

Chameleon based Clustering for Wireless Sensor Network

¹Soumitra Das and ²Sanjeev Wagh

¹*Dept. of Computer Engineering, Sathyabama University,
Chennai, Tamilnadu, India*

²*CTiF-Centre for TeleInFrastruktur, Aalborg University, Aalborg, Denmark
¹*soumitra_das@yahoo.com, ²wagh@aaau.dk**

ABSTRACT

One of the most important issues in wireless sensor network is the limited battery power within wireless sensor nodes. The in-network aggregation helps in decreasing the communication cost and energy consumption by aggregating the similar data. In this paper, we propose a novel method Energy Efficient Hierarchical Clustering for Wireless Sensor Network using Chameleon Approach, where the Chameleon algorithm selects both Relative Interconnectivity and Relative closeness, to yield the most similar pair of clusters. The proposed algorithm forms the clusters by using a two-phase algorithm to yield accurate results for these highly variable clusters. During the first phase, Chameleon uses a graph-partitioning algorithm to cluster the nodes into several relatively small sub-clusters. During the second phase, it uses an algorithm to find the genuine clusters by repeatedly combining these sub-clusters to get the final clusters. Simulation results proved that proposed method is more energy efficient as compared to Data Routing for In-Network Aggregation in wireless sensor network.

Keywords: Chameleon, Data Aggregation, DRINA, Energy Efficient Routing Protocols, Energy Efficient clustering, WSN.

1. INTRODUCTION

Recent advancements in Information and Communications Technology have enabled the development of low in cost, low in power, small in size wireless sensor nodes and Wireless Sensor Network (WSN). The power requirement is a very crucial parameter in the design of the algorithms to increase the lifetime of the nodes in WSN. There are various routing protocols proposed in the state of the literatures like multi-path,

location-aided, data-centric, QoS-based, mobility-based, heterogeneity-based, hierarchical routing etc., in which energy efficient routing can be achieved. Performance issues of such strategies in terms of energy consumption have been evaluated by Ehsan et al. [1], Das et al. [17].

Cluster based routing and data aggregation techniques are the most admired technique in WSNs for energy efficiency, as proposed by Singh et al.[2]. These two techniques greatly reduce routing control overhead and the number of packets being transmitted. In the following section, we have presented a brief review of literatures on these two state of the art techniques.

2. RELATED WORK

Ahmad et al. introduced an energy efficient cluster based routing technique to defeat the problem of energy hole and coverage hole [3]. They controlled energy hole and coverage hole problems by setting up density-controlled uniform deployment of sensor nodes and finding out the optimum number of Cluster Heads (CH). Similarly, Krishnamachari, Estrin, & Wicker presented the impact of source-destination deployment location and WSN density on energy consumption [4]. They also presented impact of these parameters on delay in data aggregation. Results show that data-centric routing like clustering presents a significant performance improvement in most of the cases.

Data aggregation is one of the very effective techniques in WSN as it reduces the energy consumption by aggregating multiple similar data packets into a single packet. This ultimately leads to extended lifetime of WSN. Localized self organizing data aggregation method for wireless sensor networks, called Localized Power-Efficient Data Aggregation Protocols (L-PEDAPs) has been proposed by Tan et al. in [5]. This method is based on the tree protocol that mimics minimum spanning tree and computes, using position or distance data of one-hop neighbours. The routing tree is constructed over these spanning trees. The L-PEDAPs also adapts to consider the remaining energy levels of the nodes in order to prolong the WSN lifetime. Yuan et. al. proposed data aggregation by restricting the forwarding nodes for Vehicular Ad Hoc Networks (VANET) [6]. Xu et. al. focused on data aggregation and scheduling problem to minimize the latency. This algorithm is distributed in nature and produces a collision-free data aggregation schedule for wireless sensor networks [7].

