International Journal of Advanced Science and Technology
Vol. 29, No. 7s, (2020), pp. 1277-1288

Realization of Autonomous Sensor Networks for Optimal Transmission and Efficient Energy Utilization using Artificial Intelligence

[1]S. Syed Ameer Abbas, [2]M. Abirami
[1]Professor,[2]UG Student
Electronics and Communication Engineering, Mepco Schlenk Engineering College, Sivakasi.
[1]abbas_mepco@yahoo.com,[2]abirami.m05@gmail.com

Abstract

Wireless sensor networks (WSN) are most sought after for their ability to perform autonomously with least maintenance in unapproachable regions. An autonomous sensor network is one that is capable of decision making and implementing controls on its own without manual intervention. Hierarchical topology schemes are more common WSN topology that suits to compensate for the energy constrained nature of WSN. Also, to cope up with the resource constraints of WSN, intelligent data transmission is the current requirement. Hence, intelligent data gathering at node level is done by exploiting temporal correlation among sensed data. To increase the energy efficiency of the WSN, the node autonomously switches between and active modes and the duration in each mode is also computed dynamically using Q-Learning exploration - exploitation technique. This paper highlights the implementation and results of such autonomous WSN inreal-time.

Keywords:*Wireless Sensor Networks, Cluster Tree topology, Temporal correlation, Q-learning, Exploration-exploitation.*

1. Introduction

WSN are typically composed of tiny, resource constrained, highly integrated nodes distributed over a wide region that gathers information about the environment and collaborates to a centralized backend unit for further processing. WSNs are distributed in nature with intelligence added to the nodes in the form of agents that perform analysis on collected data and implement algorithms to enhance the network operation by manipulating the data gathered by the sensors.

These sensors have the ability to represent the physical parameters of the environment being sensed in electrical form. The objective of an intelligent network is to interpret such raw data alter them into information rich data. This is achieved by intelligent data gathering using Artificial Intelligence (AI) algorithms. Intelligent data gathering is achieved by ensuring non-redundant data transmission and dynamically varying the sensors' sampling duration in accordance with the environment being sensed.

Typical deployment of WSN involve random scattering of hundreds or thousands of nodes in a sparse environment where they are expected to operate in autonomous fashion similar to that of ad-hoc networks for a long duration incurring minimum maintenance.

The benefits of deploying WSN lies in its key features including ease of deployment support for wireless communication resulting in mobility and extended coverage region. The nodes of such WSN are also cost-effective. The highly integrated nature of WSNs makes them suitable for deployment even in dense inaccessible regions.

Besides, the nodes support implementation of machine learning algorithms with on chip sensing, processing, storage and transmission provisions. Provision to process data at sensing end has accelerated the WSN research to a greater extent. The features that would make WSNs as ideal choice include fault detection, efficient data transmission, energy efficiency, standardization and security improvement.

The WSN have proven to be one of the potential as well as significant technologies for their ubiquitous applications ranging from disaster detection, structural health monitoring to medical and industrial applications. VANETs (Vehicular Ad Hoc Networks) and WBAN (Wireless Body Area Networks) have gained more popularity in this decade.

They seem to be the enabling technology for the industrial revolution 4.0. Though the earlier industrial WSN were composed of both IP based and non-IP based devices, the need to develop universal networks supporting heterogeneous architecture and global connectivity has led to seek more and more IP based devices. Recent researches in WSN are aiming towards standardization of MAC, application and other network layers so as to adopt interoperability among various networks that can be globally accepted.

Other challenges in implementing intelligent WSN over a wide range include, security\ privacy issues, providing QoS and enhancing the cognitive nature to mobile networks. Mobile cognitive networks capable of operating as heterogeneous networks are the need of this hour. At national level, research approaches are headed towards finding optimal source to power the sensor nodes and development of algorithms that increase the energy efficiency of these sensor nodes.

Various principles and AI algorithms focused on enhancing the performance of WSN are discussed in [1]. Also, WSNs vary from normal networks in various aspects as they require intelligence to be introduced in solving issues like real time routing, clustering, reconfiguration, fault identification and most importantly energy efficiency.

Machine learning techniques to resolve such challenges have been discussed in [2]. In [3] and [4], various computational intelligence (CI) techniques for design and deployment of dynamic and heterogeneous WSN.

