S

Day	**Outlook**	**Temperature**	**Humidity**	**Wind**	**Play**
D1	Sun	Hot	**85**	Low	No
D2	Sun	Hot	**90**	High	No
D3	Overcast	Hot	**78**	Low	Yes
D4	Rain	Sweet	**96**	Low	Yes
D5	Rain	Cold	**80**	Low	Yes
D6	Rain	Cold	**70**	High	No
D7	Overcast	Cold	**65**	High	Yes
D8	Sun	Sweet	**95**	Low	No
D9	Sun	Cold	**70**	Low	Yes
D10	Rain	Sweet	**80**	Low	Yes
D11	Sun	Sweet	**70**	High	Yes
D12	Overcast	Sweet	**90**	High	Yes
D13	Overcast	Hot	**75**	Low	Yes
D14	Rain	Sweet	**80**	High	No

⊕ **Treating numerical values**

As C4.5 is an improvement of ID3, then the first step of calculating the gain is the same except for the attributes to continuous values.

In this example we are going to detail the calculation of information gain for an attribute of continuing value.

Gain (S, Humidity) =?

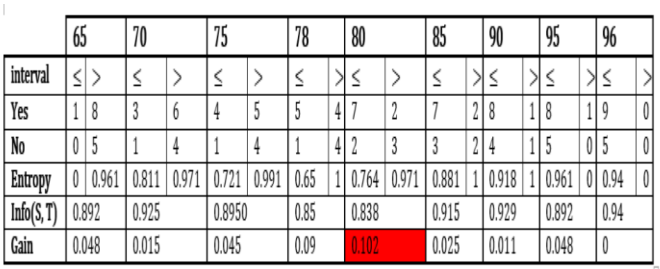
We must now sort the attribute values in ascending order, the set of values is as follows:

{65, 70, 70, 70, 75, 78, 80, 80, 80, 85, 90, 90, 95, 96}

we will remove values that are repeated:

{65, 70, 75, 78, 80, 85, 90, 95, 96}

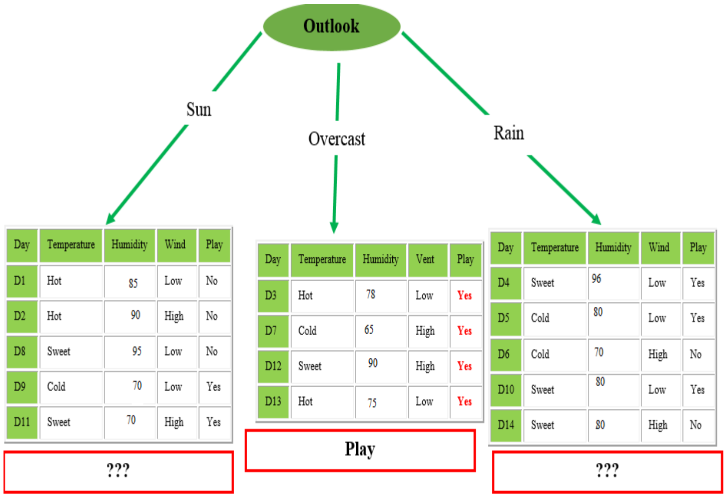
TABLE III. GAIN CALCULATION FOR THE ATTRIBUTE CONTINUOUS HUMIDITY USING C4.5 ALGORITHM



	65		70		75		78		80		85		90		95		96	
interval	≤	>	≤	>	≤	>	≤	>	≤	>	≤	>	≤	>	≤	>	≤	>
Yes	1	8	3	6	4	5	5	4	7	2	7	2	8	1	8	1	9	0
No	0	5	1	4	1	4	1	4	2	3	3	2	4	1	5	0	5	0
Entropy	0	0.961	0.811	0.971	0.721	0.991	0.65	1	0.764	0.971	0.881	1	0.918	1	0.961	0	0.94	0
Info(S,T)	0.892		0.925		0.8950		0.85		0.838		0.915		0.929		0.892		0.94	
Gain	0.048		0.015		0.045		0.09		0.102		0.025		0.011		0.048		0	

Gain (S, Humidity) = 0.102

Then assigns Visibility has the largest value of Information Gain is the root node of the tree (Figure 4).



Day	Temperature	Humidity	Wind	Play
D1	Hot	85	Low	No
D2	Hot	90	High	No
D8	Sweet	95	Low	No
D9	Cold	70	Low	Yes
D11	Sweet	70	High	Yes

Day	Temperature	Humidity	Vent	Play
D3	Hot	78	Low	Yes
D7	Cold	65	High	Yes
D12	Sweet	90	High	Yes
D13	Hot	75	Low	Yes

Day	Temperature	Humidity	Wind	Play
D4	Sweet	96	Low	Yes
D5	Cold	80	Low	Yes
D6	Cold	70	High	No
D10	Sweet	80	Low	Yes
D14	Sweet	80	High	No

Fig. 4. Root node of the C4.5 decision tree

⊕ **Treating attributed to unknown value**

C4.5 accepted principle that a sample with unknown values are distributed based on the probability relative frequency of known values (Table 2).