Xiang et al. proposed to minimize the network energy uses through joint routing and compressed aggregation [8]. The authors make use of a greedy heuristic to obtain both the optimal and the near-optimal data aggregation trees for small scale network and large scale network, respectively. PriSense [9] is a solution for privacy preservation of data aggregation in public urban sensing systems by Shi et al. It makes use of data slicing and mixing along with statistical methods. Survey of many state of the art data aggregation methods is presented by Maraiya et al. in [10].

Kumar et al. projected energy aware clustering and in network data aggregation technique for heterogeneous WSN [11]. Jiang et al. proposed clustering based on prediction of data which exploits spatial and temporal relationship between WSN data[12]. This reduces the energy consumption by avoiding continuous sensor

data gathering. The author has presented adaptive scheme to enable/disable prediction function. It avoids the requirement for slow node-to-node propagation of aggregated values and makes use of efficient cluster-to-cluster data propagation.

Nature inspired routing algorithms are becoming popular in operations. Karaboga et al. presented energy efficient clustering mechanism inspired from Artificial Bee Colony (ABC) in [13]. ABC makes use of swarm intelligence route data. Nikolidakis et. al. proposed energy efficiency through balanced clustering of nodes and routing of data, which achieves energy efficient routing in wireless sensor networks [14]. This makes use of a Gaussian elimination algorithm to calculate the permutation of nodes that can be selected as cluster heads to increase the network lifetime.

Data Routing for In-Network Aggregation (DRINA) [15], Villas et al. has focused on the least number of control packets for creating a routing tree and reliable data aggregation. It works in three stages: 1) Routing tree formation, 2) Cluster head election 3) Route establishment.

Karypis et. al. proposed clustering algorithm Chameleon-based on the similarity of two clusters, mostly used in data mining [16]. Two clusters are merged into each other only if the relative inter connectivity and relative closeness between the two clusters are high within the clusters. Experimental results show that Chameleon can discover natural clusters that many state of the art clustering algorithms fail to find.

3. Energy Efficient Clustering using Chameleon Approach

3.1 Chameleon: A clustering approach

Clustering is a discovery process that groups a set of data to form clusters. This is a new hierarchical clustering algorithm that tries to rise above the boundaries of the existing clustering algorithms. The Chameleon finds the clusters in the data set by using a two phase algorithm. In the first phase, it uses a graph partitioning algorithm to cluster the data items into a large number of relatively small sub-clusters and the second phase uses a hierarchical clustering algorithm to find the genuine clusters by repeatedly combining together these sub-clusters. The main feature of Chameleon algorithm is that it considers Relative Interconnectivity (RI) and Relative Closeness (RC) in identifying the most alike pairs of clusters. It merges the pairs of clusters that have high RI and RC. If we have two clusters C_i and C_j , the RI is defined as the sum of the weight of the edges that connect vertices in C_i to the vertices in C_j . The RC is defined as an average similarity between the points from the two clusters, which is equal to the average weight of the edges connecting vertices in C_i to the vertices in C_j .

3.2 Chameleon: A clustering approach for proposed system

Existing clustering and data aggregation algorithms do not consider the fact that all sensors deployed in the network are sensing similar parameters, e.g. some sensors sense temperature, whereas others sense humidity. Aggregating temperature and humidity data separately makes logic. Proposed algorithm makes use of the

chameleon clustering algorithm to discover clusters of sensors sensing similar parameters. Initially network is clustered into small size clusters using k-nearest neighbors through similarity measure. Small clusters are merged according to RC and RI Fig.1 shows the clustering process wherein it uses two phase algorithm, which first partitions the nodes into sub clusters and then repeatedly combines these sub clusters to obtain the final clusters.

