ISSN: 2278 – 909X
International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)
Volume 6, Issue 9, September 2017

# ENERGY-EFFICIENT ROUTING USING ALS-ML-SPANNING TREE OF INTERNET OF THINGS IN MANET

**Shubham Garg**
**Gautam Buddha University**

## ABSTRACT

*Internet of things (IoT) connects is used for everything like communication making smart cites, smart home etc. It is embedded with various technologies mobile ad hoc network one such technology play important role in IoT, sensors with limited power as battery backup problem of IoT. In this paper we evaluate of different challenges and energy efficient protocol that make IoT energy efficient using ALS-ML-ST (Alternate least square-Maximum likelihood and spanning tree) and also find the fault detection rate and MSE value using MATLAB 2014a.*

***Keywords: IoT, ALS-ML-ST, MSE, fault detection rate, MANET.***

## I. INTRODUCTION

Internet of Things (IoT) is a network that enables new forms of communication between people and things and between things themselves. Each of the things or objects in IoT communicates with the others and plays a defined role [1–4]. In the future network with IoT, each node acquires information by itself, and humans only verify the information gathered [5]. IoT can be used in the fields of transportation, healthcare, smart environments, and so forth [1] and key network systems for communicating with things in IoT are radio-frequency identification (RFID) systems, wireless sensor network (WSN), and RFID sensor network (RSN). In such networks for IoT, nodes are distributed in a certain region for specific purpose and gather the required information, for example, the information about the temperature, motion, and physical changes [6–8]. The nodes forward the gathered information to the intermediate nodes because of the limited transmission range of the node [9, 10].

Therefore, the intermediate nodes use the unintended energy for the packet forwarding of the source node, which induces high energy consumption of the nodes and thus accelerates *network partitioning*. Therefore, the energy efficiency of the nodes is the key factor that affects the network performance in distributed networks for IoT [11–15]. On the other hand, most of the current routing protocols use hop count as their route selection metric to find the shortest path between source and destination nodes. However, using only hop count as the routing metric is not appropriate in IoT with dynamic network topology, since it is insensitive to packet loss, data rates, link capacity, link quality, channel diversity, interference, or various other routing requirements.

## II. PROBLEM IDENTIFICATION

Energy conservation and fault rate are one of the key issues that requiring proper consideration. The need for energy-efficient routing protocols to prolong the lifetime of these networks is very much required. Moreover, the operation of sensor nodes in an intimidating environment and the presence of error-prone communication links expose these networks to various security breaches. As a result, any designed routing protocol need to be robust and secure against one or more malicious attacks.

This paper aims to provide an effective solution for minimizing the energy consumption of the nodes. The energy utilization is reduced by using efficient techniques for cluster head selection. To achieve this objective, two different cluster-based hierarchical routing protocols are proposed. The selection of an optimal percentage of cluster heads reduces the energy consumption, enhances the quality of delivered data and prolongs the lifetime of a network.

## III. SYSTEM MODEL

The most commonly used objective function is known as the “likelihood” and the most well-understood estimation procedures seek to find parameters that maximize the likelihood. These maximum likelihood estimates are often referred to as MLEs. The likelihood is defined as a conditional probability: P (data|model), the probability of the data given the model. Typically, the only part of the model that can change are the parameters, so the likelihood is often written as P(X|θ) where X is a data matrix, and θ is a vector containing all the parameters of the distribution(s). This notation makes explicit that the likelihood depends on both the data and a choice of any free parameters in the model.

More formally, it start by we have some unique id., observations from a pool, say $X_1$, $X_2$ … Xn, which we will refer to as a vector X. We want to write down the likelihood, L, which is the conditional probability of the data given the model. In the case of independent observations, we can use the joint probability rule to write:

$$L = P(\theta) = P(\theta)P(\theta) \ldots P(\theta) = \prod_{i=1}^{n} P(\theta)$$

Maximum likelihood estimation says: “choose the parameters so the data is most probable given the model” or find θ that maximizes L.

## IV. PROPOSED IMPLEMENTATION

This section discusses proposed methodology for Iot network. It contains MST using Prim’s algorithm,ML and ALS algorithm.