Suppose the unknown value of D12 day for visibility attributes.

$$\mathrm{Info(S)} = -8/13*\log_2(8/13) - 5/13*\log_2(5/13) = 0.961$$

$$\begin{aligned}\mathrm{Info\ (Outlook,\ S)} &= 5/13*\mathrm{Entropy}\ (S_{\mathrm{Sun}})\\ &+ 3/13*\mathrm{Entropy}(S_{\mathrm{overcast}})\\ &+ 5/13*\mathrm{Entropy}(S_{\mathrm{Rain}})\\ &= 0.747\end{aligned}$$

Entropy (S_{Sun}) $=-2/5* \log_2 (2/5) -3/5* \log_2 (3/5)= 0.9710$
Entropy ($S_{Overcast}$) $=-3/3*\log_2 (3/3) -0/3* \log_2 (0/3)=0$
Entropy (S_{Rain}) $=-3/5* \log_2 (3/5) -2/5* \log_2 (2/5)= 0.9710$
Gain (Outlook) = 13/14 (0.961 – 0.747) = 0.199

When a case of S with the known value is assigned to the subsets Si, the probability belonging to Si is 1, and in all other subsets is 0.

C4.5 therefore associated with each sample (with missing values) in each subset Si weight w representing the probability that the case belongs to each subset (Figure 5).

Fractionation of the set S using the test on the attribute visibility. A new w_i weight is equal to the probability in this case: 5/13, 3/13 and 5/13, because the initial value (Table 2) w is │S1│ = 5+5/13, │S2│ = 3 +3/13, and │S3│ = 5+5/13.

- ⊕ **Generating decision rules**

To make a clearer decision tree model, a path of each leaf can be converted into a production rule IF-THEN.

```
If Outlook= Sun then
        If Humidity <= 70 Then
         Classification = Yes (2.0 / 0);
           else
           Classification = No (3.38 / 0.6);
Else if  Outlook = Overcast
               Classification = Yes (3.2 / 0);
Else if  Outlook= Rain then
               If Wind =High
        Classification = Not (2.0 / 0);
else
Classification = Yes (3.38 / 0).
```

Fig. 5. Decision rules

V. Comparison Between Several Algorithms

A. *ID3 Vs C4.5*

ID3 algorithm selects the best attribute based on the concept of entropy and information gain for developing the tree.

C4.5 algorithm acts similar to ID3 but improves a few of ID3 behaviors:

- ⊕ A possibility to use continuous data.
- ⊕ Using unknown (missing) values
- ⊕ Ability to use attributes with different weights.
- ⊕ Pruning the tree after being created.
    - Pessimistic prediction error
    - sub-tree Raising

Performance Parameters:

Accuracy: The measurements of a quantity to that quantity's factual value to the degree of familiarity are known as accuracy.

The Table 4 presents a comparison of ID3 and C4.5 accuracy with different data set size, this comparison is presented graphically in Figure 6.

TABLE IV. Accuracy Comparison Between ID3 and C4.5 Algorithm

Size of Data Set	Algorithm	
	ID3 (%)	C4.5 (%)
14	94.15	96.2
24	78.47	83.52
35	82.2	84.12

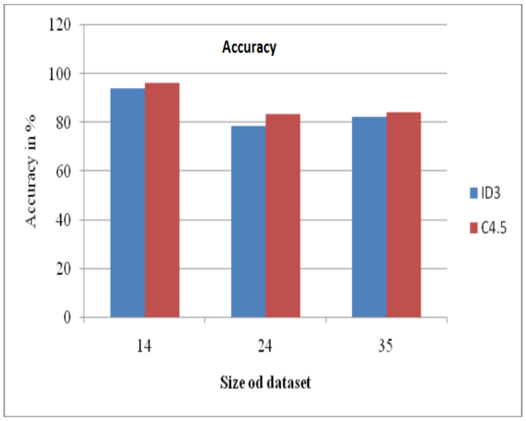


Fig. 6. Comparison of Accuracy for ID3 & C4.5 Algorithm

The 2nd parameter compared between ID3 and C4.5 is the execution time, Table 5 present the comparison.

This comparison is presented graphically in Figure 7.

TABLE V. Comparison of Execution Time For ID3 & C4.5 Algorithm

Size of Data Set	Algorithm	
	ID3 (%)	C4.5 (%)
14	0.215	0.0015
24	0.32	0.17
35	0.39	0.23

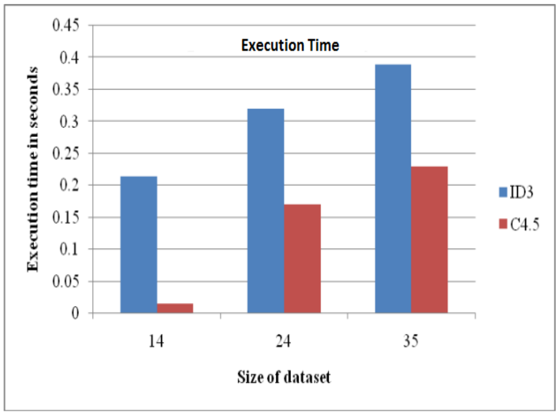


Fig. 7. Comparison of Execution Time for ID3 & C4.5 Algorithm

B. C4.5 Vs C5.0

C4.5 was superseded in 1997 by a commercial system See5/C5.0 (C5.0 for Unix / Linux, See5 pour Windows).