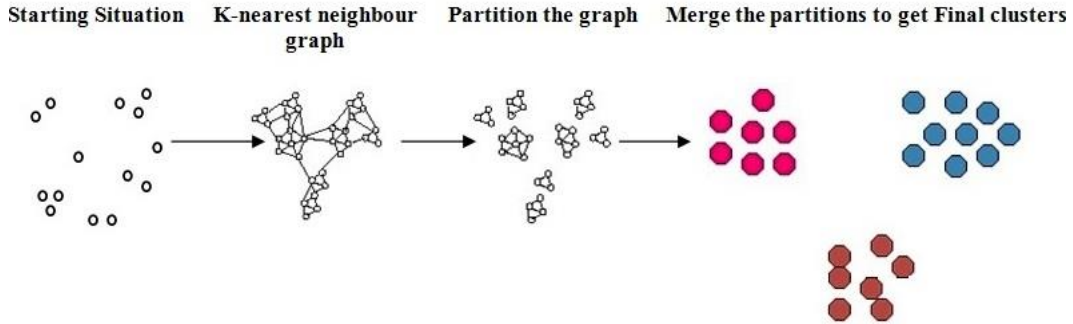


Fig.1 Phases of chameleon technique for WSN cluster formation

3.3 Calculating the Relative Interconnectivity (RI) and Relative Closeness (RC) for WSN [16]

3.3.1 RI and its proof by graph theory

RI between Cluster i and Cluster j is calculated as,

$$RI(i, j) = \frac{2 * |EC(i, j)|}{|IC(i)| + |IC(j)|} \quad (1)$$

Where IC is Internal Connectivity of cluster and Edge Connectivity (EC) is given by,

$$EC(i, j) = \sum_{k=0}^n dist(k) \quad \forall \text{ edges } k \text{ connecting cluster } i \text{ and } j \quad (2)$$

Where n is the number of edges connecting cluster i to cluster j .

Proof:

Let G_k be the sparse graph of k -nearest neighbor, constructed from the given sensor data sets.

$$G_k = \{V_k, E_k\} \quad (3)$$

Where, V_k is a set of sensors and E_k is a set of links between the sensors
Now Cluster set(Cs) is formed based on (3) and mathematically denoted as

$$C_s \subset G_k. \tag{4}$$

Let i and j be the two sub clusters of G_k then from (4), we get

$$i \text{ and } j \subset G_k. \tag{5}$$

Now let say, the set IC as a set of an Internal Connectivity of i and j such that $IC(i)$ and $IC(j) \subset E_k$. [From (5)] (6)

Now let us define an edge-cut of two clusters, as $EC[i,j]$ which is nothing but a weighted sum of links, that partition the cluster into roughly two equal parts. Thus, EC can be rewritten as

$$EC[i,j] = \sum_{k=0}^n \text{dist}(k) \text{ , edges } k \text{ connecting cluster } i \text{ and } j. \tag{7}$$

Now, we can define $RI[i,j]$, a Relative Connectivity $RI[i,j] = 2 * |EC[i,j]| / (|IC(i)| + |IC(j)|)$ [From (7)] (8)

3.3.2 RC and its proof by graph theory

RC is given by,

$$RC(i,j) = \frac{SIM(i,j)}{\frac{|C(i)| * SIM(cj)}{|ci| + |cj|} + \frac{|C(j)| * SIM(ci)}{|ci| + |cj|}} \tag{9}$$

Where Similarity $SIM(i,j)$, is the average of similarity between cluster i and j ,

$$SIM(i,j) = \sum_{k=0}^n \sum_{l=0}^n s(k,l) \text{ ,} \tag{10}$$

where $s(k,l) = \begin{cases} 1 & \text{if type of node } k = \text{type of node } l \\ 0 & \text{otherwise} \end{cases}$

Proof:

The average SIM is calculated between the two clusters i and j as:

$$SIM(i,j) = \sum_{k=0}^n \sum_{l=0}^n s(k,l) \tag{11}$$

$$\text{where, } s(k,l) = \begin{cases} 1 & \text{if type of node } k = \text{type of node } l \\ 0 & \text{otherwise} \end{cases}$$

The RC[i, j] between the two sub clusters i and j can be calculated as;

$$\text{RC}[i, j] = \text{SIM}(i,j) / [(|C_i| * \text{SIM}(C_i) / (|C_i| + |C_j|)) + (|C_j| * \text{SIM}(C_j) / (|C_i| + |C_j|))] \quad (12)$$

[From (5) & (10)]

4. IMPLEMENTATION

The Proposed clustering algorithm is evaluated through simulation in Matlab for its effectiveness. The Proposed algorithm is compared with DRINA in WSN [15]. Simulation parameters used for evaluation are presented in Table 1.; Fig. 2 and Fig. 3 depicts the output of stage 1 and stage 2 of the proposed clustering algorithm.

