Perceptron-based AOW Clustering Algorithms

Huozhu Wang [1, a], Jianguo Yu[2,b]

[1] Beijing University of Posts and Telecommunications, Beijing 100876,China

[2] Beijing University of Posts and Telecommunications, Beijing 100876,China

[a]huozhuwang@bupt.edu.cn, [b]yujg@bupt.edu.cn

Keywords: Clustering Algorithms; Ad hoc network; adaptive on-demand weighted algorithm;

Abstract. Clustering Algorithms has been widely used in Ad hoc network with its ability to construct network quickly, conveniently and flexibly and without the need of default network infrastructure. In this paper, Firstly, some shortcomings of the typical algorithm AOW (adaptive on-demand weighted algorithm) are introduced and analyzed. Then, we discusses the calculating method of nodes weights with perceptron algorithm and the cluster heads selecting process with modified algorithms based on original AOW to meet system requirements. So, a adaptive on-demand weighted algorithm based perceptron (PerAOW) is proposed to select cluster heads in Ad hoc network. Compared to AOW, simulation results proved that the proposed algorithm are improving network topological structure and giving 5.2% better load balance factor (LBF).

1. Introduction

Clustering has been used for Ad hoc network because of its ability to construct network quickly, conveniently and flexibly and without the need of default network infrastructure in mobile adhoc networks [1]. Considering the effect to the Ad hoc networks performance of node mobility, various clustering algorithms can be divided into clustering algorithms suitable for static environment, low speed environment, and middle-high speed environment. As clustering algorithms suitable for static environment, there are clustering algorithms based on connected dominating sets, and based on geographic Information [2]. As clustering algorithms suitable for low speed environment, there are lowest identification clustering algorithm[3], highest degree clustering algorithm [3], Weighted Clustering Algorithm (WCA)[2], adaptive on-demand weighted algorithm(AOW)[4], and passive clustering algorithm[5]. As clustering algorithms suitable for middle-high speed environment, there are minimum relative moving speed clustering algorithm [6], maximum speed similarity clustering algorithm [7], and affinity propagation clustering algorithm [8].

Compared to the typical algorithm AOW, the goals of the proposed algorithm are improving network topological structure and increasing load balance factor (LBF). So we proposed a adaptive on-demand weighted algorithm based on perceptron (PerAOW). The proposed algorithm takes the neighbor numbers, average distance, velocity, battery energy and transmitting power of mobile nodes to select cluster headers into consideration.

Our algorithm differs from AOW in which it calculate weighting factors with the help of perceptron algorithm instead of manual setting to select cluster heads. The result of simulations shows that the proposed algorithm provides better performance than AOW in terms of network topological structure and increasing load balance factor (LBF). The rest of this paper is organized as follows. Section 2 review some drawbacks in AOW proposed previously. Section 3 presents the proposed algorithm. Section 4 analyzed performance of the proposed algorithm. Finally, Section 5 concludes this paper.

2. Related works

In AOW[4], a weight for each node indicates the extent to which it fits acting as a cluster head. Weight can be computed by Eq.1 :

$$\text{weight}=C_1*\text{degree}+C_2*\text{distance}+C_3*\text{mobility} +C_4*\text{energy}+C_5*\text{power}+\text{other} \quad (1)$$

Thereinto, other indicates other possible factors, such as processing power and storage space and so on.Weighting factors $\{C_1, C_2, C_3, C_4, C_5\}$ are weighting factors and their values are set by manpower. Which parameter is bigger is decided by people. In this paper, we proposed a Perceptron-based Adaptive On-demand Weighted algorithm (PerAOW) that weighting factors are calculated by training nodes with perceptron algorithm without manual intervention, they can reflect Ad hoc networks environment more accurate than AOW.

In AOW, the distance of the ith node is calculate by Eq.2 :

$$\text{D}_\text{i} = \sum_{j\in Neb_2(i)} \{Dis(i, j)\} \quad (2)$$

but in PerAOW, we adjust Eq.2 to Eq.3:

$$\text{D}_\text{i} = \frac{\sum_{k\in Neb_2(i)} \{Dis(i, j)\}}{Num\{Neb_1(i)\}} = \frac{\sum_{j\in Neb_1(i)} \{Dis(i, j)\} + \sum_{k\in Neb_1(\text{j})} \{Dis(j,k)\}}{Num\{Neb_1(i)\}} \quad (3)$$

Thereinto, $Neb_1(i)$ indicates 1-hop neighbor nodes set of the ith node. $Num\{Neb_1(i)\}$ indicates the number of 1-hop neighbor nodes of the ith node. $Neb_2(i)$ indicates 2-hop neighbor nodes set of the ith node. The 2-hop is chosen because the complexity of breadth first search is $O(n^2)$. And the following Fig.1 will illustrate the adjustment.