The rest of the paper is organized as follows: In section 2, the WSN implemented is described along with the architecture adopted, principles used and the algorithm for optimal data transmission. The methods of implementation of proposed system is detailed in section 3. The results obtained from implementing proposed method and its advantages are discussed in section 4. Finally, section \5 summarizes the conclusion drawn from our work.

2. Proposed Methodology

2.1 System Model

WSNs are mostly preferred for their, portability and wireless communication support. They are deployed in enormous number covering a wide region to be monitored. However, the nodes deployed depend on batteries for their operation.

In spite of the battery power being scarce, the nodes must sustain in such environments for longer lifetime. This requirement is met by adopting cluster based hierarchical routing approach. In cluster based hierarchical routing the energy consumption by each node is minimized and a resourceful node is made as cluster head (CH).

In this work the proposed WSN is implemented in cluster tree architecture as shown in Fig. 1. Single hop communication between the sensing nodes and router is not suitable for following reasons – 1) The distant node may die out before data is successfully transmitted to the sink, 2) Energy consumption would increase drastically, 3) Data redundancy will be higher at the sink.

Hierarchical topology involves two layers - a layer for sensing and another for aggregating the sensed data and forwarding to the sink by a resourceful node acting as cluster head.

Also, more than one tier in hierarchical topology supports scalability and maximum coverage of the environment in which they are deployed. Apart from achieving energy efficiency, the cluster architecture has inherent ability to prevent redundant data in data stream.

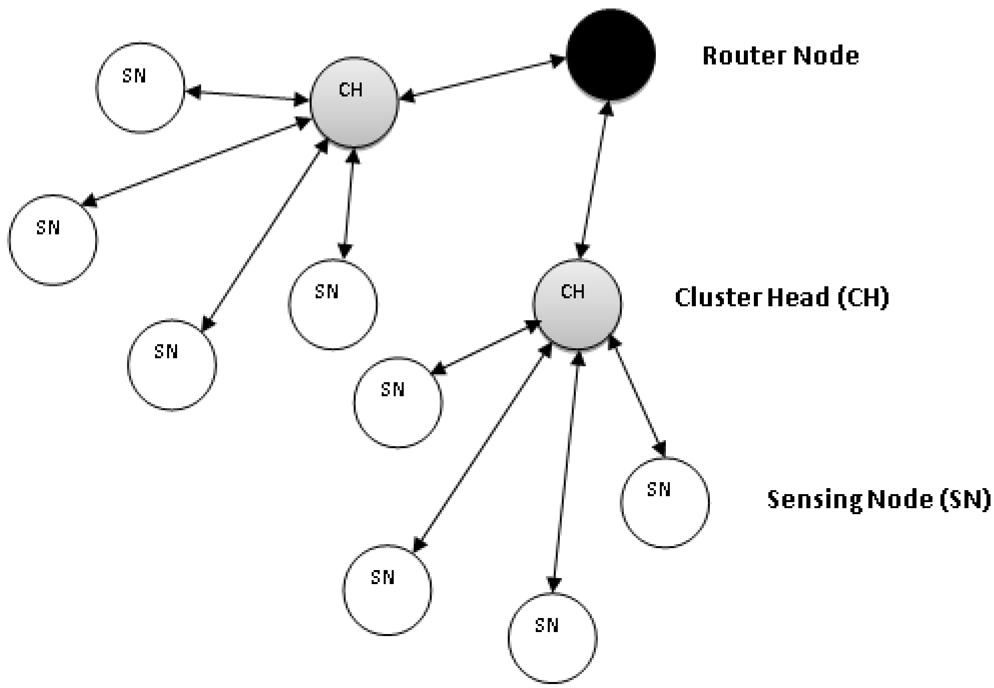

Fig 1.Cluster tree topology

Parameters to be considered while implementing such topology are – the complexity of the lightweight algorithm must increase linearly with increase in the number of nodes and load balancing at cluster heads to ensure they don't perish due to high energy consumption. Also distributed computation ensures efficient usage of available resources.

