Research Journal of Applied Sciences, Engineering and Technology 7(20): 4200-4205, 2014
DOI:10.19026/rjaset.7.789
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: June 05, 2013 Accepted: December 13, 2013 Published: May 20, 2014

Research Article

Reliable and Energy Efficient Distributed Clustering based Target Tracking Technique for Wireless Sensor Networks

[1]C. Jehan and [2]D. Shalini Punithavathani
[1]Department of IT, Jayamatha College of Engineering, Tamil Nadu, India
[2]Govt. College of Engineering, Tirunelveli, India

Abstract: In WSN most of the target detection and tracking algorithms require the sensors to work in groups in order to advance the consistency of target tracking algorithms. This makes it necessary for deploying sensors to discover and group together so that their coverage can be maximized. In this study we have proposed a distributed clustering algorithm for effectively detecting the Target location. The proposed clustering algorithm is distributed in nature and has the ability to reconfigure in the event of node failure. The algorithm is highly localized and hence does not need flooding across the entire network. Since the algorithm allows for more clusters to track the same region the system reliability is greatly improved. The algorithm builds a series of over-lapping clusters which allow for more than one cluster to track a region. This redundancy improves the overall system reliability. The overlapping clusters also allow for tracking of curvilinear targets.

Keywords: Cluster, energy conservation, route discovery, target detection, WSN

INTRODUCTION

WSNs require methods to protect limited resources, consistent with their defining characteristics: restricted wireless bandwidths, open network architecture of unattended nodes and a large number of densely distributed nodes. Nodes are designed to be expandable with limited processing, storage, transmission and standalone battery resources. Each sensor integrated at a node is a data source that monitors the environment by sampling physical phenomena. The distributed, collaborative processing tasks within Wireless Sensor Networks (WSNs), severely constrained by limited power, computational and storage resources at each sensor node, the wireless bandwidth between nodes and the amount of sensor data shared among communicating nodes, has been thoroughly examined in published studies (Jing *et al*., 2007). Compression is based on the elimination of redundancies within sensor data to reduce energy consumption due to data storage, processing and transmission. Data compression and reduction in the WSN must be scalable to network size and distributed among nodes to conserve and balance those limited resources (Kiran *et al*., 2011).

The purpose of clustering algorithm is to fuse the CPA measurements made by individual sensors at minimum cost (in terms of energy) and at the same time provide the best coverage possible for a given deployment (Oh *et al*., 2006). One way to achieve data fusion is to deliver the CPA measurements to a centralized location. However the transmission of raw data to the centralized location would mean transmitting 4 raw data packets instead of one fused value. This is not economical in terms of energy. Instead, clustering allows for local data fusion. Since each cluster head is only one hop away from its member nodes less energy is consumed in transmission of raw data and also the probability of loosing raw data is reduced. A cluster once formed can track a target only with certain accuracy. The primary sources of error are false alarms at each sensor, loss of raw data and sensor failures. Hence it is necessary to provide redundancy. Redundancy can be achieved by allowing more than one cluster to track the same region. In the proposed algorithm we allow for redundancy by allowing two clusters to share a predetermined number of nodes. By increasing the number of shared nodes more clusters are formed in the same region and hence increases the system reliability (Arora *et al*., 2004; Brooks and Griffin, 2002).

In general, target classification and tracking algorithms rely on information provided by a cluster of sensors. In case of target classification each sensor is equipped with different modalities, such as magnetic, radar, thermal, acoustic, chemical, electric, seismic and optical. Hence the target classification draws its results from observations made by a cluster of modalities. This emphasizes the need for a clustering algorithm that can exploit the redundancy in the sensor deployment and reduce the latency in the exchange of raw data and the amount of raw data that needs to be exchanged (Lung and Zhou, 2010).

Corresponding Author: C. Jehan, Department of IT, Jayamatha Engineering College, Tamil Nadu, India

The proposed clustering algorithm is distributed in nature and the number of clusters to be formed can be easily controlled. Further, since the cluster head chooses its member nodes from its one hop neighbors, the raw data has to travel only one hop. Finally, the target tracking results of each cluster head can be progressively fused with those of its neighboring clusters.