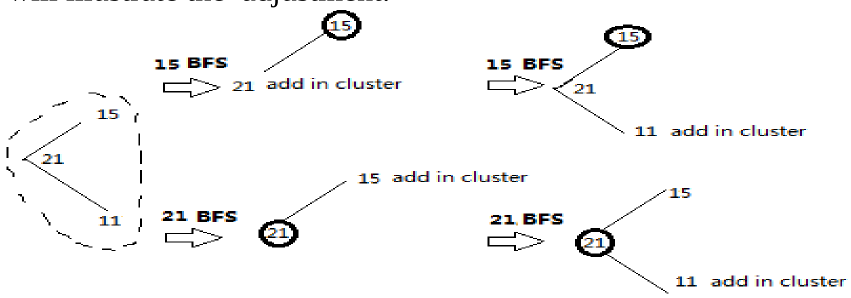

Fig. 1 the 15th and 21st node search for 2 hop neighbors

If we use Eq.3 to calculate the degree of the ith node,

After the breadth first search of the 15th node, the degree of the 15th node is:

$$D_{15} = \frac{\sum_{k\in Neb_2(15)} \{Dis(15,k)\}}{Num\{Neb_1(15)\}} = \frac{Dis(15,21) + Dis(21,11)}{1} \quad (4)$$

After the breadth first search of the 21st node, the degree of the 21st node is:

$$D_{21} = \frac{\sum_{k\in Neb_2(21)} \{Dis(21,k)\}}{Num\{Neb_1(21)\}} = \frac{Dis(21,15) + Dis(21,11)}{2} \quad (5)$$

we can see $D_{21} < D_{15}$ from Eq.4 and Eq.5, Obviously, the 21st node which is the central of network area is more suitable to becoming cluster heads than the 15th node. Contrary , if we use Eq.2 to calculate the degree of the ith node, we can get $D_{21} = Dis(21,15) + \text{Dis}(21,11)$ and $D_{15} = Dis(15,21) + \text{Dis}(21,11)$ from Fig. 2 , so $D_{21} = D_{15}$ will lead to uncertain cluster head, or the smaller node sequence number 15 will be elected wrongly.

In AOW, the mobility of the ith node is calculate by Eq.6 :

$$\text{M}_\text{i} = \frac{S}{T} = \frac{1}{T}\sum_{t=1}^{T} \sqrt{(X_t - X_{t-1})^2 + (Y_t - Y_{t-1})^2} \quad (6)$$

Thereinto, (X_t, Y_t) indicates the coordinate of the ith node at the time of t. But in PerAOW, we adjust Eq.6 to Eq.7 :

$$M_i = \frac{\sum_{j \in Neb_1(i)} R_i(j)}{Num\{Neb_1(i)\}} = \frac{\sum_{j \in Neb_1(i)} \left| P_t^{ij} - P_{t-1}^{ij} \right|}{Num\{Neb_1(i)\}} \quad (7)$$

Thereinto, P_t^{ij} indicates the power of the ith node sent from the jth node at the time of t. Fig.2 will illustrate the adjustment.

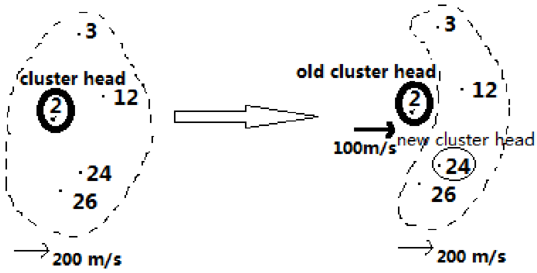

Fig. 2 the speed of old cluster head is decrease

Eq.6 indicates absolute velocity of the ith node, while Eq.7 indicates relative velocity of the ith node. Fig. 2 can describe stability of cluster heads more clearly especialy when the speed of old cluster head is decrease.