### A. Minimum Spanning Tree

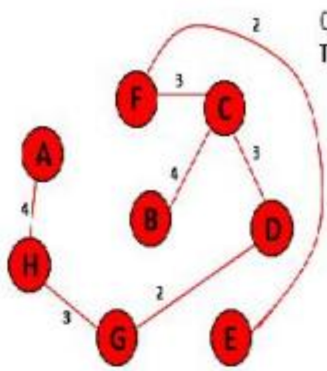
MST method is used to find minimum distance between nodes in the IoT system. MST contains Kruskal's algorithm and Prim's algorithm [7] to form spanning tree. Here we use Prim's algorithm to find minimum distance between each nodes.

**Prim's Algorithm**

In Prim's algorithm is explained below. At the initialization phase a tree with a single vertex is chosen arbitrarily from the graph. Grow the tree by one edge: of the edges that connect the tree to vertices not yet in the tree, find the minimum-weight edge, and transfer it to the tree. Repeat until all vertices are in the tree. The concept is explained in Figure 4.1.

**Algorithm: 1 Prim's algorithm**

*T = a spanning tree containing a single node s;*
*E = set of edges adjacent to v;*
*while T does not contain all the nodes {*
*remove an edge (v, w) of lowest cost from E*
*if w is already in T then discard edge (v, w)*
*else {*
*add edge (v, w) and node w to T*
*add to E the edges adjacent to w*
*}*
*}*




Cost of Minimum Spanning Tree = $\sum d_v = 21$

| | $K$ | $d_v$ | $p_v$ |
|---|---|---|---|
| A | T | 4 | H |
| B | T | 4 | C |
| C | T | 3 | D |
| D | T | 0 | – |
| E | T | 2 | F |
| F | T | 3 | C |
| G | T | 2 | D |
| H | T | 3 | G |

**Figure 4.1: Prim's MST**

In Figure 4.1, using prim's algorithm to find minimum spanning tree:(1)Start with any nodes here start from D.(2)updated distance of adjacent nodes, unselected nodes. (3)Select node with minimum distance G.(4)Repeat step 2 until all vertices are in tree.

B. **Alternating Least Square (ALS)**

Alternating Least Squares rotates between fixing one of the unknowns ui or vj . When one is fixed the other can be computed by solving the least-squares problem. This approach is useful because it turns the previous non-convex problem into a quadratic that can be solved optimally [9]. A general description of the algorithm for ALS algorithm for collaborative filtering taken from Zhou et. al [11] is as follows:

***Step 1: Initialize matrix V by assigning the average rating for that movie as the first row, and small random numbers for the remaining entries.***

The least squares simply adding up the squared differences between the model and the data. Minimizing the sum of squared differences leads to the maximum likelihood estimates in many cases, but not always. One good thing about least squares estimation is that it can be applied even when your model doesn't actually conform to a probability distribution (or it's very hard to write out or compute the probability distribution).One of the most important objective functions is the so-called posterior probability and the corresponding Maximum-Apostiori-Probability or MAP estimates/estimators. In contrast to ML estimation, MAP estimation says: "choose the parameters so the model is most probable given the data we observed." Now the objective function is P(theta|X) and the equation to solve is

$$\frac{\partial}{\partial\theta}P(\theta|X) = 0$$

As you probably already guessed, the MAP and ML estimation problems are related via Bayes' theorem, so that this can be written as

$$P(X) = \frac{P(\theta)}{P(X)}P(\theta) = \frac{P(\theta)}{P(X)}L$$

Once again, it is convenient to think about the optimization problem in log space, where the objective function breaks into three parts, only two of which actually depend on the parameters.

$\frac{\partial}{\partial\theta}log\ P(X) = \frac{\partial}{\partial\theta}log\ P(\theta) + \frac{\partial}{\partial\theta}log\ L - \frac{\partial}{\partial\theta}log\ P(X) = \frac{\partial}{\partial\theta}log\ P(\theta) + \frac{\partial}{\partial\theta}log\ L$ =***Step 2: Fix V, solve U by minimizing the RMSE function.***

***Step 3: Fix U, solve V by minimizing the RMSE function similarly.***

***Step 4: Repeat Steps 2 and 3 until convergence.***

**Development tool**

The raw data development kit provided on the (field.header.stamp) Dataset contains MATLAB demonstration code with file which gives further details.