LITERATURE REVIEW

Heinzelman *et al*. (2000) propose LEACH, a substitute clustering based algorithm. In order to save energy, LEACH deals with the heterogeneous energy condition is the node with higher energy should have a larger probability of becoming the cluster head. Each sensor node must have an approximation of the total energy of all nodes in the network to compute the probability of becoming a cluster head but it cannot make the decision of becoming a cluster head only by its local information, so the scalability of this scheme will be influenced. Fonoage *et al*. (2010) suggest a new clustering algorithm CODA in order to mitigate the unbalance of energy depletion caused by different distance from the sink. CODA divides the whole network into a small number of groups based on the distance from the base station and the strategy of routing and each group has its own number of cluster members and member nodes. The farther the distance from the base station, the more clusters are formed in case of single hop with clustering. It shows better performance than applying the same probability of the whole network in terms of the network lifetime and the dissipated energy.

Watfa *et al*. (2009) developed an algorithm based on a chain, which uses a greedy algorithm to form a data chain. Each node, aggregates data from downstream node and sends it to upstream node along the chain and communicates only with a close neighbor and takes turns transmitting to the base station, thus reducing the amount of energy spent per round. Qi *et al*. (2011) developed a HEED clustering algorithm which periodically selects cluster head based on the node residual energy and node degree and a secondary parameter, such as node proximity to its neighbors or node degree. The clustering process terminates in O (1) iterations and it also achieves fairly uniform cluster head distribution across the network and selection of the secondary clustering parameter can balance load among cluster heads.

Kiran *et al*. (2011) introduce a cluster head election method using fuzzy logic to overcome the defects of LEACH. They inquired that the network lifetime can be prolonged by using fuzz variables in the homogeneous network system, which is different from the heterogeneous energy consideration. In EDGA algorithm to achieve good performance in terms of lifetime by minimizing energy consumption for in-network communications and balancing the energy load. It is based on weighted election probabilities of each node to become a cluster head, which can better handle the heterogeneous energy capacities and adopt a simple but efficient method to solve the area coverage problem in a cluster range.

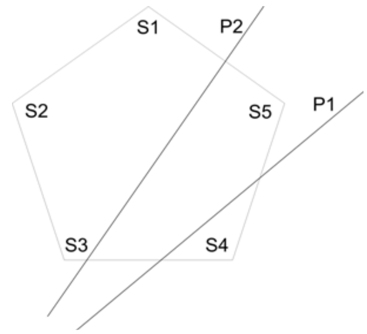


Fig. 1: A cluster formed with five sensors

The impact of heterogeneity of nodes in terms of their energy that are hierarchically clustered in WSNs and initiate an energy efficient heterogeneous clustered method for WSNs based on weighted election probabilities of each node to become a cluster head according to the residual energy in each node (Wang, 2008). For this they suppose a percentage of the population of sensor nodes is equipped with the additional energy resources. The following Fig. 1 shows formation of cluster in five sensors.

NETWORK MODEL AND TARGET DETECTION

The sensor network comprises N nodes placed randomly with uniform distribution over a finite, two-dimensional planar region to be monitored. Each node has a unique identifier and the union of sensing regions of all network nodes guarantees redundant coverage of the region to be monitored (Kaushik and Abhrajit, 2011).

Formation of WSN: A Fixed network which includes in mobile sensor nodes and base station is considered in our study with the following assumptions:

- The network is considered homogeneous and all of the sensor nodes have the same initial energy
- Each sensor node knows its own geographical position
- All nodes measure the environmental parameters at a fixed rate and send it periodically to the receiver nodes
- The radio channel is symmetric such that the energy consumption of data transmission from node A to node B is the same as that of transmission from node B to node A
- Each sensor node can operate either in sensing mode to monitor the environment parameters and

transmit to the base station or cluster head node to gather data, compress it and forward to the BS:

$$E_{Tx} = E_{elec} * l + E_{fs} * l * d^2, \ d < d_0$$

$$E_{Tx} = E_{elec} * l + E_{mp} * l * d^4, \ d \geq d_0$$

and for receiving this message, respectively is:

$$E_{RX} = E_{elec} * l$$

where, E_{elec} is the energy spent to operate the transceiver circuit and E_{fs}, E_{mp} is the energy expenditure of transmitting one bit data to achieve an acceptable bit error rate and is dependent on the distance of transmission in the case of free space model and multipath fading model. If the transmission distance is less than a threshold d0, the free space model is applied; otherwise, the multipath model is used. The threshold d0 is calculated as $d_0 = (E_{fs})^{1/2} / E_{mp}$ a well defined network is to be formed considering the following parameters like the number of nodes to be deployed, node localization i.e., their positions, the node‘s initial energy levels etc. Further the sink node floods the entire network requesting their current status, thereby the sink node gains the global knowledge about the whole system. This acquirement of global knowledge avoids the sink‘s request for the unable nodes to gather the data to be sensed. This strategy avoids the energy wastage for requesting those unable nodes to face the overhead for sending the data.

The energy exhausted for reception, transmission, computation etc., during the network initialization phase is to be found and hence the residual energy level at all the nodes are determined. The sink node establishes a threshold energy level that must be sufficiently present in a sensor node so that it is capable of sensing the particular event Hence it is a must to distinguish the nodes into two types namely the nodes which have residual energy greater than the threshold are called active ones and the nodes which have residual energy lesser than the threshold are called dormant ones.

Depending upon the purpose and the atmosphere, the amount of energy spent by a sensor node is to be dealt in two scenarios. Under scenario 1, all the active nodes may spend an equal amount of energy for sensing a specific data and hence there is no need for re-clustering the system in the next cycle.

Under scenario 2, all the active nodes may spend varying amounts of energy for sensing a specific data and hence it is essential for re-clustering the system in the next cycle.

Target detection: Centralized tracking using sensor networks is possible, but has frequent drawbacks. Sending time series data through the network introduces latency and synchronization issues. It also consumes energy and network bandwidth, while potentially introducing a single point of failure (Bar-Shalom and Fortmann, 1988). Associating sensor readings to tracks suffers from combinatorial explosion when multiple sensors are used. It becomes ambiguous when sensors have overlapping ranges, disagree, or when multiple targets are present (Hall, 1992). It has eight steps:

- Candidate track information describing approaching targets is continuously received and stored in temporary priority queues.
- Local detection and parameter estimation provide inputs to the tracking algorithm.
- Detections are merged with the track that best fits the current data. Target attributes from the candidate track record are projected forward to the time of the current detection and compared with the current data.
- Confidence threshold is set so that when no candidate tracks adequately match the current detection, a new track record is created.
- Estimate future track from recent information and update the track record.
- Report track update to the user community.
- Transmit updated track record to regions along the target trajectory. Using multiple regions of varying size can provide fault tolerance. Queues containing precise regions are considered first.

Local constraint estimation is done using a position centric approach. Closest Point of Approach (CPA) data is public locally. The CPA is a vigorous marker and easily detected. It corresponds to the signal peak. Cells form dynamically within a limited space-time window. The manager node is chosen as the sensor node with the strongest signal in the space-time window. Linear regression using the trigonometry of node locations is used to estimate target position, velocity and heading. In the numerical results presented below, typically one CPA event from each of three modalities of four to five nodes was used in this calculation.

PROPOSED ALGORITHM

Our proposed algorithm is composed of two procedures: the distributed Cluster Head Selection procedure (CHS) and the Cluster Construction procedure (CC). In this section, we will first introduce the calculation of the Weight, which is a key variable for each procedure of the algorithm and then describe the detail of the algorithm itself.

Weight calculation: The calculation of node‘s Weight tries to find out a measurement for each node to identify in what degree a node is correlated with other nodes within its communication radios.

Given an undirected graph G with vertex V and edges E within the communication radius α.