Table 1. Simulation Parameters

Simulation Parameter	Value
Network Area	100 X 100 Meters
Number of Nodes	0 to 500
Threshold value (RC+RI)	1
Communication Range	10 meters
Initial Energy	100Jules
Receive Power	5-6 Jules/Byte
Transmit Power	6-10 Jules/Byte
Packet Rate	1 packet per Sec
Packet Size	1000 bytes
Number of Rounds	99999

4.1 Chameleon clustering techniques for WSN

1. Read x and y coordinates of all the sensor nodes in the network.
2. Find Euclidean distance between each sensor node.
3. If the distance between two sensor nodes is more than the communication range, then mark distance as infinity.
4. Let C be the set of clusters.
5. Let the number of clusters (nc), nc=0.
6. For each node, repeat the following steps
 - a. Check the distance with each cluster
 - b. If the sensor node is closer than the threshold
 - i. Add sensor node into the cluster and set the flag.
 - c. If the sensor node is not included into any cluster
 - i. Create a new cluster.
 - ii. Increment nc by one.
7. For each cluster formed in step 6, follow the below mentioned steps

- a. Calculate RI and RC with all other clusters.
- b. If $(RC+RI)$ greater than a threshold the merge the clusters
8. Create cluster information packet.
9. Assign cluster Identification to each sensor node.
10. Assign Time To Leave(TTL) to cluster information packet.
11. Broadcast cluster information packet.

After the execution of all the above steps for chameleon clustering technique, the following clusters are formed as shown in Fig. 2 and the final clusters are formed as shown in Fig.3.

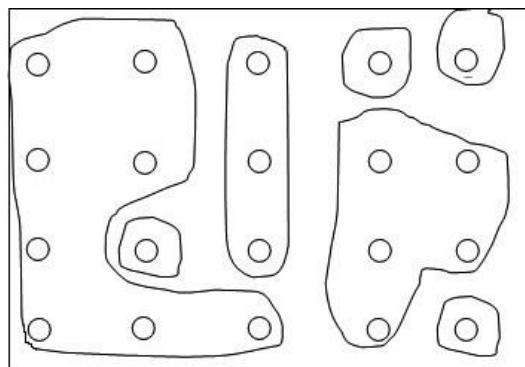


Fig.2 Clusters formed after k-nearest neighbours clustering algorithm for 20 nodes

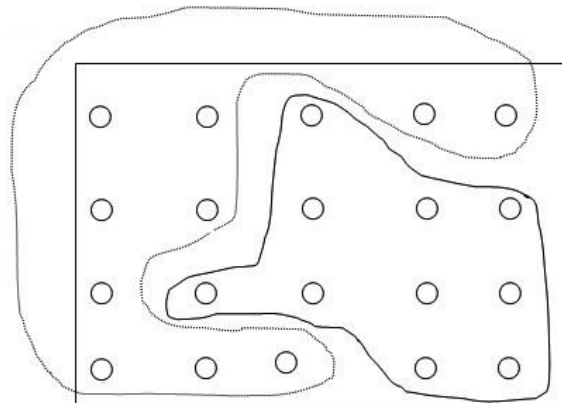


Fig.3 Final clusters in a proposed algorithm after merging clusters for 20 nodes

4.2 Pseudocode for CH selection

1. If the node is sink node
 - a) CH = self address
 - b) hops to sink = 0
- 2) Else

- a) CH = broadcast address
- b) hops to sink = infinite
- 3) If node is the sink node
 - a) send CH election packet (Seqno, node ID, cluster ID, hops to sink)
- 4) Else
 - a) wait for CH election packet.
- 5) If the CH election packet is received from neighbor nodes
 - a) if CH not yet assigned and cluster ID is equal to received cluster ID
CH = node Id in CH election packet
 - b) else
 - if the CH election packet has hops to sink < hops to sink and cluster ID is equal to received clusterID
cluster node = node Id in CH election packet
 - else
neglect CH election packet
- 6) Repeat step 5 for every CH election packet.
- 7) Send acknowledgement to CH.