2.2Node Architecture

The nodes used are nodemcu and Wemos D1 R1. The features of both the nodes are tabulated in Table.1. Both have common elements except that Wemos D1 R1 is designed so as to substitute Arduino UNO with ESP8266 shield. The ESP8266 module can operate only at 3.3V and any high voltages may damage the IC. The version of ESP used is ESP–12E ESP8266.

It operates in ISM 2.4GHz band and supports IEEE 802.11b/g/n. The channel in which Wi-Fi transmission and reception occurs can be configured through software while the default channel used before configuration is channel 1.

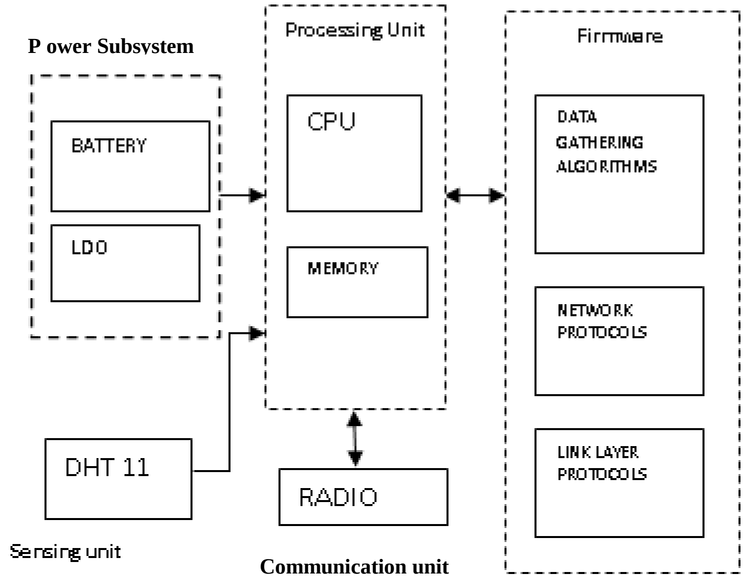

Fig 2.Wireless node architecture

The architecture of both boards includes a USB convert to serial interface. An LDO regulator that supplies the 3.3V required by the ESP module is provided as a protection unit for the ESP chip. The ESP 12E – ESP8266 performs sensing, processing and communication by itself. The functional block diagram from the datasheet of ESP module [12] of the ESP8266 is depicted in Fig 2.

TABLE I. Comparision of nodes

Nodes	*Nodemcu, Wemos D1 R1*
Flash Memory	*4MB*
GPIO Pins	*10*
Operating Frequency	*80 MHz*
V_{DD}	*3.3*

2.3 Data Redundancy Check

After the deployment of WSN, efficient data transfer objective has to be achieved. Ensuring the flow on non-redundant data alone in the network is a way of achieving optimized data transmission in WSN. The redundancy in data can be estimated by observing the frequency of the redundant data appearance. The parameter expressing such measure is correlation. The data may be correlated both temporally and spatially. The temporal correlation occurs in data samples obtained from same node while spatial correlation in data obtained from nodes placed within a common region. Spatial correlation would be evident while sensing natural phenomenon like temperature, humidity etc. in natural environments. When deployed in industrial environments the spatial correlation cannot be considered for the phenomenon measured are not natural events but parameters of machines (i.e., manmade environment). Hence for the network to adapt to every kind of environment, the data gathering in this network is done exploiting temporal correlation alone.

2.3.1 Temporal association

Redundant data denotes duplicate\repeated data without which the physical parameters sensed can be extracted. These data can be identified by computing the correlation co-efficient since the repeated data occurs after particular interval which is to be estimated. Our aim is to estimate this time interval between the occurrences of redundant data and make our node learn to sense intelligently in turn reducing energy consumption. Jaccard coefficient computes the magnitude of correlation among the available data while cosineco-efficient computes both magnitude and direction of correlation vector. The direction of correlation would be relevant while considering spatial correlation. A Comparison between Jaccard co-efficient and cosine correlation coefficient to express similarity among data is discussed in[7]. Since spatial correlation is not considered, Jaccard co-efficient is adopted for finding the extent of redundancy in the sensed data.

