TITLE:

Using MOPSO to find optimal number of clusters in a wireless network based on energy-efficiency.

**TOPIC FOR: DESIGN AND ANALYSIS OF ALGORITHMS**

# **Abstract: (>500 words)**

In this article we propose a method to solve the problem of routing in wireless sensor networks (WSN) by requiring minimum energy consumption between nodes in the network and forming clusters to minimize node transfers.

We have used multiple objectives based on different parameters to solve for optimality of the solution. Parameters such as temperature, minimal clustering, routing path distance and energy efficiency have been taken into account to solve multi-objective problem as stated above.

An energy efficient network means that the network will have minimal energy transmission between nodes and thus temperature of the nodes will be minimal. This enhances the longevity of the nodes present in the network.

Our main emphasis in this paper is on utilizing artificial intelligence techniques such as clustering and Multiple Objective Particle Swarm Optimization (PSO) algorithm for finding a safe and efficient routing in wireless sensor networks.

The dispersed sensors are placed in a large area and mainly help in monitoring, detecting and recording the physical or the environmental conditions of that particular place. These physical conditions mainly include temperature, sound, wind, etc. It is described as collection of nodes that are placed randomly in sensor field. These nodes are connected to each other through a wireless channel such that the data transmission can take place between them. Energy efficient algorithms are required as the sensor nodes are battery operated. This is mainly required because of several reasons which are no human intervention, inaccessible remote areas, no recharging facility, etc. Data transmission is not possible over the network, whenever node is energy deficient. Clustering is one such technique which is used to enhance the lifetime of these nodes. Hierarchical routing protocol divides the network in to clusters with one cluster head and member nodes. The CH nodes collect the data from remaining nodes in the cluster and then send the data to a Base Station (BS). But this is possible only till the network is alive. Network life time is directly related to the battery. Therefore, major concern in WSN is to save node energy. Energy is required in formation of clusters as well as in selection of CHs. CHs depletes their energy in receiving the data from sensor nodes, in data aggregation and in transmission of data to the sink. Therefore, CHs must be energy efficient nodes because of the transmission and reception responsibility. If CH nodes die quickly the respective cluster disconnected from the network and important events may be missed out. In this research, we have focused on achieving energy efficiency through optimal selection of cluster head.

1. **Introduction:**

**Wireless sensor network** (**WSN**) refers to a group of spatially dispersed and dedicated sensors for monitoring and recording the physical conditions of the environment and organizing the collected data at a central location. WSNs measure environmental conditions like temperature, sound, pollution levels, humidity, wind, and so on.

These are similar to wireless ad hoc networks in the sense that they rely on wireless connectivity and spontaneous formation of networks so that sensor data can be transported wirelessly. WSNs are spatially distributed autonomous sensors to *monitor* physical or environmental conditions, such as temperature, sound, pressure, etc. and to cooperatively pass their data through the network to a main location.

Wireless sensor networks (WSNs) are consist of tens, hundreds or even thousands of self-directed sensors which are embedded in an environment wirelessly at a distance from each other to communicate with each other, and their task is discovering and aggregation of environmental information and transmitting it to a monitoring centre. Continuous movement of sensor nodes and their limited battery power causes routing problems for these types of networks. Thus, providing a reliable and secure protocol in wireless sensor networks seems crucial.

 A [sensor node](https://en.wikipedia.org/wiki/Sensor_node) might vary in size from that of a shoebox down to the size of a grain of dust, although functioning "[motes](https://en.wikipedia.org/wiki/Sensor_node)" of genuine microscopic dimensions have yet to be created. The cost of sensor nodes is similarly variable, ranging from a few to hundreds of dollars, depending on the complexity of the individual sensor nodes. Size and cost constraints on sensor nodes result in corresponding constraints on resources such as energy, memory, computational speed and communications bandwidth. The topology of the WSNs can vary from a simple [star network](https://en.wikipedia.org/wiki/Star_network) to an advanced [multi-hop](https://en.wikipedia.org/wiki/Multi-hop_routing) [wireless mesh network](https://en.wikipedia.org/wiki/Wireless_mesh_network).

Today, Wireless Sensor Networks (WSN's) are used at various places in the form of alarm detectors and sensors. Numbers of clustering algorithms have been developed to improve the energy balance of the WSN's because energy is the main aspect of WSN's during data transmission. These algorithms are mainly used for increasing the lifetime of these sensor networks. One such basic algorithm is LEACH. It is also seen that PSO based algorithms give the best results in this perspective.