The changes encompass new capabilities as well as much-improved efficiency, and include [13]:

- A variant of boosting, which constructs an ensemble of classifiers that are then voted to give a final classification. Boosting often leads to a dramatic improvement in predictive accuracy.
- New data types (e.g., dates), "not applicable" values, variable misclassification costs, and mechanisms to pre-filter attributes.
- Unordered rule sets—when a case is classified, all applicable rules are found and voted.
- This improves both the interpretability of rule sets and their predictive accuracy.
- Greatly improved scalability of both decision trees and (particularly) rule sets. Scalability is enhanced by multi-threading; C5.0 can take advantage of computers with multiple CPUs and/or cores [13].

C. C5.0 Vs CART

Classification and Regression Trees (CART) is a flexible method to describe how the variable Y distributes after assigning the forecast vector X. This model uses the binary tree to divide the forecast space into certain subsets on which

Y distribution is continuously even. Tree's leaf nodes correspond to different division areas which are determined by Splitting Rules relating to each internal node. By moving from the tree root to the leaf node, a forecast sample will be given an only leaf node, and Y distribution on this node also be determined.

CART uses GINI Index to determine in which attribute the branch should be generated. The strategy is to choose the attribute whose GINI Index is a minimum after splitting.

Let S be a sample, a the target attribute,S1, ..., SK were starting from S, according to the classes of a

$$\mathbf{Gini(S)} = \sum_{i=1}^{K} \frac{|S_i|}{|S|}\left(1 - \frac{|S_i|}{|S|}\right) = \sum_{i \neq j} \frac{|S_i| \times |S_j|}{|S|^2} \quad (7)$$

The C5.0 algorithm differs in several respects from CART, for example:

- The CART tests are always binary, but C5.0 allows two or more outcomes.
- CART uses the Gini diversity index for classifying tests, while C5.0 uses criteria based on the information.
- CART prunes trees using a complex model whose parameters are estimated by cross-validation; C5.0 uses a single-pass algorithm derived from binomial confidence limits.
- CART looks for alternative tests that approximate the results when tested attribute has an unknown value, but C5.0 distributes cases among probabilistic results.
- Speed of C5.0 algorithm is significantly faster and more accurate than C4.5.

VI. CONLUSION

Decision trees are simply responding to a problem of discrimination is one of the few methods that can be presented quickly enough to a non-specialist audience data processing without getting lost in difficult to understand mathematical formulations. In this article, we wanted to focus on the key elements of their construction from a set of data, then we presented the algorithm ID3 and C4.5 that respond to these specifications. And we did compare ID3/C4.5, C4.5/C5.0 and C5.0/CART, which led us to confirm that the most powerful and preferred method in machine learning is certainly C4.5.

REFERENCES

[1] Johan Baltié, DataMining : ID3 et C4.5, Promotion 2002, Spécialisation S.C.I.A. Ecole pour l'informatique et techniques avancées.

[2] Benjamin Devéze & Matthieu Fouquin, DATAMINING C4.5 – DBSCAN, PROMOTION 2005, SCIA Ecole pour l'informatique et techniques avancées.

[3] E-G. Talbi, Fouille de données (Data Mining) -Un tour d'horizon - Laboratoire d'Informatique, Fondamentale de Lille, OPAC.

[4] Ricco Rakotomalala, Arbres de Décision, Laboratoire ERIC, Université Lumière Lyon 2, 5, av. Mendés France 69676 BRON cedex e-mail : rakotoma@univ-lyon2.fr

[5] Arbres de décision, Ingénierie des connaissances (Master 2 ISC).

[6] Thanh Ha Dang, Mesures de discrimination et leurs applications en apprentissage inductif, Thèse de doctorat de l'Université de Paris 6, spécialité informatique, juillet 2007.

[7] Vincent GUIJARRO.K, Les Arbres de Décisions L'algorithme ID3, Elissa, "Title of paper if known," unpublished.

[8] Rakotoarimanana Rija Santaniaina, Rakotoniaina Solofoarisoa, Rakotondraompiana Solofo, Algorithmes à arbre de décision appliqués à la classification d'une image satellite.

[9] J. Fürnkranz, Entscheidungsbaum-Lernen (ID3, C4.5, etc.) (V1.1, 14.01.; neue Folie zu C4.5 Pruning)-- Site web : http://www.ke.tu-darmstadt.de/lehre/archiv/ws0809/mldm

[10] Ankur Shrivastava and Vijay Choudhary ,Comparison between ID3 and C4.5 in Contrast to IDS Surbhi Hardikar, VSRD-IJCSIT, Vol. 2 (7), 2012, 659-667.