Consider a scenario in which M sensor nodes are scattered across the region to be sensed. A sensor node, S_i in the network senses the environment for a duration 't' and takes '2n' samples of data - $S_1, S_2, S_3, ...,$ S_{2n}. These '2n' samples can be grouped into two sets $A = S_1, S_2, S_3, ...,S_n$ and $B = S_{n+1}, S_{n+2}, S_{n+3}, ..., S_{2n}$. Now, the Jaccard correlation co-efficient can be computed as in (1).

$$J(A,B)=A\cap B/A\cup B \quad (1)$$

where, the value of J varies between (0,1). J value of 0 indicates no redundant data while entire data would be redundant if J value is 1. Based on the extent of correlation among the collected data they are classified as follows.

low correlation - $J \leq 0.3$
medium correlation - $0.3 \leq J \leq 0.6$
high correlation - $J \geq 0.6$

2.3.2 Q- Learning based data redundancy reduction

From previous sections it is evident that the number of packets transmitted can be decreased by eliminating temporally correlated data which decreases packet overhead and increases energy optimization. Data mining techniques are widely adopted to implement in WSN for the lightweight algorithms and requirement of minimum data set for processing. One such approach is the reinforcement learning or reward based learning process where the agent learns to act based on the rewards it receives for its actions. Q- learningstands for Quality learning where the agent learns about the trends of data being sensed and estimates the environmental changes based on this observation. Thistypeofregretlearning can occur with minimal dataset which makes this approach suitable for resource constrained WSN nodes. Exploration exploitation is a technique where the learner is allowed to act randomly to certain extent. This short-term randomness is introduced to discover all possible actions that could optimize the solution. Such short term sacrifices are included to achieve high accuracy in the long-run. The elements of Q-Learning are available states, possible actions, reward for actions and an environment to be sensed. The three states considered here are 1) High correlation, 2) Medium correlation and 3) Low correlation as described in previous subsection. The actions to be implemented as the output of algorithm are 1) Transmit entire data, 2) Transmit non- redundant data only and 3) Do-not transmit. The node enters sleep state during action 3 i.e., neither sensing nor communication occurs. This mode preserves maximum energy and is quintessential in extending thelifetime of networks built using ESP8266 based nodes. The primary duty of the agent is to update the Q – matrix which is initialized as all zeros during training phase. It is based on this matrix; the agent decides the action to be implemented. The exploration and exploitation is implemented using ε – greedy algorithm. This algorithm explores forever finding all possible actions during implementation. By following ε – greedy approach, the suitable action is predicted using (2).

$$\text{Current\_action} = \text{Argmax}\ (Q(S, A)) \qquad (2)$$

The above function implements the action with maximum Q value, i.e., the action with maximum reward. The learner updates Q-matrix using Bellman's equation in(2). The Q value after implementation of current_action is updated using Bellman's equation (3).

$$Q(S, A) = Q(S, A)+\alpha(R+\gamma[max[Q(newstate:)]])-Q(S, A)) \qquad (3)$$

where,

α – learning rate s (0,1)
γ – discount factor s (0,1)
R – reward corresponding to the action.

Algorithm: Q Learning based data redundancy reduction

Inputs: Jaccard co-efficient (J), Reward matrix

Outputs: Current action, Sleep duration

1. Estimate current state based on J value.
2. If (random(0,1) $\geq$ 1- ε)
   Current action = estimated action
3. Else
   Current action = random (states)
4. Update Q matrix
5. Diff = Q_{new} - Q_{old}
6. If (Reward $\geq$ 0)
   $\Theta_{new} = \Theta_{old} + W*Diff$
7. Else
   $\Theta_{new} = \Theta_{old} - W * Diff$
8. End If

In the algorithm, Θ denotes duration of the action and w is weight factor. This ensures the redundant data is not transmitted. To ensure that the node does not miss out any event that may occur while the node is in sleep mode, the duration for which the node remains in particular state is also dynamically calculated based on Q-value variation.

