Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

# A New Clustering Algorithm based on ACO and K-medoids Optimization Methods

**Li Peng*, Guo-Yong Dong**, Fei-Fei Dai***, Guo-Ping Liu******

** School of IoT Engineering, Jiangnan University, Wuxi, Jiangsu, P.R. China, (e-mail: penglimail2002@163.com).*
*** School of IoT Engineering, Jiangnan University, Wuxi, Jiangsu, P.R. China, (e-mail: qqdongguoyong@163.com).*
**** School of IoT Engineering, Jiangnan University, Wuxi, Jiangsu, P.R. China, (e-mail: daifeifei0229@qq.com).*
**** School of Engineering, University of South Wales, Pontypridd CF37 1DL, UK, CTGT Center, Harbin Institute of Technology, Harbin 150001, China, (e-mail: guoping.liu@southwales.ac.uk)}*

**Abstract:** The existing wireless sensor network clustering routing algorithms commonly have the problems of unbalanced network energy consumption and uneven clustering. A new clustering algorithm based on ACO and K-medoids optimization methods is proposed in this paper. The optimized K-medoids clustering algorithm can cluster sensor nodes effectively to solve the problem of uneven clustering. At the same time based on the improved ACO algorithm, this new algorithm can fully consider nodal residual energy either when cluster heads are replaced or in time of route selection and data transmission between cluster heads. Compared with other routing algorithms, this new algorithm has better performance and good capacity of balancing network energy consumption and lengthening network life cycle as result verified by simulation experiments.

## 1. INTRODUCTION

With such technologies as integrated circuit wireless sensor and MEMS increasingly becoming mature and gaining rapid development, those multifunctional wireless sensors with low cost and low energy consumption have been put into mass production and WSN containing abundant wireless sensors have emerged.

How to improve the utilization effectiveness of nodal energy, balance energy consumption of nodes and lengthen the life cycle of network has become the hotspot issue of studying WSN, due to sensors' limited energy and other factors like inconvenience of battery charging and replacing in relatively bad environments.

In terms of the above issue of energy saving, some new routing algorithms have been introduced by many scholars home and abroad, among which Ant Colony Optimization (ACO) algorithm has prominent effect regarding network route, in order to reduce the energy consumption of data acquisition and data communication. However, some algorithms in the available literature do not take nodal residual energy into consideration, thus leading to unevenness of network energy consumption, while others pay no attention to the features of WSN in light of energy saving.

Based on the paper Li PENG has proposed before, which may cause uneven clustering, a new clustering routing algorithm combining K-medoids optimization algorithm and ACO is proposed. This new algorithm begins by using the optimized K-medoids clustering algorithm to cluster sensor nodes within a region so that clustering can become even. Then with respect to route selection, this new algorithm benefiting from ACO pays much attention to nodal residual energy, hence the equalization of network energy consumption is realized when cluster heads are replaced or in time of routing selection and data transmission between cluster heads.

## 2. THE NETWORK ENVIRONMENT

### 2.1 Network Model

We make the following hypothesises in the bidimensional WSN of this paper: The wireless channels are equivalent, and the transmit-receive is all-dimensional. The base station is fixed and far away from the target area. All the nodes have the same energy at the very beginning.

### 2.2 Model of Energy Consumption in Wireless Transmission

Each time when $j$ bits data are transmitted and the transmission distance is $d$, the energy consumption of the transmitter is:

$$E_{tx}(j,d)=\begin{cases}E_{tx-clec}\times j+\varepsilon_{fs}\times j\times d^2, d<d_0\\ E_{tx-clec}\times j+\varepsilon_{amp}\times j\times d^4, d\geq d_0\end{cases} \tag{1}$$

$E_{tx-elec}$ stands for the energy consumed by the transmitting circuit when 1 bit of data is transmitted. The power amplification parameter $\varepsilon_{fs}$ and $\varepsilon_{amp}$ are relevant to the adopted energy consumption model of transmission. $\varepsilon_{fs}$ represents free space transmission and $\varepsilon_{amp}$ multipath attenuation transmission. $d_0$ is the boundary condition to differentiate two models, $d_0 = \sqrt{\varepsilon_{fs} / \varepsilon_{amp}}$ . When the transmission distance $d$ is greater than $d_0$ , the energy consumption of the transmitter will increase sharply with the transmission distance widening.