α-neighbor set: For a predefined communication ┌α (i) radius of all the nodes, let be the set of vertexes within the circle of the communication radius of α if there is a transmission route from i to j the sampled data in and can be denoted as X (x1, x2, ... x3) and Y (y1, y2, ... y3), respectively. We define the distance between the measurements of i and j by the Euclidean distance between X and Y as:

$$d_{ij} = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + ... + (x_n - y_n)^2}$$

The expected value of d_{ij} is:

$$E(d_{ij}) = \frac{1}{|N(i)|} \sum_{j \in N(i)} d_{ij}$$

Then the spatial correlated weight w_i ($0 \leq w_i \leq 1$) of node i is defined as:

$$\frac{\sum_{j \in N(i)} \left| d_{ij} - \bar{d}_{ij} \right|^2}{|N(i)|^2 \; D(d_{ij})}$$

A large value of w implies small distance variation between node i and its α-neighbors i has high spatial correlation with its neighbors.

Cluster head selection: It can be performed in two stages:

- A node has very low correlation with all its α-neighbors.
- A node has very high correlation with most of it α-neighbors for the first situation a node become a dominator without any cluster member.

Weighted α-dominating set: In this the nodes with non negative weight are selected as CHs. Cluster CH selection two possible situations are considered A node decides whether it becomes a dominator or not:

- A node has very low correlation with all its α-neighbors.
- A node has very high correlation with most of its α-neighbors.

For the first situation a node become a dominator without any cluster head. This kind of dominator is called as Isolated Dominators (ID). For the second situation, a node becomes a dominator with at least one cluster member For the first situation a node become a dominator without any cluster member. This kind of dominator is called as General Dominator (GD) under this criteria the CHs selection procedure is done as follows.

In this the user will select a weight value W_{LB} (lower bound for all weight wi) $0 \leq W_{LB} \leq 1$. If the nodes in the weighted α-dominating set has weighted more than W_{LB} are selected.

Cluster construction procedure: After all the CHs are selected by the CHS procedure, each dominant has to choose a cluster to join. The Euclidean distance is applied here to construct clusters. During CC procedure, if a dominate can be dominated by several dominators, it must choose the nearest dominator (the Euclidean distance is smallest among them) to join. The details of the CC procedure are described as follows.

Each GD i broadcast an INDICATOR message embedded with its identity to all its α-neighbors. Each node has to choose a cluster to join. If j receives only one INDICATOR message i then it join the cluster i. If a node j decides to join i, it sends a JOIN message embedded with its identity to i If GD i receives a JOIN message from a node j then it sends back an ACK message to node. Then GD i is the cluster head of node j.

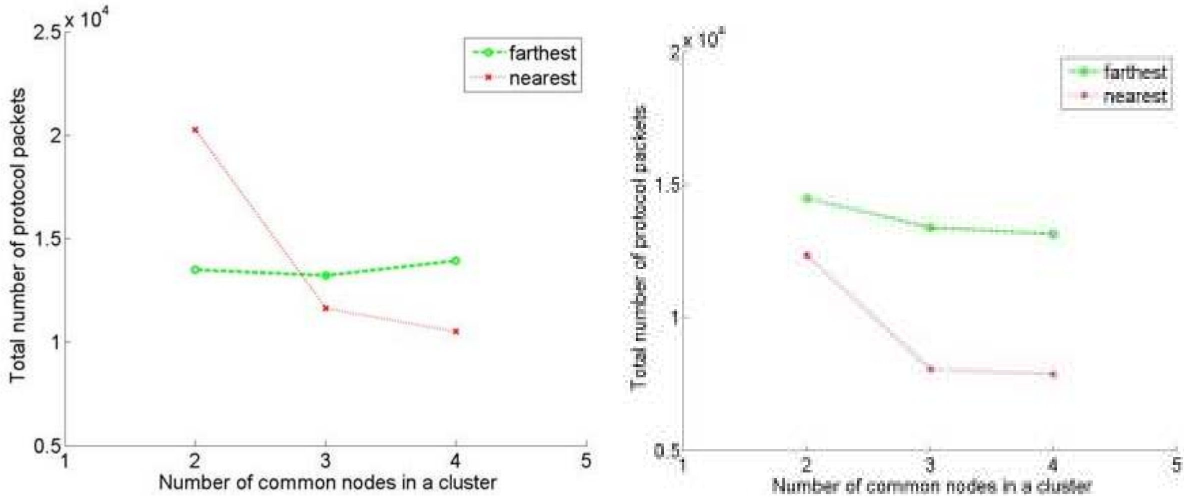
Algorithm:

- Each GD i (i £ D) broadcasts an INDICATOR message embedded with its identity to all its-neighbors j (j £ N (i)) to indicate its dominator status
- Each dominate j (j £ N (i)) chooses a cluster to join:
  - If receives only one INDICATOR message from a dominator, then it join the cluster of i (denoted as C_i)
  - If receives n ($2 \leq n \leq D$) INDICATOR messages from a set of dominators S ($S \leq D$), then j chooses a C_i (i £ S) to join if it satisfies:

    $d_{ij} = \min\{d_{ij}\ k £ S\}$
- If dominate j (j £ N (i)) decides to join C_i, it sends a JOIN message embedded with its identity to i
- If a dominator i receives a JOIN message from a Dominate j (j £ N (i)), it sends back an ACK message to j. Then i is the CH of C_i and j is a member of C_i

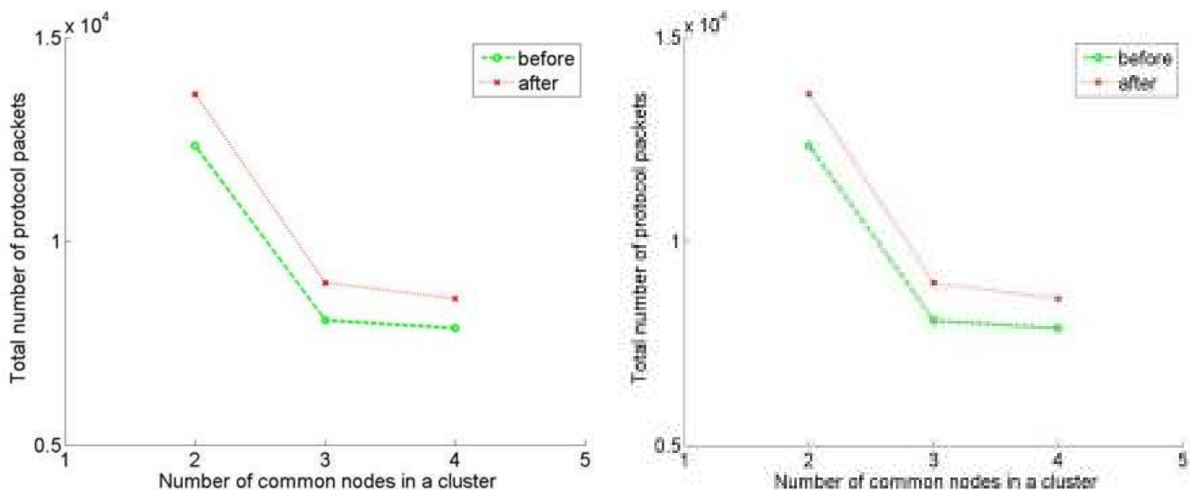
SIMULATION RESULTS

The clustering algorithm was implemented in NS-2 ver. 2.31. The total number of protocol packets generated in order to maintain the clusters for a period of 100 sec was measured. Four different deployment topologies were evaluated. The first was grid deployment in which 100 nodes were laid in 10*10 matrix and separated by 40 m. The second topology was a random deployment consisting of 150 sensors uniformly deployed over a 670*670 m field. The Protocol overhead Grid and Random deployment is



(a) (b)

Fig. 2: (a) Protocol overhead grid deployment, (b) protocol overhead in random deployment



(a) (b)

Fig. 3: (a) Protocol overhead in case of node failure in grid deployment (minimum coverage approach), (b) protocol overhead in case of node failure in random deployment (minimum coverage approach)

shown in Fig. 2. The Protocol overhead in case of node failure in Grid and Random deployment is shown in Fig. 3.