3. The proposed algorithm

In this section, the proposed Perceptron-based Adaptive On-demand Weighted algorithm (PerAOW) is presented to improve network topological structure and increasing load balance factor (LBF). PerAOW algorithm is depicted briefly as follows:

1) Initially, each node i is in undetermined state, they broadcasts a $hello(i, x_i, y_i, z_i)$ message which contains its ID and position coordinates (x_i, y_i, z_i) to notify its presence to its neighbors. Then, each node i which receive $hello(j, x_j, y_j, z_j)$ message can learning d_i, $Dis(i, j)$ and P_t^{ij},where d_i is the number of its neighboring nodes, $Dis(i, j)$ is the distances between node and its neighboring nodes j which can be calculate by $Dis(i, j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}$, P_t^{ij} is the signal power from node j at the time t which can be detected by receiver of node i.

2) Then Each node i calculate its first weight which is the difference between its node degree and total nodes number, that is $\Delta_i = |d_i - N|$. where N is the total nodes number in Ad hoc network. The second weight of node i is D_i metioned in Eq.3. The third weight of node i is M_i metioned in Eq.7. The fourth weight of node i is the consumption of battery (E_i) can be got from the battery of node i. The fifth weight of node i is the antennas transmit power (P_i) can be got from the antennas of node i. The fifth weight is added because we hope the the antennas transmit power of cluster heads is more bigger than ordinary member nodes.

3) Each node i broadcasts a weights message which contains its five weights in step 2), they compare their five weights $\{\Delta_i, D_i, M_i, E_i, P_i\}$ with its neighboring nodes and find out maximum value $\{\Delta_{max}, D_{max}, M_{max}, E_{max}, P_{max}\}$ and minimum value $\{\Delta_{min}, D_{min}, M_{min}, E_{min}, P_{min}\}$ of every kind of weight. Each node i receive the five maximum value and the five minimum value and normalize its five weights by Eq.8.

$$\Delta_i^* = \frac{\Delta_i - \Delta_{min}}{\Delta_{max} - \Delta_{min}}, D_i^* = \frac{D_i - D_{min}}{D_{max} - D_{min}}, M_i^* = \frac{M_i - M_{min}}{M_{max} - M_{min}}, E_i^* = \frac{E_i - E_{min}}{E_{max} - E_{min}}, P_i^* = 1 - \frac{P_i - P_{min}}{P_{max} - P_{min}} \quad (8)$$

4) Put training nodes which cluster relationship and five weights are known in advance in Ad hoc networks, then , we can calculate five weighting factors $\{C_1, C_2, C_3, C_4, C_5\}$ by perceptron algorithm. Thereinto, $C_1+C_2+C_3+C_4+C_5=1$ must be met.

5) Each node i calculate its combined weight $W_i = C_1\Delta_i^* + C_2D_i^* + C_3M_i^* + C_4E_i^* + C_5P_i^*$.

6) Each node i broadcasts the number of its neighboring nodes (d_i). Then all node maintain a array(ID_Des_Array) which stored node ID that sorted by each neighbor nodes number.

7) Get array elements of ID_Des_Array as the node ID successively, the node with the smallest W_i among neighboring nodes is elected as cluster-head. Node with the smallest node ID acts as cluster-head if there are several nodes with the same W_i. Then cluster-head broadcast messages to declare itself. Nodes which receive Cluster(ID) messages will become member-node of the cluster of the cluster-head node and will not participate in remaining cluster procedure.

4. Performance comparison and analysis

In this part, we present the simulation the proposed algorithm(PerAOW), then make performance comparision with that of AOW. We simulate a system of N nodes on an 1000m ×1000m area. The value of N was varied between 10 and 150. Transmission range for each node was varied between 100m and 1000m. Fig.4 show an example of PerAOW. In Fig.4 (a) shows neighbor relation of 30 nodes before clustered with 20 training nodes, Dots represent training nodes and triangle represent nodes to be clustered. Fig.4(b) shows the scenario after using PerAOW.