3. Implementation

There are numerous wireless communication protocols that support WSN implementation like Bluetooth, ZigBee, Wi-Fi, RF etc., Among these, the WSN proposed is realized in real time using the wireless protocol IEEE 802.11 i.e., Wi-Fi for its larger bandwidth, increased coverage capability and high data rate. The network proposed has three level hierarchies including sensing nodes at level 1, cluster heads or coordinator nodes at level 2 and routers at level 3. The number of levels can be extended to increase scalability by making routers as second level coordinators and routers at higher levels. The nodes used to realize IEEE 802.11 based WSN are ESP8266 supported nodes - 1. nodemcu 2. Wemos D1 R1.ESP8266 based nodes are cost effective, high integrated wireless SOCs, designed for power and space constrained environments. In this WSN implemented using three level hierarchies Wemos D1 R1 was chosen as the cluster head and router while nodemcu were chosen as sensing nodes.

Each ESP8266 node can connect to four other nodes thus forming a cluster of four sensing nodes with one cluster head. When nodes are deployed covering large area, the sensing nodes perform Wi-Fi scan to identify the coordinator networks available and connect to the coordinators having higher RSSI value thus forming clusters. As each node has the capability to connect only to four other nodes, the fifth sensing node would try to connect to some other coordinator if one coordinator is fully connected. The role of sensing nodes is to gather information about the environment being sensed and relay the data sensed to the coordinators. The coordinators in turn process the data for redundancy checks and forward the resultant data to the router. Thus, in this architecture data forwarding is done hierarchically from bottom most nodes towards the top node. Painlessmesh [6] library supporting the cluster formation has been used for implementation. Using painlessmesh library the nodes in the cluster are arranged in a mesh like architecture with only one route between any two nodes. Also, the communication established within such clusters is not IP based but chip ID based which in turn reduces the delay in data transmission. The data is sent as JSON (JavaScript Object Notation) format with source and sink indicated by their chip IDs.

The nodes connected in same cluster are listed as the elements of a tree constructed to depict the cluster connectivity and every node in the cluster is aware of all the members of that cluster. The cluster head here acts as the root for the cluster architecture. The sensing nodes are configured as station points while the router and cluster heads act as both station and access points. Arduino IDE was used to implement the algorithm. The temperature sensor used is DHT11 temperature and humidity sensor. The algorithm is implemented at level-1 (sensing nodes) in the hierarchy to reduce the transmission of redundant data to the cluster heads. Further, the node dynamically estimates the duration for which it has to implement the action predicted using the algorithm. The implementation of this algorithm and three level tree formation is displayed in Fig 3.

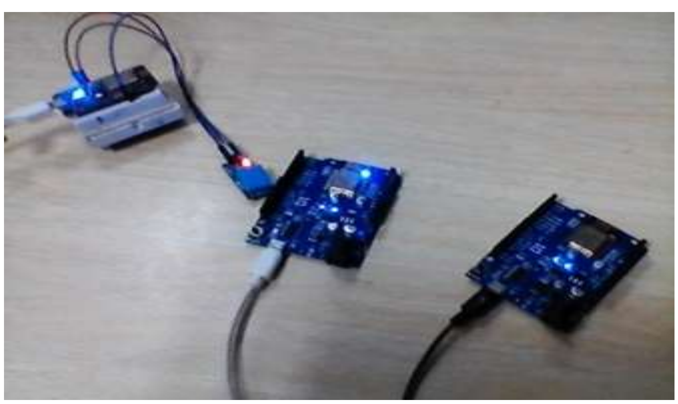

Fig 3.Nodes at three levels of cluster tree

The reward matrix is initialized according to the application requirement. Thetemporal associations were computed after receiving '2n' samples and then the next action is decided by the Q-learner for the duration computed. The Q – learner operation occurs in two phases – Training and testing. In training phase, the agent updates the Q – matrix based on actions implemented and reward received. In testing phase, real time data are considered and actions are implemented based on equation (2). The algorithm initially chooses random states during training phase building the Q-matrix and updates test phase with real time data acquired from the sensors.