Here, we will briefly give the most important features. Before running the scripts, steps are:

***Step 1: Read curve's points from the Excel file***

***Step 2: Extract Row Co-ordinate and column too.***

***Step 3: Define fault rate detection***

***Step 4: fault Detected Parameter Machine// Compute current from voltage vector***

***Step 5: Do "coerce" to limit extract values to positive values***

***Step 6: fault Detection rate minimization as per coverage distance***

***Step 7: Define sigmoid function regularization for numerical stability***

***Step 8: Define ML weight matrix response update***

***Step 9: check convergence***

***Step 10: Determine Minimum spanning tree using Primes algorithm for path optimality.***

***Step 11: compute MSE and fault detection rate.***

## V.RESULT

This work presents a framework for energy efficient and sensor fault detection rate. The motive proposed technique in the context of the problem of the data value fault detection and calculate MSE. We use the ALS (Alternate least square) to cover the current data and improve energy efficiency and detect fault detection rate. The regression result can be used to detect the data value fault. We uses maximum likelihood estimation method for estimating the parameter of statistical model given clarifications, by finding the parameter values that maximize the likelihood of making the observations given the parameters namely Fault detection rate and MSE value. We also used the prim's algorithm to finding minimum distance for energy level.

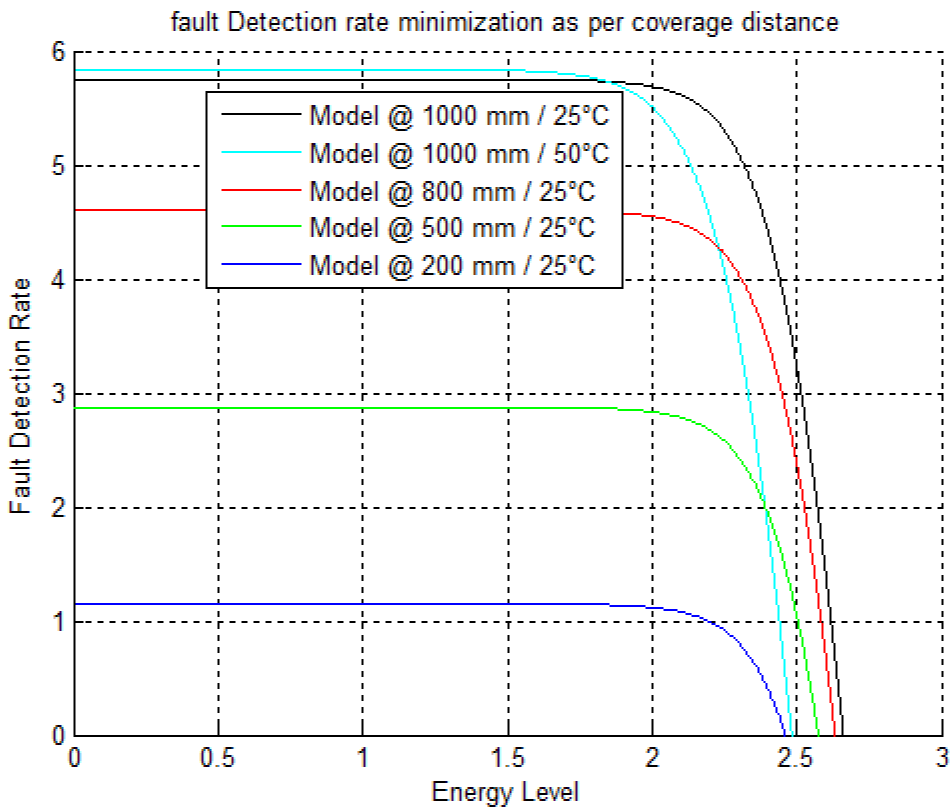



**Figure 5.1: Fault detection rate minimization as per coverage distance over Energy level**

Fault detection rate is the ratio of coverage distance failures divided by total energy level.

Fault Detection = (coverage distance failures / Total Avionics Failure) x100

Usually fault detection rate lies between 90-95%.