They are used in geo-sensing applications, assessing the pollution levels in the atmosphere and area monitoring purposes.

1. **Describe the Model with Methodology:**

The process of combining the solutions can be explained as combining current solution with the target solution. This can be conducted by firstly determining the nearest sensor in the target solution which can be represented in the following equation:

(𝑥𝑛𝑒𝑤,𝑦𝑛𝑒𝑤) = (𝑥,𝑦)+𝑅𝑎𝑛𝑑 ×((𝑥𝑡𝑎𝑟𝑔𝑒𝑡,𝑦𝑡𝑎𝑟𝑔𝑒𝑡)−(𝑥,𝑦)) hk (x)=0 k=1,2,…K;

Hypermetric Volume(HV) metric is also called as S-metric and aims to give information about both closeness and diversity within the set of non-dominated solutions 𝑃 𝑠. It aims to calculate the volume covered by the solutions in the objective space. Hence, the worst possible point would be used as a reference point W in the objective space.

Let x be a solution within 𝑃 𝑠, the Hypercube(x) will be initiated by taking into the account both W and x as the corners of the hypercube in the objective space. In this regard, HV can be computed by the volume of the union of the hypercubes as in the following equation:.

𝐻𝑉 = 𝑣𝑜𝑙𝑢𝑚𝑒 (∪ 𝑥 ∈ 𝑃𝑠 𝐻𝑦𝑝𝑒𝑟𝐶𝑢𝑏𝑒(𝑥))

In fact, the greater value of HV indicates superior performance.

For calculating the routing distance :

distance=sqrt( (sink.x-PriorityX(PresenceSize(2)) )^2 + (sink.y-PriorityY(PresenceSize(2)))^2) ;

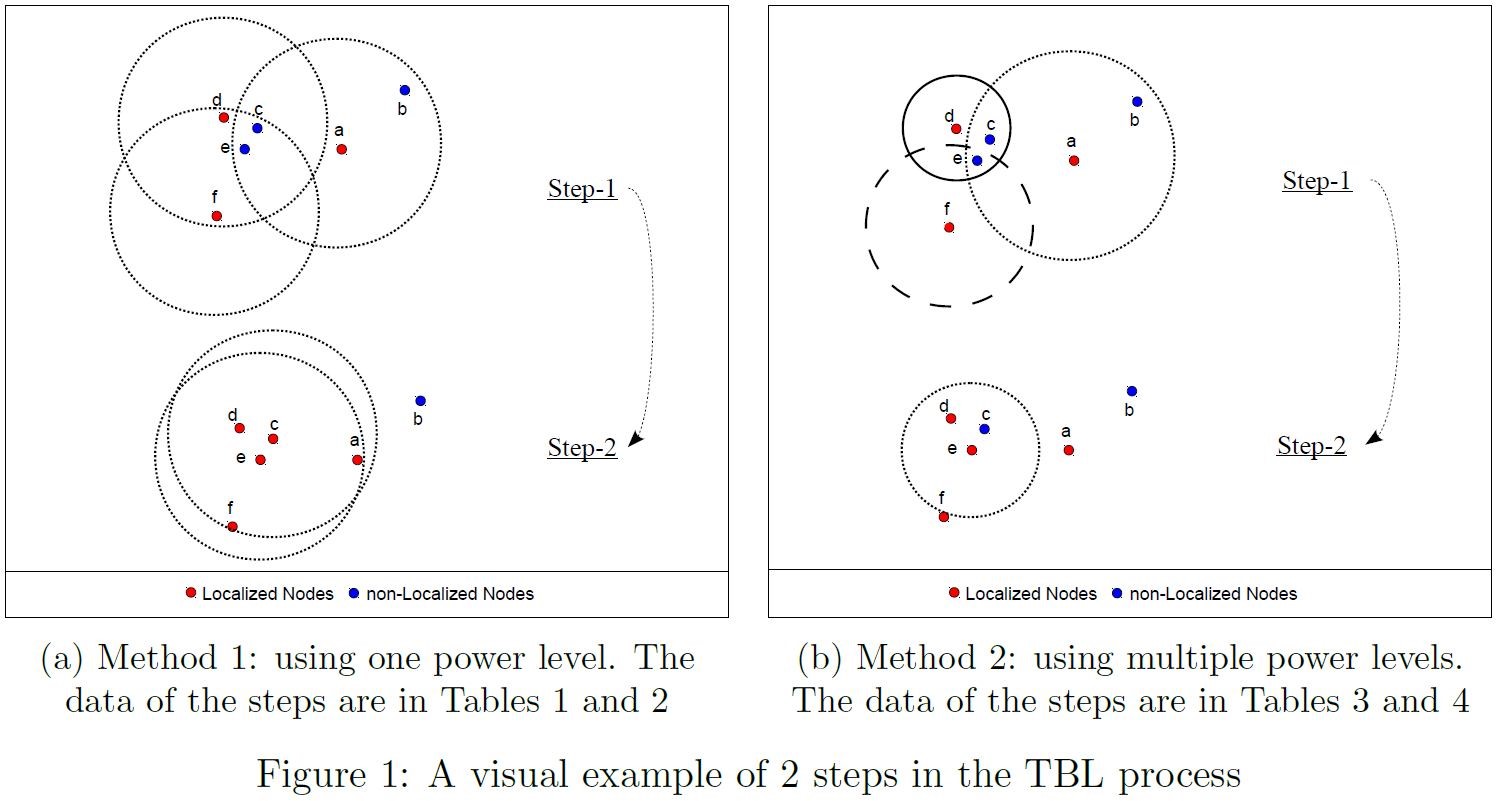
Where PresenceSize represents area of influence of a node and sink represents its current position. Priority represents order of node in the network.