Firstly, we put four groups of training nodes marked with labels $\{1,2,3,4,5\}$ in Ad hoc network. We have already known cluster-heads are nodes marked with labels {5} and members are nodes marked with labels $\{1,2,3,4\}$ in advances. So all weights and cluster relationship of 20 training nodes are input of perceptron algorithm. Five weight factors $\{C_1, C_2, C_3, C_4, C_5\}$ are output of perceptron algorithm. The iterative process is shown in Figure 4(a). Initial values of five weight factors are same 0.2 because we considered all weight are of equal importance. Then with the iterative process, five weight factors are changed by 20 groups of training nodes weights. After 25 iterations, five weight factors finally trends to be five constant values. $\{C_1, C_2, C_3, C_4, C_5\} = \{0.02109, 0.77329, 0.042728, 0.059707, 0.11424\}$. From the result, we found that the second weight factor is much more bigger than other four weight factors. That is to say, the second weight(D_i) should be given much more importance. Fig.4(b) shows the neighbor relation of 30 nodes after using PerAOW with the help of $\{C_1, C_2, C_3, C_4, C_5\}$. Square represent cluster-head and triangle represent cluster-member.

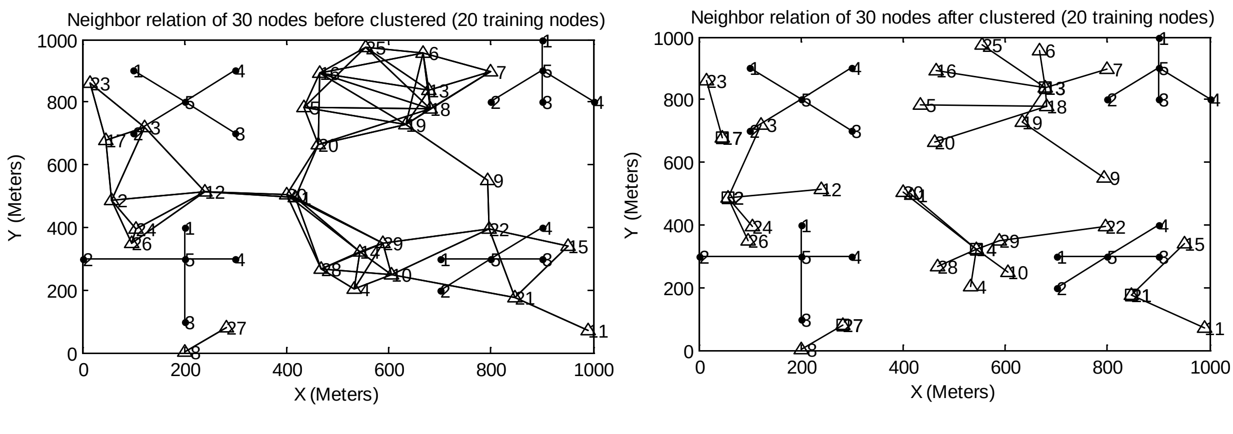

Fig. 3 An example of PerAOW

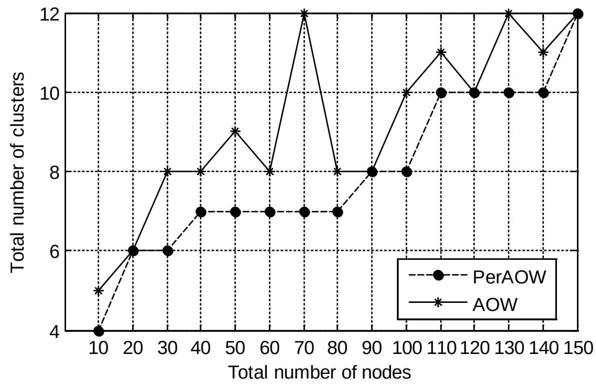

Fig. 4 Weight factor and Total number of clusters

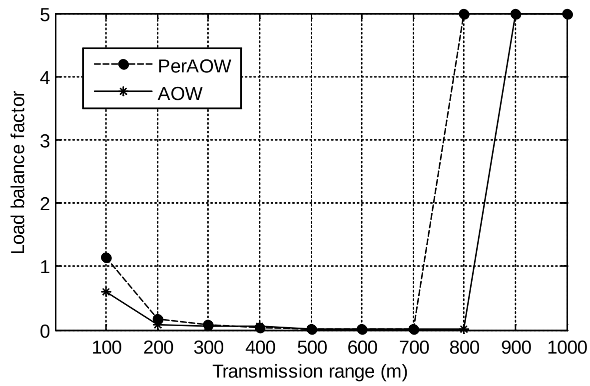

Fig. 5 Average number of nodes in the cluster and Load balance factor

Fig.4(b) shows the total number of clusters with respect to the total number of nodes in the Ad hoc network. The total number of clusters increased as the total number of nodes increased. As we can see in Fig.4(b), the proposed algorithm (PerAOW) produced less clusters than AOW, because the ID of nodes has been sorted according to the number of neighbor nodes before comparing total weight(W_i) in PerAOW. Then we elect cluster-heads of nodes in the order of the number of neighbor nodes. In this way, large clusters will not be broken into small clusters as far as possible, so that the number of clusters can be greatly reduced.