In the receiving end, the receiving circuit consumes the following energy when receiving $j$ bits data:

$$E_{rx}(j,d) = E_{tx-elec} \times j \tag{2}$$

*2.3 Model of Data Fusion*

It is hypothesized in this paper that when any node in WSN is used for data processing and data fusion, its energy consumption is $E_{DA}$ once 1 bit of data is processed.

## 3. IMPROVED CLUSTERING ROUTING ALGORITHM

*3.1 Improved K-medoids Clustering Algorithm*

XIA Ning-xia once improved K-medoids algorithm with the purpose of getting better clustering. This paper improved the method combined with ACO algorithm we have optimized to gain better routing performance.

The basic idea of traditional K-medoids algorithm is: Firstly, a representative object is randomly chosen for each cluster and the rest objects will be distributed to a proximate cluster according to the distance from the representative object; after that the non-representative object repeatedly substitutes the representative object so as to improve clustering quality. A cost function is used to estimate the average dissimilarity degree between the measurement object and the reference object of the function, which can show the quality of clustering result. But it has drawbacks:

A) Owing to the random selection of initial centre, the clustering result might change with the alternation of initial points, becoming sensitive to the initial centre and is prone to fall into locally optimal solution.

B) In terms of centre replacement, the original K-medoids algorithm involves all the non-representative objects and redistributes them according to dissimilarity function, which can guarantee clustering accuracy, but will cause low search efficiency.

The algorithm is improved according to the following principles aimed to solve the problems existing in traditional K-medoids:

A) Fine adjustment of the initial centre. Namely: Each object inside a cluster alternately becomes the cluster centre; compute the distance sum between the center and other objects and select the object which has the minimum distance sum as the adjusted center of the cluster.

B) The adoption of centre replacement strategy based upon gradually increasing center candidate set. Some relatively optimal candidate points-if they want to replace a centre-are most likely to occur nearby the centre, like the points inside the same or the adjacent cluster. Therefore, the search for new centers can be made only within these regions. The specific replacement method is: The center is replaced $k$ times ($k$ stands for targeted clustering number, the same below). A center candidate set is firstly formed every time before replacement and the candidate set keeps on extending in the iteration process. Suppose $i$ was the present number of replacement times. When $k$ centers were replaced last time, the searching region was set as all the noncentral objects within $i$ clusters (including the corresponding cluster of the replaced centre) closest to the centre. In this way, the candidate set of new centers keep enlarging starting from $1,2,\ldots,k$ clusters, thus forming the candidate set of increment centers. At last, the searching region will extend to all noncentral sets, which is in accordance with the original algorithm. Time complexity:

Through adjustments of initial centers, this new improved algorithm raises the possibility that more centers could become corresponding centers of the final clustering result. In this way, the number of iterations can be reduced and the algorithmic execution efficiency can be improved. Yet through integral adjustments some relatively optimal initial centers can be found, which to a certain extent upgrades the clustering quality. Under normal circumstances, suppose clustering data points were equally distributed, i.e. each cluster had equivalent quantity of data points(about $(n-k)/k$ ) every time when data were partitioned. When replacement went on to round $i$ ($i = 1,2,3\ldots,k$), $i$ proximate clusters would form a center candidate set with the quantity of data points within this candidate set being $i \times (n-k)/k$. Choose one point within this set to replace the present center and partition cluster to the rest $n-k$ noncentral points, then the algorithmic time complexity is $O(i \times (n-k)^2/k)$ . Therefore, the time complexity of replacing and verifying all $k$ centers was $O(i \times (n-k)^2)$ . But it needs $k$ times to complete the forementioned replacement process according to the previous algorithm and the total time complexity is $O(\sum_{i=1}^{k} i \times (n-k)^2)$, namely $\mathrm{O}(k(k+1)(n-k)^2/2)$.The improved algorithm totally needs $k$ rounds of replacement and its average time complexity is $O((k+1)(n-k)^2/2)$. The average time complexity of the improved algorithm, compared with that of the traditional algorithm $O(k(n-k)^2)$, has smaller coefficient though without decreased order of magnitudes. Therefore, it can definitely save execution time, which will be proven by subsequent experiments.