5. Analysis of the proposed technique for efficiency

1. Let, n : the number of nodes and c : the number of initial sub-clusters.
2. Computing the KNN graph, we get
 - 2.1 Time complexity Low dimensional nodes: $T(n) = O(\log n!)$ and High dimensional nodes: $T(n) = O(n^2)$
3. The graph partitioning algorithm gives: $O(|V| + |E|)$
4. Since using KNN graph: $|E| = O(|V|)$
5. First phase : $O(n \log(n/c))$
6. Splitting each one of the initial c clusters is: $O(n/c)$, leading to an overall complexity of $O(n)$
7. During each merging step: $O(nc)$
8. Finding the most similar pair of cluster: $O(c^2 \log c)$
9. Overall complexity: **$O(nc + n \log n + c^2 \log c)$**

6. RESULTS

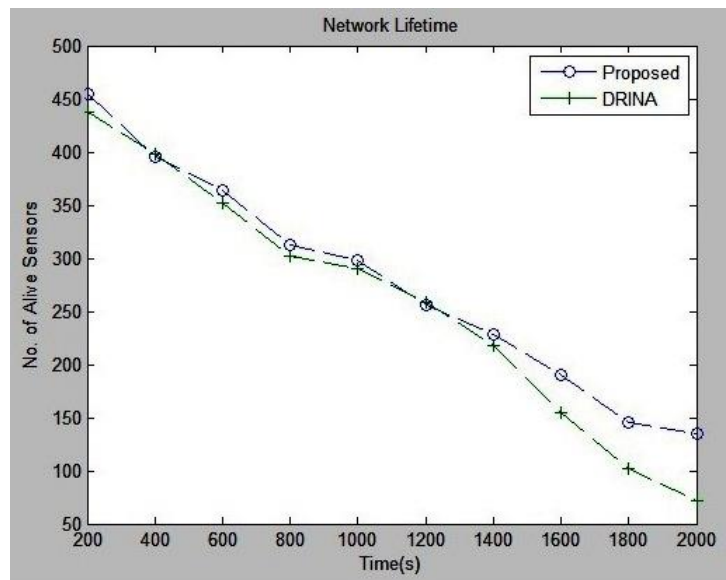


Fig.4 Network lifetime

Table 2. Comparison of number of alive sensors for different protocols

Time (Secs)	Number of Alive Sensors	
	DRINA	Proposed
200	438	455
600	352	364
1000	290	298
1400	218	228
2000	72	135

Fig.4 Shows the graph of number of alive sensors with respect to time in seconds (Network lifetime) for proposed algorithm and DRINA. From Table 2 it is evident that our proposed algorithm has more number of alive nodes as compared to DRINA which clearly indicates that the network life of the proposed algorithm is better than DRINA.