From Fig. 2 and 3 it can be seen that both the number of identical nodes two different clusters is allowed to have and the approach taken to choose the member nodes while forming a new cluster affect the protocol overhead.

The probability of conflicts between clusters is also dependant on the deployment strategy. In random and tessellation based deployment strategy it can be seen that for the case N = 2 the minimum coverage approach spends even more time in resolving conflicts whereas the protocol overhead for the maximum coverage approach is constant for all the deployment patterns.

CONCLUSION

In this study we propose a distributed clustering algorithm to track intruders. The localized nature of the algorithm ensures that reconfiguration does not significantly increase the protocol overhead. The algorithm builds a series of over-lapping clusters which allow for more than one cluster to track a region. This redundancy improves the overall system reliability. It is expected that the simulation results show better performance than the existing algorithms and also, the algorithm decreases the number of failed nodes and provides higher network lifetime and better coverage. It is believed that this algorithm can offer significant improvement on the performance and energy-efficiency of mobile sensor networks.

REFERENCES

Arora, A., P. Dutta, S. Bapat, V. Kulathumani, H. Zhang, V. Naik, V. Mittal, H. Cao, M. Demirbas, M. Gouda, Y. Choi, T. Herman, S. Kulkarni, U. Arumugam, M. Nesterenko, A. Vora and M. Miyashita, 2004. A line in the sand: A wireless sensor network for target detection, classification and tracking. Int. J. Comput. Telecom. Network., 46: 605-634.

Bar-Shalom, Y. and T.E. Fortmann, 1998. Tracking and Data Association. Academic Press, Boston.

Brooks, R. and C. Griffin, 2002. Traffic model evaluation of ad hoc target tracking algorithms. Int. J. High Perform. C., 16(3): 221-234.

Fonoage, M., M. Cardei and A. Ambrose, 2010. A QoS based routing protocol for wireless sensor networks. Proceeding of IEEE 29th International Performance Computing and Communications Conference (IPCCC, 2010), pp: 122-129.

Hall, D.L., 1992. Mathematical Techniques in Multisensor Data Fusion. Artech House, Boston.

Heinzelman, W.B., A.P. Chandrakasan and H. Balakrishnan, 2000. Energy-efficient communication protocol for wireless microsensor networks. Proceeding of the 33rd Annual Hawaii International Conference on System Science. Maui, Hawaii, pp: 1-10.

Jing, T., S. Hichem and R. Cedric, 2007. Binary variational filtering for target tracking in sensor networks. Proceeding of IEEE/SP 14th Workshop on Statistical Signal Processing, pp: 685-689.

Kaushik, C. and S. Abhrajit, 2011. An energy efficient data gathering scheme in WSN using spannig tree. Proceeding of IEMCON 2011 Organised by IEM in Collaboration with IEEE on 5th and 6th of Jan, 2011.

Kiran, M., K. Kamal and G. Nitin, 2011. Efficient cluster head selection scheme for data aggregation in wireless sensor network. Int. J. Comput. Appl., 23(9):112-119.

Lung, C.H. and C. Zhou, 2010. Using hierarchical agglomerative clustering in wireless sensor networks: An energy-efficient and flexible approach. Ad Hoc Netw., 8: 328-344.

Oh, S., P. Chen, M. Manzo and S. Sastry, 2006. Instrumenting wireless sensor networks for real-time surveillance. Proceeding of the International Conference in Robotics and Automation.

Qi, Y., Z. Yuxiang and L. Hui, 2011. An multi-hop cluster based routing protocol for wireless sensor networks. J. Converg. Inform. Technol., 6(3).

Wang, Y., 2008. Topology Control for Wireless Sensor Networks. In: Li, Y., M.T. Thai and W. Wu (Eds.), Wireless Sensor Networks and Applications, Signals and Communication Technology. Springer, US, pp: 113-147.

Watfa, M., W. Daher and H. Al Azar, 2009. A sensor network data aggregation technique. Int. J. Comput. Theor. Eng., 1(1): 19-26.