For calculating the movement of the node:

Xi=Xi-1+(sign((rand()\*2)-1)\*rand()\*Rate)

Where xi is the current node and Xi-1 is the previous node. Rate signifies the rate of movement of energy between nodes.

When tackling multi-objective problems however, a few modifications must be made. First, the objective is to find not one “global best” solution, but a set of solutions comprising the Pareto Front. To do this, an *archive* of non-dominated solutions is kept, where all non-dominated solutions found at each iteration are stored.



1. **Algorithm of the MOPSO implementation:**

**Input**: All parameters (no. of nodes and initial energy of each node)

**Output**: Optimal no. of clusters in the network.

1. Initialize the no. of nodes, initial energy in the nodes, repository limit.

(E0=initial energy value, NodeNum=no. of nodes, RepLimit=maximum limit)

1. Now we initialize no of steps and no. of iterations in each seed.

(iteration=integer value, steps=no. of step)

1. We initialize rate of movement of energy or transmission rate of nodes from values obtained from archive vector set.
2. For each iteration find the energy of transmission after round taking into account routing distance and rate of transmission.

Energy(ii,PriorityEnergy(jj))=E(PriorityEnergy(jj))- ( (ETX+EDA)\*(4000) + Emp\*4000\*( distance\*distance\*distance\*distance ));

1. Select clusters based on optimality values, based on condition.

* Temp = CurFitness >= LBest;

Result = (sum (Temp,2) == ObjNum); or,

* Temp = CurFitness <= LBest;

Result = (-1).\*(sum (Temp,2) == ObjNum) + Result;

1. Find dead nodes (which do not have energy left) in them. Eliminate them and put in separate clusters.

NewX=(Energy<=0 )\*(-XBound/2) + ~(Energy<=0).\*NewX

1. For each particle in the swarm:
   1. Select leader from the archive, and obtain global best.

NewV = Weight \* V + C1 \* (rand (1,ParticleSize(2)).\* (LBestP - X))+

C2 \* (rand (1,ParticleSize(2)).\* (GBestP - X));

* 1. Update clusters as in step 5.
  2. Update order of nodes, from PSO(non-dominated pairs).

[CurFitness Energy] = MultObjFitness (GBestAC, GBestVal,X,Y,R,E,next,sender,ETX,EDA,Emp,PlotSize,do,Efs,2,SenderIndex);

* 1. Update energy values of nodes from step 4.

1. Update the archive of non-dominated solutions
2. Repeat for each seed(updation), goto step no. 3.
3. Now plot graph in MATLAB for final energy values, optimality figures and objective plots.
4. Print the optimal no. of clusters.

The solution works well if no. of iterations for each step and no. of steps is increased allowing the seeding of the archive and generating dominated pairs.