[11] A.P.Subapriya, M.Kalimuthu, EFFICIENT DECISION TREE CONSTRUCTION IN UNREALIZED DATASET USING C4.5 ALGORITHM, ISSN: 2230-8563.

[12] Anurag Upadhayay ,Suneet Shukla, Sudsanshu Kumar, Empirical Comparison by data mining Classification algorithms (C 4.5 & C 5.0) for thyroid cancer data set. ISSN:2249-5789 .

[13] Xindong Wu, Vipin Kumar, J. Ross Quinlan, Joydeep Ghosh, Qiang Yang, Hiroshi Motoda, Geoffrey J. McLachlan, Angus Ng, Bing Liu, Philip S. Yu, Zhi-Hua Zhou, Michael Steinbach, David J. Hand, Dan Steinberg. Top 10 algorithms in data mining, Knowl Inf Syst (2008) 14:1–37 DOI 10.1007/s10115-007-0114-2

[14] Introduction to Data Mining and Knowledge Discovery, Third Edition by Two Crows Corporation.

[15] G. COSTANTINI R. Nicole, Probabilité conditionnelle. Indépendance, "Title of paper with only first word capitalized," J. Name Stand. Abbrev., in press.

[16] CELINE ROBARDET, Data Mining, http://prisma.insa-lyon.fr/sc.

[17] P. Habermehl et D. Kesner, Algorithmes d'apprentissage, Programmation Logique et IA

[18] Guillaume CALAS, Études des principaux algorithmes de data Mining, EPITA Ecole d'ingénieures en informatiques.

[19] , Payam Emami Khoonsari and AhmadReza Motie, A Comparison of Efficiency and Robustness of ID3 and C4.5 Algorithms Using Dynamic Test and Training Data Sets International Journal of Machine Learning and Computing, Vol. 2, No. 5, October 2012.

[20] Ron Kohavi, Ross Quinlan, Decision Tree DiscoveryOctober 1999, ISBN:0-19-511831-6.

[21] J.R. QUINLAN, Induction of Decision Trees, 1986, Machine Learning 1:81-106

[22] Leila BENAISSA KADDAR et Djamila MOKEDDEM, Construction Parallèle des Arbres de Décision, University of Science, Technology USTO- Algérie.

[23] Dr. Sanjay Ranka, Classification Part 1,,Professor Computer and Information Science and Engineering ,University of Florida, Gainesville.

[24] K. Ming Leung, Decision Trees and Decision Rules OLYTECHNIC UNIVERSITY, Department of Computer Science / Finance and Risk Engineering.

[25] http://id3alg.altervista.org/

[26] http://www.cise.ufl.edu/~ddd/cap6635/Fall-97/Short-papers/2.htm

[27] J.R. Quinlan, C4.5 programs for machine learningMorgan Kaufmann Publishers, livres Google.

[28] http://rulequest.com/see5-comparison.html (cité dans l'article de R. Quinlan)

[29] http://www.grappa.univ-lille3.fr/polys/apprentissage/sortie004.html#toc10

[30] http://www2.cs.uregina.ca/~dbd/cs831/notes/ml/dtrees/c4.5/tutorial.html

[31] Comparison of C5.0 & CART Classification algorithms using pruning technique International Journal of Engineering Research & Technology (IJERT),Vol. 1 Issue 4, June - 2012, ISSN: 2278-0181.

The Impact of Black-Hole Attack on AODV Protocol

FIHRI Mohammed
Mathematics and Computer Science
Dept, LAVETE Laboratory
Faculty of Sciences and Technical
Settat, Morocco

OTMANI Mohamed
Mathematics and Computer Science
Dept, LAVETE Laboratory
Faculty of Sciences and Technical
Settat, Morocco

EZZATI Abdellah
Mathematics and Computer Science
Dept, LAVETE Laboratory
Faculty of Sciences and Technical
Settat, Morocco

Abstract—**In mobile Ad-Hoc networks, each node of the network must contribute in the process of communication and routing. However this contribution can expose the network to several types of attackers. In this paper, we study the impact of one attack called BLACK-HOLE, on Ad hoc On-Demand Distance Vector routing protocol. In this attack a malicious node can be placed between two or several nodes, and begin dropping all packets from a source and breaking communications between nodes. The vulnerability of the route discovery packets is exploited by the attacker with a simple modification in the routing protocol, in order to control all the traffic between nodes. In this study we simulate the attack with NS2, taking into account the mobility of the network and the attacker, the position of the attacker and finally the number of the attackers. We will also see the impact of this attack in a higher number of loss packet compared with AODV in normal situation.**

Keywords—Black-Hole; AODV; Attack

I. INTRODUCTION

Mobile ad hoc network (MANET) is the network of mobile nodes that requires no infrastructure or centralized management in order to communicate. The nodes can join or leave the network any time thus have a dynamic approach of network topology. Here nodes carry the responsibility of router and host both. It does not have any preexistent infrastructure or centralized controller, and the nodes in it rely on each other in order to communicate. This type of network allows to create and deploy a wide field of communication quickly, and that's what we need in several cases such as a natural disaster or battlefield surveillance where there is no centralized infrastructure and all nodes are capable of movement and must be connected to each other dynamically and arbitrary. It offers better coverage and higher throughput with lower operating cost. However, due to distributed nature of the wireless nodes they are several vulnerabilities and the Black hole is one of the most known.