*3.2 The Improvement of Ant Colony Algorithm*

Ant Colony Optimization (ACO) algorithm, first proposed by M.Dorigo and other scholars in 1991, is a sort of swarm intelligence algorithm inspired by a study of real-life ants feeding behaviors. It is proved that ACO algorithm has been successfully applied not only to solving TSP problem but to optimizing the solution to numerous complex problems like path planning and task scheduling. The traditional ant colony algorithm, though it can reduce nodal energy consumption and strengthen pheromones of the optimal path in the process of pheromone updating, fails to strengthen pheromones of other paths and even leads to constant pheromone volatilization, thus increasing the difficulty of searching and falling into locally optimal solution. As a result, ants will so excessively gather in the optimal path that the energy of network nodes in this path will decrease sharply and the life cycle of the entire network will be shortened.

The traditional ACO algorithm does not take the current energy consumption of nodes into consideration in time of path selection. Therefore, we improve the algorithm to select the next round of head nodes on the basis of considering the remaining energy. Calculate the state transition probability of the ant according to (3) and decide the shifting direction:

$$p^k_{v_m v_n}(t) = \begin{cases} \dfrac{\tau^{\alpha}_{v_m v_n}(t)\eta^{\beta}_{v_m v_n}(t)}{\sum\limits_{\mu \in V_{allowed}} \tau^{\alpha}_{v_m \mu}(t)\eta^{\beta}_{v_m \mu}(t)} & n \in V_{allowed} \\ 0 & otherwise \end{cases} \tag{3}$$

Parameters $\alpha$, $\beta$ respectively represent the pheromone level and energy consumption weight. The value of pheromone is the reciprocal of energy distance between $v_m$ and $v_n$:

$$\tau^{\alpha}_{v_m v_n}(t) = \frac{1}{E'_{d_{v_m v_n}}(t)} \tag{4}$$

$E'_{d_{v_m v_n}}(t)$ is the energy distance from node $v_m$ to node $v_n$, and $E'_{d_{v_m v_n}}(t)$ can be computed by (5) :

$$E'_{d_{v_m v_n}}(t) = \frac{E_{d(v_m, v_n)}}{e_1(v_m) \times e_2(v_m, v_n)} \tag{5}$$

And in (5):

$$e_1(v_m) = \frac{E_{cur}(v_m)}{E_{init}}$$

$$e_2(v_m, v_n) = \frac{E_{estimate}(v_m, v_n)}{E_{init}}$$

$$E_{d(v_m, v_n)} = E_{tx-elec} \times k + \varepsilon_{amp} \times k \times d^2_{v_m v_n}$$

According to (6):

$$E_{estimate}(v_m, v_n) = E_{init} - \frac{E_{init} - E_{estimate}(v_m, v_n)}{time(v_m, v_n)} \times [time(v_m, v_n) + 1] \tag{6}$$

$time(v_m, v_n)$ is the time spent from $v_m$ to $v_n$ and $E_{init}$ represents the primary energy of each node, while $E_{estimate}(v_m, v_n)$ represents the estimated energy needed to be consumed from node $v_m$ to node $v_n$.

Therefore, when a node needs to transmit data to another node, it will evaluate the remaining energy of the neighbouring nodes, and update the routing table dynamically.