When implemented in three level cluster tree topology, the results obtained at various levels are displayed in Fig. 4-6. The details displayed are taken from Arduino serial monitor. Local Wi-Fi network is used to upload the data to the cloud for future analysis.

```
16:11:53.188 -> setLogLevel: ERROR | CONNECTION | DEBUG |
16:11:54.118 -> CONNECTION: stationScan(): Mesh1
16:11:56.338 -> CONNECTION: scanComplete(): Scan finished
16:11:56.338 -> CONNECTION: scanComplete():-- > Cleared old APs.
16:11:56.338 -> CONNECTION: scanComplete(): num = 6
16:11:56.338 -> CONNECTION:         found : Mesh1, -16dBm
16:11:56.338 -> CONNECTION:         Found 1 nodes
16:11:56.338 -> CONNECTION: connectToAP(): Best AP is 3955992954<---
16:11:56.338 -> CONNECTION: connectToAP(): Trying to connect, scan rate set to 4*normal
16:11:58.110 -> CONNECTION: Event: Station Mode Disconnected
16:11:58.110 -> CONNECTION: eraseClosedConnections():
16:11:58.110 -> CONNECTION: connectToAP(): No unknown nodes found scan rate set to normal
16:11:59.657 -> CONNECTION: Event: Station Mode Connected
```

Fig 4.Sensing node output

Fig.4 shows the results of scanning done by the sensing node. Scan is initiated and it finds the coordinator network named “Mesh1” and connects as station point to the particular network. Fig. 5 the messages from coordinator node. The data has been received from sensing node and displayed. It is then forwarded to next level i.e., router node.

```
-> Changed connections
-> Num nodes: 1
-> Connection list: 840382810
-> Co-ordinator: Received from 840382810 msg={"sensor":"Temperature","value":14.4}
-> Co-ordinator: Received from 840382810 msg={"sensor":"Temperature","value":28.9}
```

Fig 5.Coordinator node output

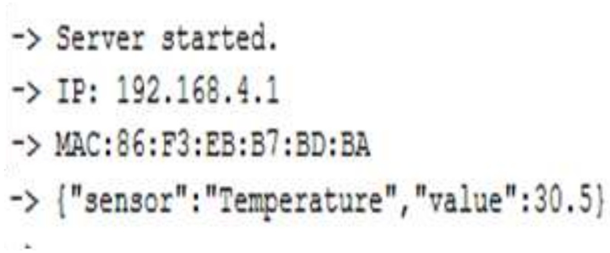

Fig 6.Router node output

The router acts as server receiving the data relayed by the sensor node and updates the data to the cloud. Thee cloud host utilized is ThingSpeak. It supports MATLAB program implementation on data received.

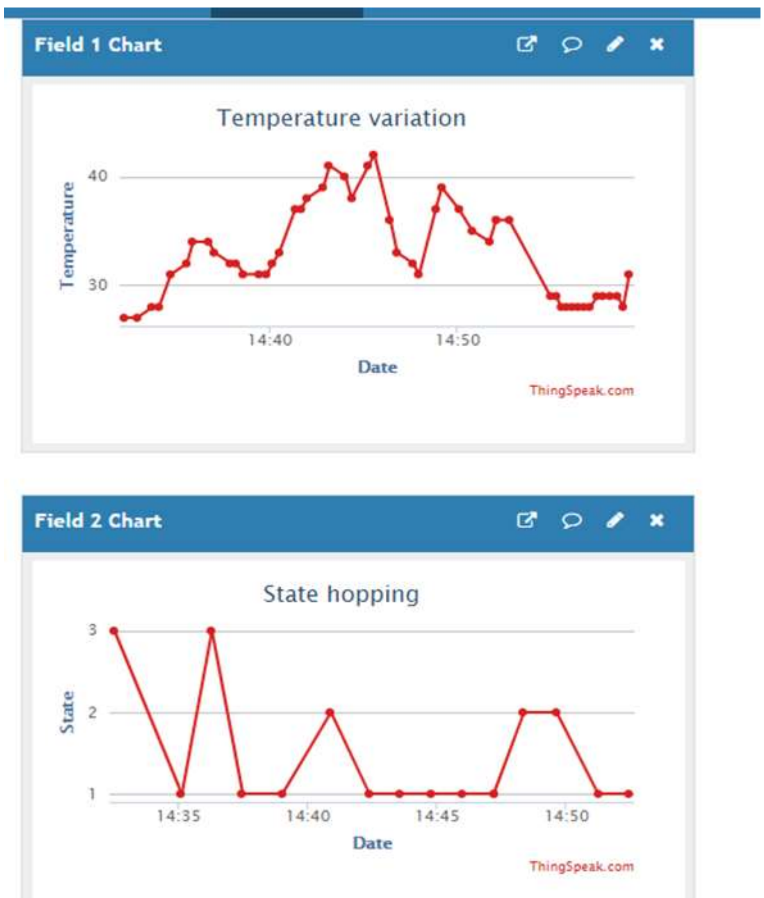

Fig 7.Router node output

Fig. 7.Depicts the data being plotted at the cloud. Both temperature readings and states predicted by Q-learner are plotted.