In this paper we will focus on the performance of AODV (Ad hoc On-Demand Distance Vector) protocol under Black hole attack. we did our simulation with ns2 by implementing a new protocol that adopts the algorithm of AODV and the behavior of a Black hole attacker.

II. OVERVIEW OF AODV ROUTING PROTOCOL

Ad-hoc routing protocols determine the appropriate path from the source to destination and efficiently notify the network with link failure, if it occurs. These protocols are broadly divided into two categories.

- Table-driven routing protocols.
- Source-initiated on-demand driven routing protocols.

Table-driven routing protocols are also known as proactive routing protocols. These protocols desire to maintain consistent and up-to-date routing information in the network. The nodes exchange the routing information periodically and also when there is even a minor change in the network topology and thus, every node maintains one or more routing table to store routing information about every other node in the network.

As a result, these protocols are not preferred in large network. The highly dynamic network also avoids it, as there is lot of message exchanges and it will create congestion and delay in the network. The protocol evolves periodic exchanges even when there is no change in topology and this is simply the wastage of network resources. The mobile devices may also drain out their battery power sooner in such cases. In spite of several drawbacks, these protocols also have the advantage that there is no initial delay as routing information is always available.

AODV is a reactive routing protocol used to find a route between a source and a destination, and allows mobile nodes to obtain new routes for new destinations in order to establish an ad hoc network. In this order several messages are exchanged, different types of link are established, and many information can be shared between the participants nodes. In AODV protocol we find hello message and three others significant type of messages, route request RREQ, route reply RREP and route error RERR. The Hello messages are used to monitor and detect links to neighbours, every node send periodically a broadcast to neighbours advertising it existent ,if a node fails to receive an hello message from neighbour a link down is declared. In order to communicate every node must create routes to the destinations, to achieve that the source node send a request message RREQ to collect information about the route state; if the source receives the RREP message the route up is declared and data can be sent and if many RREP are received by the source the shortest route will be chosen . Any nodes have a routing table so if a route is not used for some period of time the node drop the route from its routing table and if data is sent and a the route down is detected another message (Route Error RERR) will be sent to the source to inform that data not received.

A. *Route Request (RREQ) Message*

This type of message is used by AODV at first in order to locate a destination, this message contains identification of request, sequence number, destination address and also a count of hop initialled by zero.

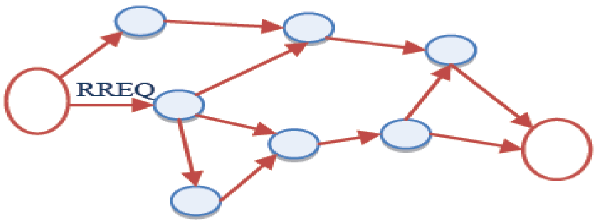


Fig. 1. Route Request (RREQ) Message

B. Route Reply (RREP) Message First

This type of message contains the same fields like Route Request (RREQ) Message, and it sent in the same route of reception of RREQ message. When the source received this message it mean that the destination is ready to accept information and the rout is working correctly.

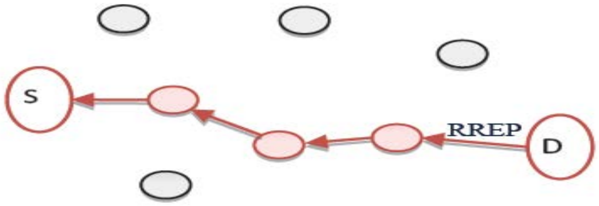


Fig. 2. Route Reply (RREP) Message

C. Route Error (RERR) Message

Sometimes a node detect a destination node that not exists in network, in this scenario another message (Route Error RERR) is sent to the source informing that the data is not received. RERR is like an alert message used to secure table of routing.

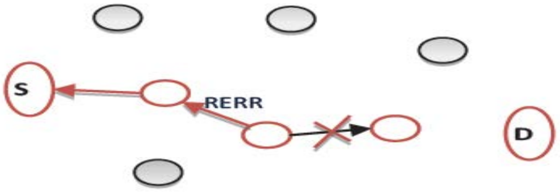


Fig. 3. Route Error (RERR) Message

III. Ad Hoc Network and Security Issues

With the help of routing protocols, nodes in a MANET exchange the information organizing the topology accordingly. This information can be sensitive and targeted by malicious adversaries with an objective to intercept and harm the network or applications. There are two types of security attacks active and passive. In passive attack, the attacker does not affect the functionality of routers or we can say does not inject any kind of disturbance. It just spies on or monitors the routing. While in active attack, the attacker intercepts the routing by several means that can be done by impersonating the newly launched node, repeating the old data packets, disturbing the correct routing by faulty information etc. There are mainly two kinds of sources of MANET threats:

- There can be attackers who attack from outside of the MANET, that is external attacks on a mobile ad hoc network by distortion, overload, redundancy and injecting false routing information.
- In second type of approach, sources of attacks are internal that means the compromised nodes can affect the data adversely to cause failure and can misuse the information of routing to other nodes . There are many drawbacks of MANET that make it vulnerable to various malicious attacks. It doesn't have any fixed infrastructure, nodes can leave and join the network anytime, dynamic topology, limited physical security, frequent routing updates and many other attributes are there due to which MANET often suffers with security attacks.