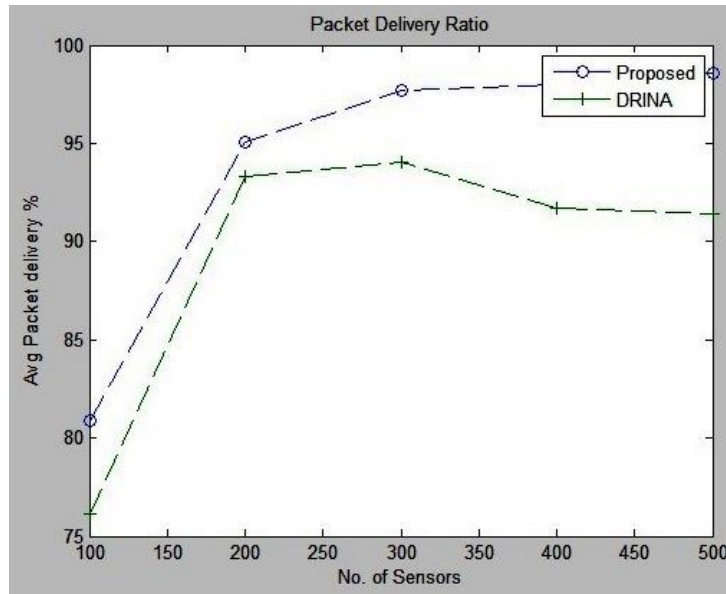


Fig.5 Average PDR

Table. 3 Comparison of average PDR

No. of Sensors	Average PDR	
	DRINA	Proposed
100	76.10	80.89
300	94.03	97.67
500	91.39	98.56

Fig.5 shows the average Packet Delivery Ratio (PDR) with respect to the number of sensors for the proposed algorithm and DRINA. As shown in Table 3, the proposed algorithm has 98.56 percent of an average PDR as compared to DRINA which is about 91.39 percent.

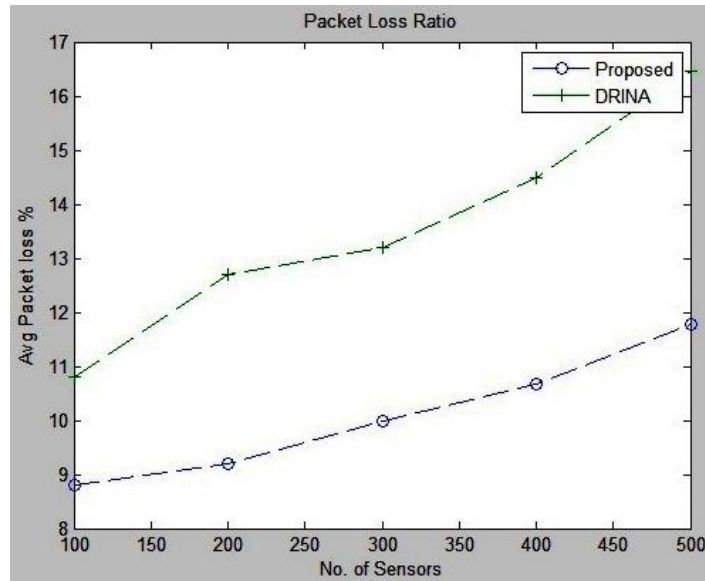


Fig. 6 Average PLR

Table. 4 Comparison of average PLR

No. of Sensors	Average PLR	
	DRINA	Proposed
100	10.8	8.8
300	13.2	10
500	16.45	11.78

Fig.6 shows the average Packet Loss Ratio (PLR) with respect to the number of sensors for the proposed algorithm and DRINA. Table 4 illustrates that the proposed algorithm has 11.78 percent of average PLR which is less than DRINA (16.45 percent).