Fig.5(a) shows average number of nodes in the cluster with respect to the total number of nodes in the Ad hoc network. Average number of nodes in the cluster increased as the total number of nodes increased. There are more nodes in each cluster produced by PerAOW than AOW, because distance weight of nodes corrected by Eq.3 and mobility weight of nodes corrected by Eq.7 make total weight (W_i) of nodes more reasonable in PerAOW. The interval between the total weight of each node becomes smaller due to normalizations of five weight in PerAOW, otherwise it is meaningless to add five weight of different units without normalizations.

Fig.5(b) shows Load Balancing Factor (LBF) of network with respect to transmission range in the Ad hoc network. LBF is defined as the reciprocal of the variance of the number of members of the cluster, that is $\mathrm{LBF} = \frac{n_c}{\sum_{i=1}^{n_c}(x_i - u)^2}$ [9],As we can see in Fig.5(b), it is a saddle curve that was high at the beginning and at the end, There are 3 reasons are as follows: (1)each node is cluster head because transmission range is small at the beginning; (2) some cluster is embracing more nodes gradually as transmission range becoming lager; (3) there only exist a cluster contains all nodes when transmission range is big enough at the end, LBF is infinite in this scene, we use 5 to represent infinity in order to present Fig.5(b). The greater LBF represents the network load balancing performance is better. The proposed algorithm(PerAOW) produced 8.9% better LBF than AOW when transmission range was 300m. Also, when transmission range was 500m, our algorithm(PerAOW) gave 7.4% better LBF than AOW. And PerAOW provided 5.2% better LBF than AOW if transmission range was 500m. As a result, The LBF of PerAOW is greater than that of AOW since we calculate weighting factors with the help of perceptron algorithm instead of manual setting to select cluster heads.

5. Conclusions

Simulation results show that the proposed algorithm(PerAOW) provides better performance than AOW in terms of network topological structure and giving 5.2% better load balance factor (LBF), then achieve high performance (high throughput, low cost, low-delay).

ACKNOWLEDGMENT

This work was supported by NSFC under Grant No. 61531007.

References

[1] Basagni, S., “Distributed clustering for ad hoc networks,” in Proceedings of I-SPAN '99, 1999, pp.310-315

[2] Chatterjee, M., Das, S., and Turgut, D., “WCA: a weighted clustering algorithm for mobile ad hoc networks,” Journal of Cluster Computing (Special Issue on Mobile Ad hoc Networks), 5,2002, pp. 193-204.

[3] Gerla, M, and Tsai, J.T.C, “Multicluster, mobile, multimedia radio network,” ACM/Baltzer Journal of Wireless Networks 1, 3, 1995, pp. 255-265.

[4] Haitao Wang, Shaoren Zheng. Clustering Algorithms of MANET and Performance Comparisons.Journal of PLA University of Science and Technology, 2004, 5 (3): 28-32 .

[5] KWON T.J., GERLA M., VARMA V.K., Efficient flooding with passive clustering — an overhead-free selective forward mechanism for ad-hoc/sensor networks. Proceedings of the IEEE,2003, 91(8): 1210 – 1220

[6] BASU P., KHAN N., LITTLE T.D.C., A mobility based metric for clustering in mobile ad hoc networks. International Conference on Distributed Computing Systems Workshop, 2001 , 413 –418.

[7] HU Xi, WANG Jinkuan, WANG Cuirong. Stable Clustering Based on Motion Similarity Evaluation in Mobile Ad Hoc Networks. Third International Symposium on Intelligent Information Technology Application Workshops, 2009, 234 – 237.

[8] SHEA C., HASSANABADI B., VALAEE S., Mobility-Based Clustering in VANETs Using Affinity Propagation. Global Telecommunications Conference, 2009, 1 – 6

[9] Haitao Wang; Hui Chen; Lihua Song, "Improved AOW clustering algorithms for wireless self-organized network and performance analysis," in Cross Strait Quad-Regional Radio Science and Wireless Technology Conference (CSQRWC), 2012 , vol., no., pp.142-146, 23-27 July 2012.