Some main security issues are briefly described here.

A. Security Issues in MANET

1) Decentralized Connection: Unlike the traditional approach of networks having a fixed infrastructure and central points (access points), MANET is connected in a decentralized manner. It works without a pre-existent infrastructure. The nodes in it work as routers and host, forwarding and receiving the data packets. Due to this absence of a central management, detecting the attacks or monitoring the traffic is very difficult in large scale or highly dynamic MANETs.

2) Uncertain Boundaries: Mobile Ad Hoc Networks do not have any clear or secure boundary. As the nodes can leave or join the network anytime and can communicate with other nodes in the network, it is not possible for a MANET to have certain boundaries. If a node is in the radio range of a MANET, it automatically joins it. This characteristic makes a MANET more susceptible to security threats. Network or the applications running in it can be disturbed through redundancy, distortion, leakage and injection of false information .

3) Dynamic Topology: In MANET, nodes are free to frequently leave and join the network and move arbitrarily. Thus the routes change very often, changing the topology dynamically. These changes in nodes, routes and topologies are very frequent and unpredictable. This results as partitioning of network and cause loss of data packets affecting the integrity of information.

4) Scalability issues: Mobile Ad Hoc Networks are quite different from the traditional approach of fixed networks, where the network is created by connecting the devices through wires so that one can define the network during the initial phase of design and it does not changes during the use. On the other hand, in MANETs nodes are free to move in and out of the network. Nobody can predict the number of nodes a MANET had in past or can have in future.

5) Compromised Node: Compromised node is a node in MANET, on which the attackers get the control through unfair means with the intentions of performing malicious activities. The nodes in MANET are free to move and autonomous in nature. They cannot prevent the malicious activities they are communicating with. As the nodes can join and leave the network anytime, it becomes very difficult to track or monitor the malicious activity because the compromised node changes its position too frequently.

6) Physical Security Limitations: MANET often suffers with security attacks. Mobility of nodes increases this possibility and makes it more susceptible to malicious activities. These attacks include monitoring of traffic with unfair intentions, denial of service attack in which a malicious node claims to be a different node to get the sensitive information, masquerading, spoofing etc.

7) Limited resources: The nodes in a MANET rely only on battery power for energy means, as they do not have any centralized management. Bandwidth constraint also affects as they have lower capacity than that of the infrastructure based networks. MANETs have variable capacity links. Along with limited power, the storage capacity of a MANET is also limited.

B. Security Issues in AODV

AODV protocol is exposed to a variety of attacks, the impact of these attacks on AODV protocol are not the same. Some of these attacks can cause a breakdown of the network connectivity, increasing the end-to-end delay, increasing the number of the loss packets, or shutting down some nodes by consuming all the energy left in there batteries.

1) Black hole attack

2) In black hole attack: A malicious node must be placed between two or more nodes and begin dropping all the traffic. This attack exploits the vulnerability of the route discovery packets of the routing protocol by modifying this last one in order to control all traffic that circulates between nodes.

3) Wormhole attack: In this type of attack, an attacker saves the packets generated in one location of the network and redirects it to another and replays it. This type of attack can be performed by several malicious nodes in same time.

4) Byzantine attack: In this type of attack, individually or cooperatively a malicious nodes carry out attacks such as creating routing loops and forwarding packets through non-optimal paths.

5) Rushing attack: Rushing attacker forwards data and messages very quickly by skipping some of the routing processes. So, in on-demand routing protocol such as AODV, the route between source and destination include rushing nodes.

6) Resource consumption attack: In this type of attack, an attacker attempts to consume battery life of other nodes to take it down.

7) Location disclosure attack: In this type of attack, the related information to the structure of network is revealed by attacker nodes.