*roundupdated* stands for the given fixed cycle index. After a fixed cycle, pheromones will be adjusted as (7):

$$\eta(v_m, v_n) = \eta(v_m, v_n) \times (1 - \rho) \tag{7}$$

The pheromone of each node should be strengthened according to (8). And select the neighbouring node with the most energy as the next cluster head.

$$\eta(v_m, v_n) = E_{d(v_m, v_n)} + \eta(v_m, v_n) \tag{8}$$

Above are the improvements of traditional ACO algorithm.

### *3.3 Data Fusion Algorithm on the Basis of ACO and improved K-medoids*

The steps of this algorithm are as follows:

Step1: Select the number of ants and therein lies $w$ initial centers. Suppose $(M_1, M_2, \ldots, M_w)$ as original cluster head set; $Num\_max$ represents maximum iterations.

Step2: Calculate the Euclidean Distance between each ant. To decide the clustering centre and optimal path, we need to consider both the shortest path and node remaining energy according to (3), and make the centre the historical optimal position.

Step3: Using the optimized K-medoids algorithm above to begin new clustering computation for the historical optimal position to confirm each ant's cluster and new cluster centre.

Step4: Compute the optimal solution in the new ant colony according to the method in Step2. At the same time, update the global optimal solution and historical optimal position.

Step5: Recompute the Euclidean Distance between each ant, and determine the new cluster centre $O_j$ and the optimal path.

Step6: If we get the optimal path or cluster centre, end the clustering. Otherwise, go to Step3.

Step7: Member nodes in each cluster transmit packets to cluster heads directly after finishing the clustering. Then cluster heads choose the optimal path following (5) to transmit some related packets to the base station.

Step8: Each cluster collects data respectively, and when the number of collecting data is multiple of *roundupdated* , we must rechoose new cluster heads according to the remaining energy of the nodes.

According to the results of clustering, we could refer to the cluster heads changing method of LEACH, and begin data transmission. Each cluster head node broadcast the information of having been the cluster head to the other member nodes in the respective cluster, according to CSMA protocol. Cluster nodes will join the cluster upon receiving the information above. During the periods of stable operation, cluster nodes will send data to cluster heads by CSMA protocol. Each cluster node will transmit data to the cluster head within its own time slot in accordance with a TDMA timetable. The cluster head begins to transmit data in the direction of first-of-chain after receiving and fusing all the data from cluster nodes. Each cluster head node will fuse the data got from the node of previous level and then transmit the data to the next hop of node.

## 4. EMULATION AND COMPARISON

The emulation in this paper is carried out on the platform of Matlab. Emulation conditions include: The base station is set up at (0, 0). 500 sensor nodes are randomly distributed in a square area (500 m * 500m). Some related parameter values are shown in Table 1:

**Table 1. Values of some related parameters**

| Parameter | Value |
|---|---|
| *Roundupdated* | 20 |
| $E_{tx-clec}$ | 50 nJ |
| $\varepsilon_{fs}$ | 10 pJ |
| $Num\_max$ | 2000 |
| $w$ | 1000 |
| $E_{init}$ | 1 J |
| $\alpha, \beta$ | $\alpha = 2, \beta = 2$ |
| $\rho$ | 0.2 |
| *Data packet size* | 4000 bits |
| $\varepsilon_{mp}$ | 0.0013 pJ |
| $E_{DA}$ | 5 nJ |

*4.1 The Optimal Cluster Head Number*

We assume that $E_{Total}$ will be consumed in every round's transmission. So under ideal conditions, the total rounds are:

$$R = E_{Total} / E_{Round} \tag{9}$$

If the total energy of the network is fixed, then the life cycle will depends on energy consumption in each round. Suppose that each cluster member node send *k* bits of data to the cluster head in each round. And after data fusion, the cluster head also send *k* bits of data outward. Energy consumption in each round will be:

$$E_{Round} = k(2NE_{elec} + NE_{DA} + n_{cus}\varepsilon_{amp}d_{hs}^{\ 4} + N\varepsilon_{fs}d_{nh}^2) \tag{10}$$