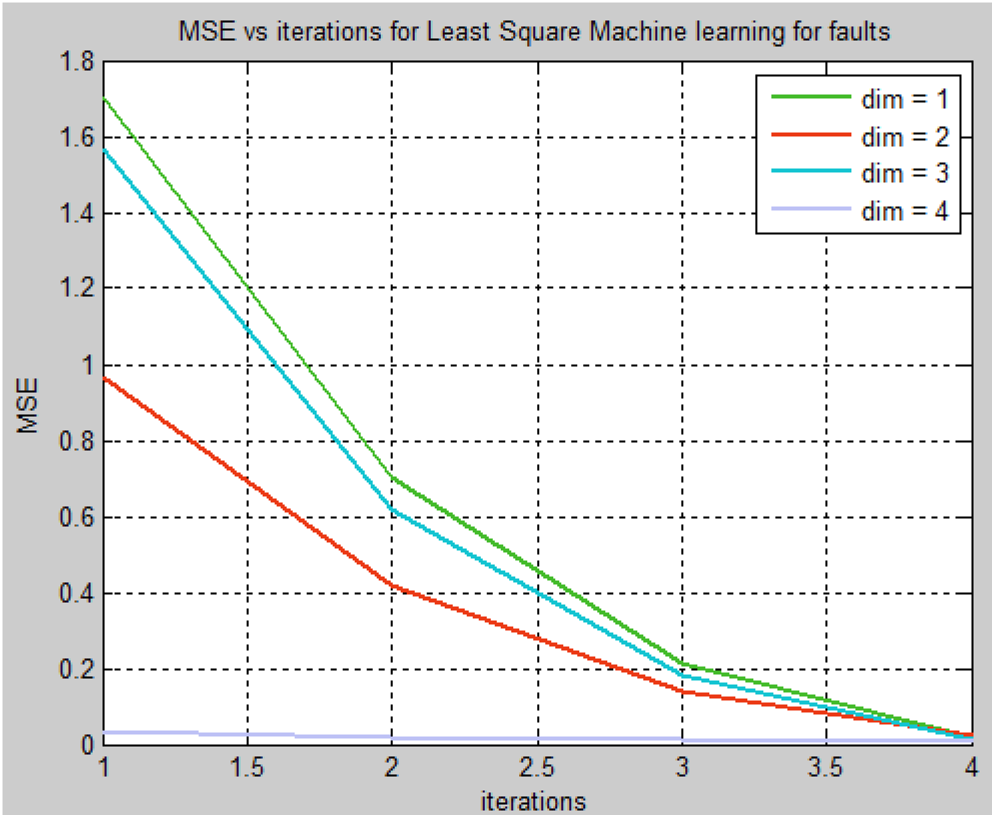



**Figure 5.2: MSE vs iteration for least square machine learning for faults**

## VI.CONCLUSION

This paper presented a method for cost-effective IoT device production that addresses the grand scale of the IoT paradigm through a mass-customization-based intelligent IoT system. We present a self-learning sensor fault detection framework in this paper. We propose a model which can represent the sensor value, sensor relationship, and sensor status transformation. Experimental results show that our system can detect 95% of data fault in the simulation data. Internet today covering all aspect of our life"s. with development of technology like MANET able to senses environment.

## REFERANCES

[1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: a survey," *Computer Networks*, vol. 54,no. 15,pp. 2787–2805, 2010.

[2] ITU Internet Reports, *The Internet of Things*, 2005.

[3] O. Vermesan, *Internet of Things Strategic Research Roadmap*, IERC-European, 2011.

[4] Cluster of EuropeanResearch Projects on the Internet of Things, *Vision and Challenges for Realising the Internet of Things*, 2010.

ISSN: 2278 – 909X
International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)
Volume 6, Issue 9, September 2017

[5] L. Tan and N. Wang, "Future internet: the internet of things," in *Proceedings of the 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE '10)*, vol. 5, pp. V5376–V5380, chn, August 2010.

[6] G. Lee, "A cluster-based energy-efficient routing protocol without location information for sensor networks," *International Journal of Information Processing Systems*, vol. 1, no. 1, pp. 49– 54, 2005.

[7] T. Dubey, "Self-localized packet forwarding in wireless sensor networks," *International Journal of Information Processing Systems*, vol. 9, no. 3, 2013

[8] A. Sinha, "Performance evaluation of data aggregation for cluster-based wireless sensor network," *Human-Centric Computing and Information Sciences*, vol. 3, article 13, 2013

[9] X. Li, "Achieving load awareness in position-based wireless ad hoc routing," *KITCS/FTRA Journal of Convergence*, vol. 3, no. 3,
2012.

[10] C.Huang, "Enhancing network availability by tolerance control in multi-sink wireless sensor networks," *KITCS/FTRA Journal of Convergence*, vol. 1, no. 1, 2010.

[11] M. Yoon, "An energy-efficient routing protocol using message success rate in wireless sensor networks," *KITCS/FTRA Journal of Convergence*, vol. 4, no. 1, 2013

[15] B. Singh, "A novel energy-aware cluster head selection based on particle swarm optimization for wireless sensor networks," *Human-Centric Computing and Information Sciences*, vol. 2, no. 13, 2012.