IV. Black Hole Attack

Due to these above mentioned issues, MANET is susceptible to many security attacks. Black Hole Attack is one of these attacks. It is a simple but certainly effective Denial of Service attack in which a malicious node, through its routing protocol, advertises itself for having the shortest path to the destination node or to the node whose packets it wants to intercept. It pretends to have enough of fresh routes for a certain destination. The source node assumes it to be true and the data packets are forwarded to a node which actually does not exist, causing the data packets to be lost. When a source node wants to initiate the communication, it broadcasts a RREQ message for route discovery. As soon as the malicious node receives this RREQ packet, it immediately responds with a false RREP message to the respective node advertising itself as the destination or having the shortest path for that destination. Since the malicious node needs not to check its routing table before responding to a routing request, it is often the first one to reply compared to other nodes. When the requesting node receives this RREP, it terminates its routing discovery process and ignores all other RREP messages coming from other nodes. Thus the data packets are sent to such a “hole” from where they are not sent anywhere and absorbed by the malicious node. Often many nodes send RREQ simultaneously; the attacker node is still able to respond immediately with false RREP to all requesting nodes and thus easily takes access to all the routes. In this way source nodes are bluffed by malicious node which gulps a lot of network traffic to itself resulting severe loss of data. Black Hole nodes may also work as a group in a network. This kind of attack is called Collaborative Black Hole attack or Black Hole Attack with multiple malicious nodes.

The main objective of black hole attack is to drape packets and break communications between nodes, all the network's traffic is redirected to a specific node which does not exist at all. Black hole node work with two scenarios, in the first one the node exploits all the vulnerability that exists in an ad hoc network such as announcing itself having a valid route to a destination node; the Second one, the node drupes and controls all the intercepted packets. The Black hole attack in AODV protocol can be classified into two categories: black hole attack caused by RREP and black hole attack caused by RREQ.

A. Black hole attack caused by RREQ

This attack work by sending fakes RREQ messages, an attacker can form a black hole attack as follows:

- Set the originator IP address in RREQ.
- Set the destination IP address in RREQ.
- Set the source IP address of the IP header to its own IP address.
- Set the destination IP address of the IP header to broadcast address or to a nonexistent IP address.
- Increase the sequence number and declaring a low hop count and put them in the related fields in RREQ.

False information about source node is inserted to the routing table of nodes that get the fake RREQ, if these nodes want to send data to the source, at first step they send it to the malicious node.

B. *Black hole attack caused by RREP*

This attack work by sending fakes RREP messages after receiving RREQ from source node, a malicious node can generate black hole attack by sending RREP as follow:

- Set the originator IP address in RREP to the originator node's IP address.
- Set the destination IP address in RREP to the destination node's IP address.
- Set the source IP address of the IP header to its own IP address.
- Set the destination IP address of the IP header to the IP address of the node that RREQ has been received from it.

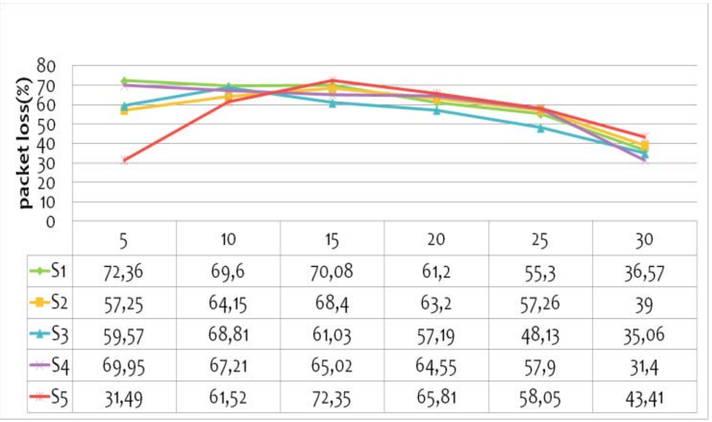
V. Simulation of Black Hole Attack On AODV Protocol

In our simulation of the Black hole attack, we did use Ns2 as a simulator and We fixed some cases where we will study the impact of the attack on AODV protocol and the hole network without knowing the attacked node or the way the traffic is generated. We try to determine the of the attack on the network with the most real way possible,

In order to simulate a Black hole behavior we did integrate a new protocol in NS2 using the source code of AODV protocol and adding the black hole algorithm in it by modifying the AODV functions.

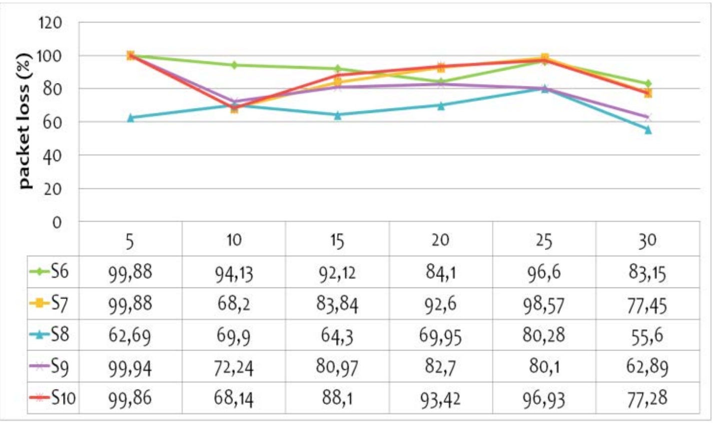
Simulator	Ns2.34
Time	500s
TRAFFIC	CBR
Pause Time	1.0
Max speed	20 m/s
Number of nodes	5 , 10 , 15 , 20 , 25 , 30
Flat space	750 * 750 m