4. Results

MATLAB is used to analyze the results of this algorithm. The dataset used for the analysis was downloaded from online dataworld repository available at" https://data.world/fangfufu/nanning-2016-temp- pressure". The dataset was prepared by measuring the temperature and pressure readings taken in a residential building using BME280 sensor. About 320 samples were selected for this analysis. The plot of temperature variation is in Fig. 8. The corresponding states according to the temperature correlation as predicted by the Q-learner is plotted in Fig. 9. The Q-learner ensures the node is not struck in one particular state forever. This feature is ensured by the randomness factor included in the form of exploration-exploitation technique.

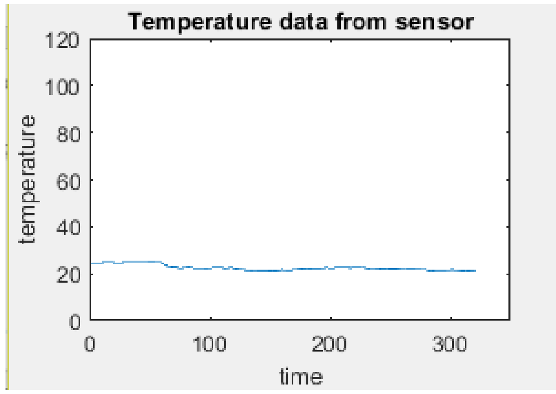

Fig 8.Temperature variation

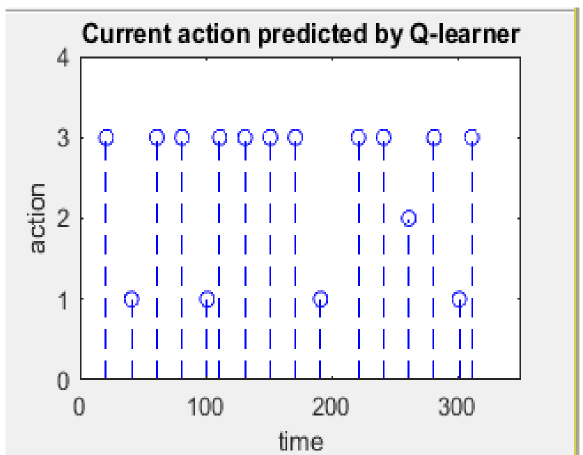

Fig 9.Current action predicted by Q-learner

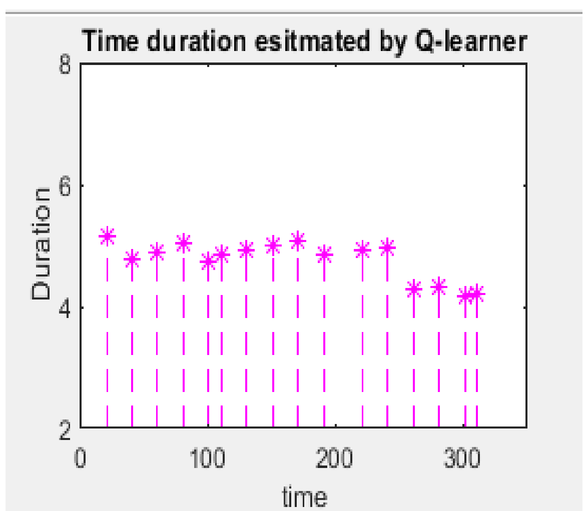

Fig 10.Sleep duration estimation

After predicting the current action to be implemented, the duration for which the node remains in that particular state is again calculated dynamically and plotted in Fig. 10. This time is proportional to change in Q-value. The duration is incremented for positive rewards received and decremented for negative rewards. Negative reward indicates the action is not beneficial and hence the node has to switch from that particular state. Accordingly, the sleep time is reduced. The dynamic estimation of sleep duration ensures the node does not miss to detect the occurrence of any event at the same time the node's energy is also preserved.