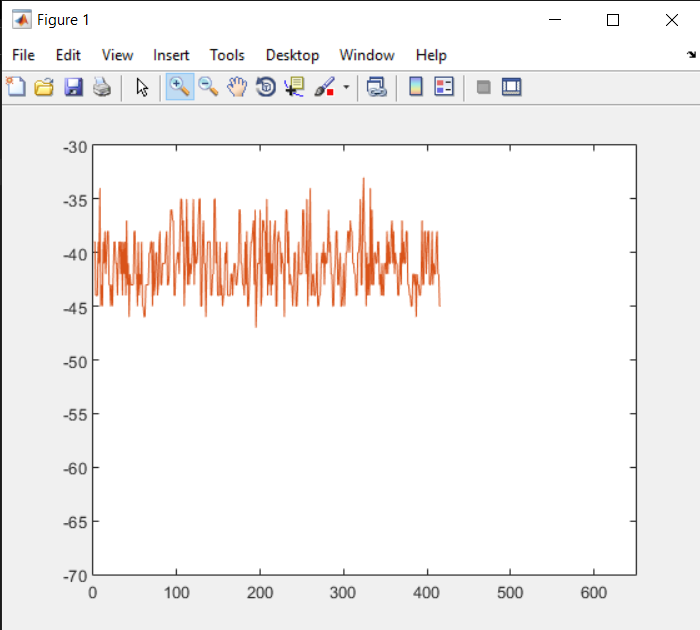
**Example ILLUSTRATION:**

We run the model for 50 steps and for each step we keep 5 iterations, seeding the archive 3 times.

Here we select 8 as the initial energy of nodes and keep 100 nodes in our model. Rate of transmission are randomly fed to the model.

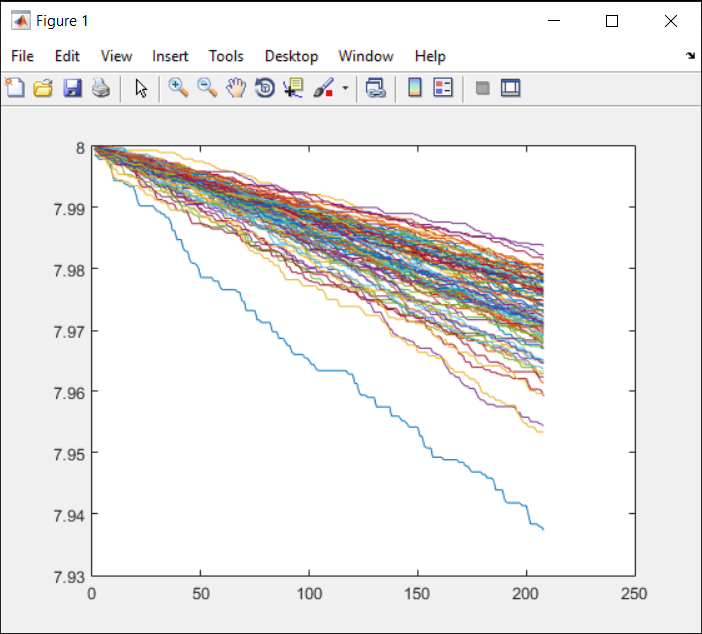
**Experimental Results**

The figure shows optimality of clusters obtained over the stages.



*The no. of final clusters in the network obtained is 45.*

The following figure shows distribution of energy values in nodes at each step.