$d_{hs}$ is the average distance from cluster heads to the observer. $d_{nh}$ represents the average distance from member nodes to cluster heads. $n_{num}$ is the final number of cluster heads. *N* means the total of sensor network nodes. If sensor nodes are randomly distributed inside a square whose length of side is *M*, then we get:

$$d_{nh} = \frac{M}{\sqrt{2\pi n_{num}}} \tag{11}$$

Put (11) into (10). Make the first derivative of $E_{Round}$ 0, then calculate the optimal number of cluster heads:

$$n_{numopt} = \sqrt{\frac{N}{2\pi}}\sqrt{\frac{\varepsilon_{fs}}{\varepsilon_{amp}}}\frac{M}{d_{hs}^{\ 2}} \tag{12}$$

$n_{numopt}$ represents the final optimal number of cluster heads. Eventually the optimal cluster head number is calculated and should be 25. Specific clustering result is shown in Fig. 1.

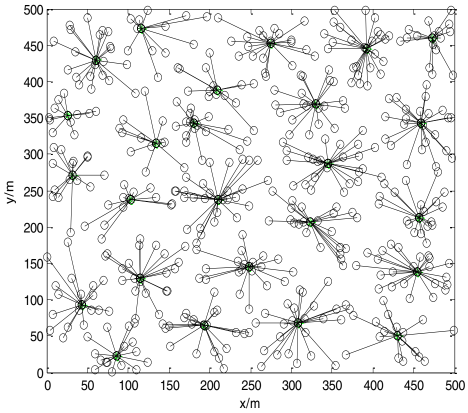


Fig. 1. Clustering result in this paper

*4.2 Time Complexity*

In Step 1, the time complexity is:

$$T_1 = O(w * Num\_max) = O(1000 * 2000)$$

In Step 2, the time complexity of Euclidean Distance is:

$$T_2 = (d^2 \ * w) = (d^2 * 1000)$$

In Step 3, the time complexity of updating of cluster centres and reference nodes is:

$$T_3 = (n * k) = (500 \times 25)$$

The total time complexity is:

$$T_{total} = (d^2 \ * w + w * Num\_max + n * k)$$
$$= (d^2 \ * 1000 + 1000 \times 2000 + 500 \times 25)$$

*4.3 Space Complexity*

Assume there are $w$ ants, $k$ cluster, and $n$ nodes, which are randomly distributed in the network. The space complexity will be:

$$T_{space} = O(w * k * n) = O(1000 \times 25 \times 500)$$

*4.4 The Distribution of Dead Nodes*

In order to demonstrate the feasibility and validity of the algorithm proposed in this paper, experiments are made to emulate such algorithms as LEACH, HEED and traditional K-medoids, whose performance is evaluated by the life cycle and residual energy of the network. Let's observe the situation of nodes distribution in the emulation process when the failure rate of nodes (i.e. the proportion of number of nodes running out of energy to the total number of nodes) reaches 60%. As shown in Fig. 2, Circles stand for non-failure nodes and the black spots represent dead nodes. It is shown in Fig. 2 that dead nodes' localizations are reasonably distributed in the entire network. The protocol uses a communication mode of combining single hop and multiple hop. There aren't extensive dead nodes in the region far away from Sink node, which meets the preliminary design requirements.