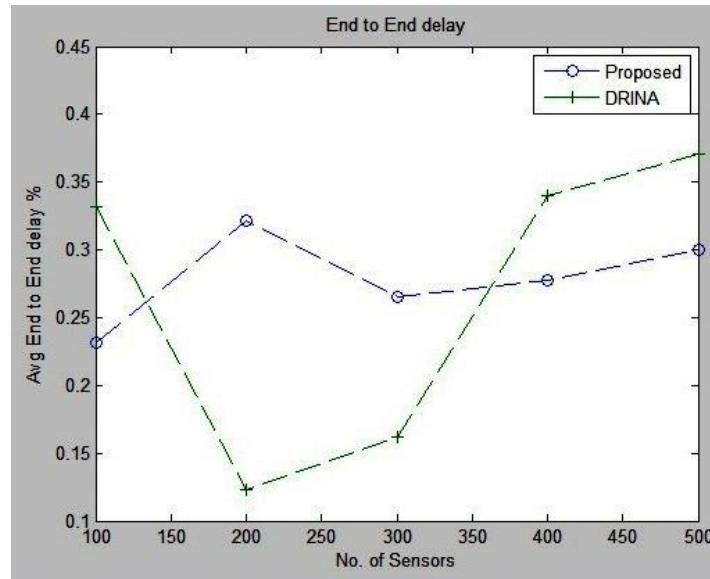


Fig. 7 End to end delay

Table 5. Comparison of the average end to end delay

No. of Sensors	Average End to End Delay	
	DRINA	Proposed
100	0.332	0.232
300	0.162	0.265
500	0.371	0.3

Fig.7 represents the average end to end delay with respect to the number of sensors for the proposed algorithm in comparison to DRINA. Table 5 clearly proves that the proposed algorithm has a similar average end to end delay of 0.3 percent as compared to DRINA which is around 0.371 percent.

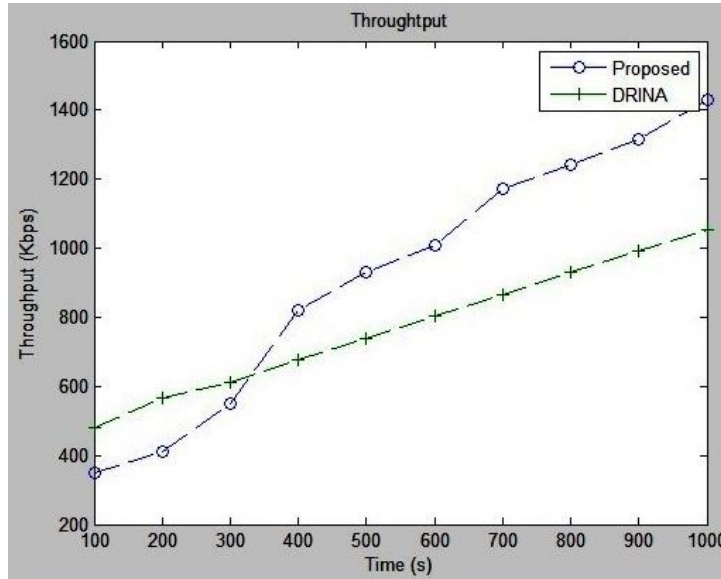


Fig. 8 Throughput

Table.6 Comparison of throughput

Time (Secs)	Throughput (Kbps)	
	DRINA	Proposed
200	566	410
400	675	820
600	803	1010
800	929	1242
1000	1055	1430

Fig.8 illustrates the throughput of the network. From Table 6, it is noticeable that our proposed algorithm is an improved throughput by approximately 1430 percent as compared to DRINA (1055 percent).

CONCLUSION

In this paper, we have proposed an algorithm which forms clusters dynamically according to the position of the node and the type of values it is sensing. Cluster information is propagated to all the nodes and routing is formed. Through simulation results, it is proved that the proposed algorithm is "energy balanced" as well as "energy efficient". The simulation results also clearly indicate that the proposed algorithm improves the lifetime of the network. Currently, we have evaluated our proposed algorithm for approximately 500 sensor nodes. Our future work will be focused to improve the available schemes of energy efficient clustering in wireless sensor network with more number of sensor nodes.

ACKNOWLEDGEMENTS

We would like to express our gratitude to Dr. B.P. Patil and Dr. S.S. Sonavane for their mindful and inventive remarks, all through the readiness of the paper. Last yet not the least, all my colleagues, family members, friends and Dr. Soma for their colossal support.