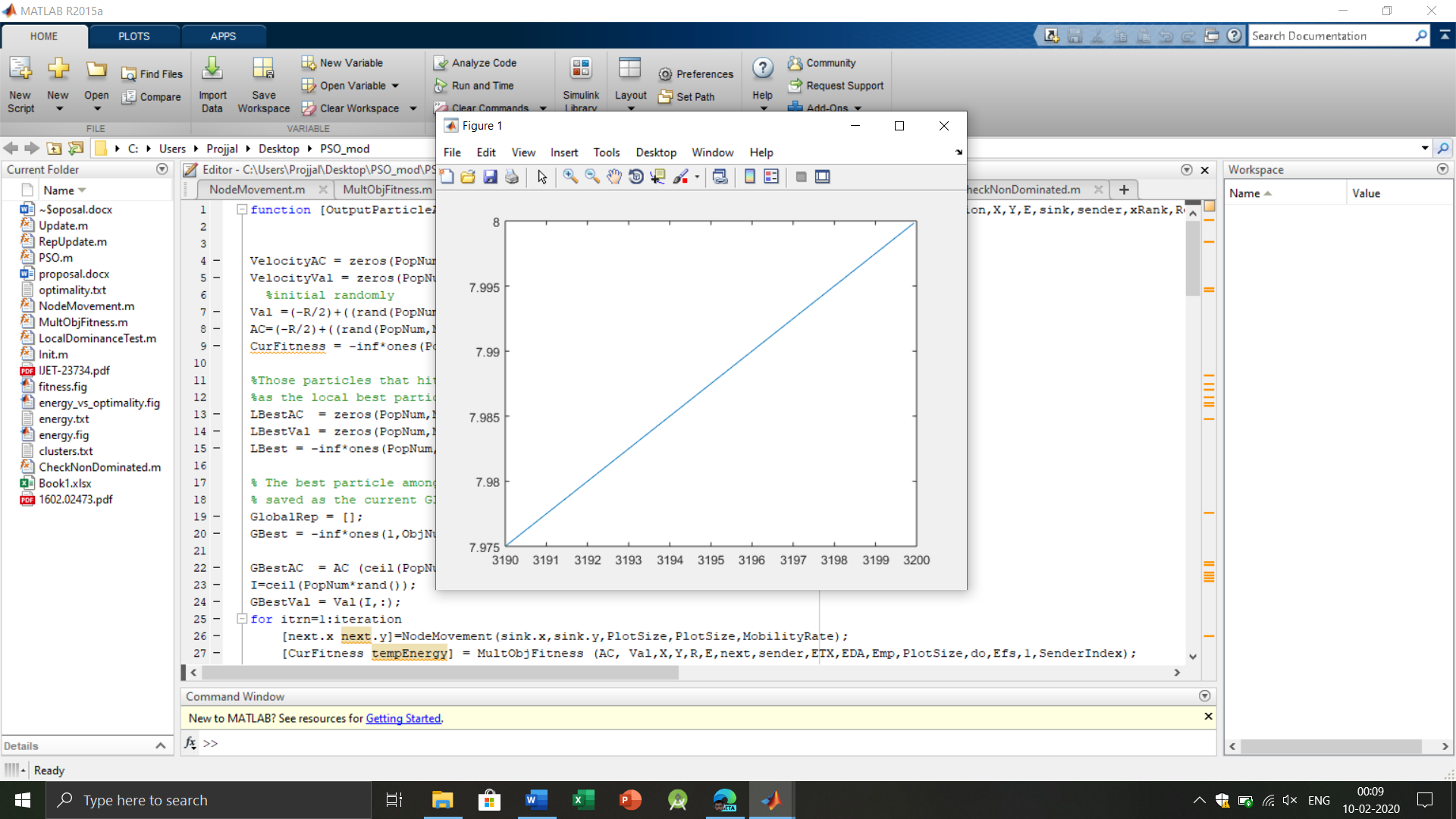


Thus from the figure we see that energy values shift downward towards the minimal value. For the last stage represented in ‘blue-line’, we see that there is a sudden drop which is our solution.

Also we find that for minimum values of clusters, the energy values are steep but then for greater values the energy decreases, this is for dead nodes and

Further we can see that with increasing steps our energy levels tends to minimize which predicts a stable or a feasible environment, which is our main objective. As, distances between frequent nodes are minimized (ideal clusters are formed), energy transmission is greatly reduced.

The following figure gives us the energy vs optimality plot.



Thus from the nature of our solution we can determine that with decreasing energy levels of the setup, the stability increases (or, our solution tends to be more optimal)

**Conclusion:**

The above algorithm gives us the minimal no. of clusters taking into account both objectives of routing-distance, energy transmission, and minimization of clusters of nodes.

**References:**

Zhang, Q.; Li, H. MOEA/D: A multi-Objective evolutionary algorithm based on decomposition. *IEEE Trans. Evol. Comput.* **2007**, *11*, 712–731

I. Khan, F. Belqasmi, R. Glitho, N. Crespi, M. Morrow, and P. Polakos, "Wireless sensor network virtualization: A survey," IEEE Communications Surveys & Tutorials, vol. 18, pp. 553-576, 2016.