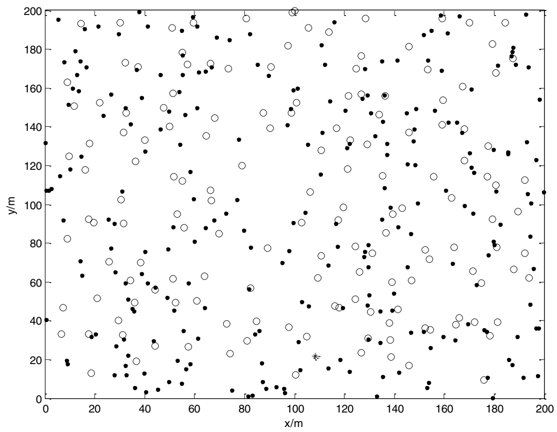


Fig. 2. Nodes death map

*4.5 Energy Consumption Speed Comparison*

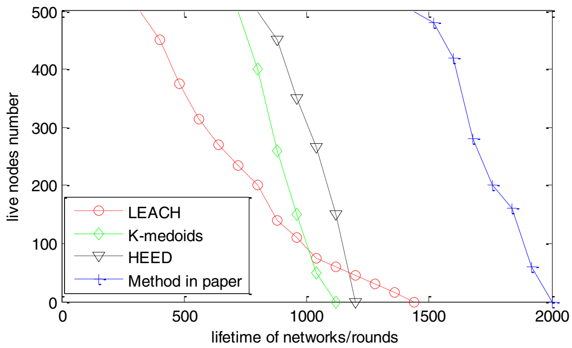


Fig. 3. Performance comparison

Fig. 3 showed the energy consumption speed comparison of HEED, LEACH, and algorithm introduced in this paper. The traditional LEACH rechooses head nodes about every 20 rounds. But due to the uncertainty of head nodes' locations and amount, energy consumption of different head nodes will be quite different, which will bring the problem of energy consumption imbalance in the WSN. Algorithm of this paper also rechooses head nodes about every 20 rounds. But the energy consumption can be balanced and the life cycle of the network can be extended, due to equal number of member nodes.

ACKNOWLEDGEMENTS

This work was supported in part by the National Natural Science Foundation of China under Grant 61333003.

REFERENCES

Dorigo, M. and Gambardella, L.M. (1997). Ant colony system: a cooperative learning approach to the traveling salesman problem. *IEEE Trans. On Evolutionary Computation*, **1(1),** 53-66.

Ji, W.Y., Shi, C.Q. and Zou, J. (2012). Routing algorithm of cluster heads selection in regions based on ant colony algorithm. *Journal of Changsha University of Science & Technology (Natural Science )*, **9(2),** 75-80.

Liang, H.W. and Chen, W.W. (2007). ACO-based routing algorithm for wireless sensor networks. *Chinese Journal of Sensors and Actuators*, **20(11),** 2450-2455.

Liao, W. and Kao, Y. (2008). Data aggregation in wireless sensor networks using ant colony algorithm. *Netw. Comp. Appl*, **31(4),** 387–401.

Liu, C.T. (2010). Cluster-based routing protocol for multi-convergent points in WSN based on ant colony optimization. *Computer Applications and Software*, **27(11),** 209-212.

Peng, L., Dong, G.Y. and Dai, F.F. (2014). Improved ACO-K clustering routing algorithm for wireless sensor networks. *Taylor & Francis Group*, in press.

Ma, Q. and Xie, J.Y. (2012). New K-medoids clustering algorithm based on granular computing. *Journal of Computer Applications*, **32(7),** 1973-1977.

Meng, Y. and Luo, K. (2012). K-medoids clustering algorithm method based on differential evolution. *Application Research of Computers*, **29(5),** 1651-1653.

Wang, G.F. and Wang, Y. (2010). Clustering routing algorithm for wireless sensor network based on ant colony. *Computer Engineering*, **36(18),** 73-75.

Wang, T.J. and Yang, Z. (2007). An adaptive data fusion routing algorithm based on genetic algorithm for wireless sensor networks. *Journal of Electronics & Information Technology*, **29(9),** 2244-2247.

Wang, X., Wang, S. and Jiang, A.G. (2007). Cluster-based data fusion decision method in wireless sensor networks. *Control and Decision*, **22(11),** 1208-1212, 1217.

Heinzelman, W.R.,Chandrakasan A.P. and Balakrishnan, H. (2002). An application-specific protocol architecture for wireless micro sensor networks. *IEEE Transactions on Wireless Communication*, **1(4),** 660 – 670.

Xia, N.X., Su, Y.D. and Qin, X. (2010). Efficient K-medoids clustering algorithm. *Application Research of Computers*, **27(12),** 4517-4519.