- Scenario 1: we simulate with mobile nodes that use AODV as routing protocol and one non-mobile node with the behavior of a black hole attacker.
- Scenario 2: we simulate with mobile nodes that use AODV as routing protocol and one non-mobile node and another mobile node with the behavior of black hole attackers.
- Scenario 3: we simulate with mobile nodes that use AODV as routing protocol and one mobile node with the behavior of a black hole attacker.
- Scenario 4: we simulate with mobile nodes that use AODV as routing protocol and two mobile nodes with the behavior of black hole attackers.
- Scenario 5: we simulate with mobile nodes that use AODV as routing protocol and two non-mobile nodes with the behavior of black hole attackers.
- Scenario 6: we simulate with non-mobile nodes that use AODV as routing protocol and one non-mobile node with the behavior of a black hole attacker.
- Scenario 7: we simulate with non-mobile nodes that use AODV as routing protocol and two non-mobile nodes with the behavior of black hole attackers.
- Scenario 8: we simulate with non-mobile nodes that use AODV as routing protocol and one mobile node with the behavior of a black hole attacker.
- Scenario 9: we simulate with non-mobile nodes that use AODV as routing protocol and two mobile nodes with the behavior of black hole attackers.
- Scenario 10: we simulate with non-mobile nodes that use AODV as routing protocol, one non-mobile node and another mobile node with the behavior of black hole attackers.



	5	10	15	20	25	30
S1	72,36	69,6	70,08	61,2	55,3	36,57
S2	57,25	64,15	68,4	63,2	57,26	39
S3	59,57	68,81	61,03	57,19	48,13	35,06
S4	69,95	67,21	65,02	64,55	57,9	31,4
S5	31,49	61,52	72,35	65,81	58,05	43,41

Fig. 4. Simulation results for the first five scenarios where the AODV nodes are mobile. X number of nodes, Y % of packet loss.



	5	10	15	20	25	30
S6	99,88	94,13	92,12	84,1	96,6	83,15
S7	99,88	68,2	83,84	92,6	98,57	77,45
S8	62,69	69,9	64,3	69,95	80,28	55,6
S9	99,94	72,24	80,97	82,7	80,1	62,89
S10	99,86	68,14	88,1	93,42	96,93	77,28

Fig. 5. Simulation results for the last five scenarios where the AODV nodes are non-mobile. X number of nodes, Y % of packet loss.

VI. Conclusion

Ad Hoc Network is independent of any fixed infrastructure or central management and have frequent routing updates which makes it easy to set up, low in cost, provides communication by wireless means with nodes working as routers as host.

But along with advantages these features of MANET make it vulnerable to many active and passive security attacks, which affects the confidentiality, integrity and availability of data being transmitted. Black Hole Attack is one of these The Black hole is one of the most powerful attacks on an Ad hoc network; it can cause a complete failure of the network by dropping all the traffic specially when the nodes are non-mobile. In some protocols where we use cluster heads an attacker can be placed between two cluster and cause isolation. In this study we implemented an new protocol that communicate like AODV but behaves like a Black hole and we did choose some study cases where we did use this new protocol to see how the Black hole attack can increase the packets loss.

REFERENCES

[1] H. Deng, W. Li, and D. P. Agrawal, “Routing security in ad hoc networks,” IEEE Communications Magazine, vol. 40, no. 10, pp. 70-75, Oct. 2002.

[2] M. Ghonge, S. U. Nimbhorkar, “Simulation of AODV under Blackhole Attack in MANET,”International Journal of Advanced Research in Computer Science and Software Engineering, Vol 2, Issue 2, Feb 2012.

[3] Y. A. Huang and W. Lee, “Attack analysis and detection for ad hoc routing protocols,” in The 7th International Symposium on Recent Advances in Intrusion Detection (RAID’04), pp. 125-145, French

[4] H.A. Esmaili, M.R. Khalili Shoja, Hossein gharaee, “Performance Analysis of AODV under Black Hole Attack through Use of OPNET Simulator”, World of Computer Science and Information Technology Journal (WCSIT), Vol. 1, No. 2, 49-52, 2011.

[5] Jasvinder, M. Sachdeva, “Effects of Black Hole Attack on an AODV Routing Protocol Through the Using Opnet Simulator,” International Journal of Advanced Research in Computer Science and Software Engineering, Vol 3, Issue 8, Aug 2013.

[6] Charles E. Perkins and Elizabeth M. Royer, “Ad-hoc On-Demand Distance Vector Routing ,” 2nd IEEE workshop on mobile computing systems and applications, New Orleans, Louisiana, USAp. 90-100, Feb. 1999

[7] Seung Yi and Prasad Naldurg, “Security-aware ad hoc routing for wireless networks ,” 2nd ACM international symposium on Mobile ad hoc networking & computing, MobiHoc'01, 2001,p. 299 - 302

[8] Charles Perkins and Elizabeth Royer, “ Ad hoc On-Demand Distance Vector (AODV) Routing ,” RFC 3561, 2003, p. 1-37