REFERENCES

- [1] Ehsan, S., & Hamdaoui, B. (2012). A survey on energy-efficient routing techniques with QoS assurances for wireless multimedia sensor networks. *Communications Surveys & Tutorials, IEEE*, 14(2), 265-278.
- [2] Singh, S. K., Singh, M. P., & Singh, D. K. (2010). A survey of energy-efficient hierarchical cluster-based routing in wireless sensor networks. *International Journal of Advanced Networking and Application (IJANA)*, 2(02), 570-580.
- [3] Ahmad, A., Latif, K., Javaidl, N., Khan, Z. A., & Qasim, U. (2013). Density controlled divide-and-rule scheme for energy efficient routing in Wireless Sensor Networks. In *Electrical and Computer Engineering (CCECE), 2013 26th Annual IEEE Canadian Conference on* (pp. 1-4). IEEE.
- [4] Krishnamachari, B., Estrin, D., & Wicker, S. (2002). The impact of data aggregation in wireless sensor networks. In *Distributed Computing Systems Workshops, 2002. Proceedings. 22nd International Conference on* (pp. 575-578). IEEE.
- [5] Tan, H. O., Korpeoglu, I., & Stojmenovic, I. (2011). Computing localized power-efficient data aggregation trees for sensor networks. *Parallel and Distributed Systems, IEEE Transactions on*, 22(3), 489-500.
- [6] Yuan, Y., Luo, J., Yan, W., Zhao, T., & Lu, S. (2014, February). DA2RF: A data aggregation algorithm by restricting forwarders for VANETs. In *Computing, Networking and Communications (ICNC), 2014 International Conference on* (pp. 393-397). IEEE.
- [7] Xu, X., Li, M., Mao, X., Tang, S., & Wang, S. (2011). A delay-efficient algorithm for data aggregation in multi hop wireless sensor networks. *Parallel and Distributed Systems, IEEE Transactions on*, 22(1), 163-175.
- [8] Xiang, L., Luo, J., & Vasilakos, A. (2011, June). Compressed data aggregation for energy efficient wireless sensor networks. In *Sensor, Mesh and Ad Hoc Communications and Networks (SECON), 2011 8th Annual IEEE Communications Society Conference on* (pp. 46-54). IEEE.
- [9] Shi, J., Zhang, Y., & Liu, Y. (2010, March). Prisense: privacy-preserving data aggregation in people-centric urban sensing systems. In *INFOCOM, 2010 Proceedings IEEE* (pp. 1-9). IEEE.
- [10] Maraiya, K., Kant, K., & Gupta, N. (2011). Wireless sensor network: a review on data aggregation. *International Journal of Scientific & Engineering Research*, 2(4), 1-6.

- [11] Kumar, D., Aseri, T. C., & Patel, R. B. (2011). EECDA: energy efficient clustering and data aggregation protocol for heterogeneous wireless sensor networks. *International Journal of Computers Communications & Control*, 6(1), 113-124.
- [12] Jiang, H., Jin, S., & Wang, C. (2011). Prediction or not? An energy-efficient framework for clustering-based data collection in wireless sensor networks. *Parallel and Distributed Systems, IEEE Transactions on*, 22(6), 1064-1071.
- [13] Karaboga, D., Okdem, S., & Ozturk, C. (2012). Cluster based wireless sensor network routing using artificial bee colony algorithm. *Wireless Networks*, 18(7), 847-860.
- [14] Nikolidakis, S. A., Kandris, D., Vergados, D. D., & Douligeris, C. (2013). Energy efficient routing in wireless sensor networks through balanced clustering. *Algorithms*, 6(1), 29-42.
- [15] Villas, L. A., Boukerche, A., Ramos, H. S., de Oliveira, H. A., de Araujo, R. B., & Loureiro, A. A. F. (2013). DRINA: A lightweight and reliable routing approach for in-network aggregation in wireless sensor networks. *Computers, IEEE Transactions on*, 62(4), 676-689.
- [16] Karypis, G., Han, E. H., & Kumar, V. (1999). Chameleon: Hierarchical clustering using dynamic modeling. *Computer*, 32(8), 68-75.
- [17] Soumitra Das, Sanjeev Wagh and Pramod D Ganjewar, "Energy Efficient Cluster based Hierarchical routing protocols in Wireless sensor networks- A Survey", *Multidisciplinary Journal of Research in Engineering and Technology*, 2014,pp.02-22, Vol1 Issue1.