5. Conclusion

The sensor nodes deployed to monitor an environment is implemented in cluster tree topology to exploit the advantages of energy efficiency, scalability and wide coverage. Further, optimized data transfer is achieved by using temporal correlation among the sensor data to filter redundant data and forward non-redundant data alone. This optimization is essential to increase the energy efficiency of the sensor nodes which rely upon the attached battery for their operation. In this paper the nodes apply reinforcement learning i.e., Q-learning based on exploration exploitation approach to schedule the node's sleep - wake cycle. This ensures the node enters sleep mode after certain period of operation increasing the network lifetime. The duration for which the node remains in the chosen state is again determined by the Q-learner. The sleep duration is increased after receiving positive reward and decreased after receiving negative reward for the action chosen to be implemented. The duration is directly proportional to the extent of variation in Q-value.

The network implemented in this paper considers only homogeneous nodes. More than one sensor can be connected to the nodes to sense numerous parameters and gather them based on temporal correlation. Heterogeneous nodes can also be included in this network and the same algorithm can be implemented in those nodes too. However, the parameters to be considered while implementing heterogeneous nodes in a network include ensuring Quality of Service (QoS) to the network using temporal correlation.

References

1. Alcides Montoya, Diana Carolina Restrepo and Demetrio Arturo Ovalle, "ArtificialIntelligenceforWirelessSensorNetworksEnhancement," Smart wireless sensor networks,National University of Colombia -Campus Medellin Colombia.
2. Mohammad Abu Alsheikh,Shaowei Lin, DusitNiyato and Hwee-Pink Tan, "Machine Learning in Wireless Sensor Networks:Algorithms, Strategies, and Applications," in arXiv:1405.4463 , VOL. 2, 19 March 2015.
3. BasmaSolaiman and AlaaSheta,"Computational Intelligence for WirelessSensorNetworks:ApplicationsandClusteringAlgorithms,"in International Journal of Computer Applications (0975 - 8887), VOL.73,NO. 15, July2013.
4. Raghavendra V Kulkarni, Anna Forst and Ganesh Kumar Venayagamoorthy, "Computational Intelligence in Wireless Sensor Networks: A Survey,"inIEEE Communications Surveys and Tutorials, VOL.13,NO.1,27 May2010.
5. Richard S. Sutton and Andrew G.Barto, "Temporal-Difference Learning," in Reinforcement Learning: An Introduction,Cambridge, MA, USA: MIT Press,2017.
6. Yoppy, R Harry Arjadi, EndahSetyaningsih, PriyoWibowo and M I Sudrajat,"Journal of Physics: Conference Series," Volume 1230, 2nd International conference on Mechanical, Electroncs, Computer and Industrial Technology, Medan, North Sumatera, Indonesia, (2018) December 12-14.
7. BakhtiarQutub Ali,NikiPissinou and Kia Makki,"Identification and Validation of Spatio Temporal Associations in Wireless Sensor Networks",IEEE Third InternationalConferenceonSensorTechnologiesandApplications2009.

8. Hassan Harb, AbdallahMakhoul, Samar Tawbi and Raphael Couturier,"Comparison of Different Data Aggregation Techniques in Distributed Sensor Networks," in IEEE Access - Special Sectiononheterogeneouscrowdsourceddataanalytics,March10,2017.
9. Linghe Kong, Daqiang Zhang, Zongjian He, Qiao Xiang, Jiafu Wan andMeixia Tao,"Embracing Big Data with Compressive Sensing: A Green Approach in Industrial Wireless Networks," IEEE CommunicationsMagazine,VOL.54,NO.10,OCTOBER2016.
10. David Silver, "UCL course on Reinforcement Learning,", UCL London Global University,Available: http://www.cs.ucl.ac.uk/ staff/d.silver/web/Teaching.html.
11. Bilal Jan, 1 HaleemFarman HumaJaved , BartolomeoMontrucchio , Murad Khan and Shaukat Ali, "Energy Efficient Hierarchical Clustering Approaches in Wireless Sensor Networks: A Survey", Hindawi Wireless Communication and Mobile Computing. Vol. 2017, 29 October 2017.
12. K. Yuen, B. Liang, and B. Li, "A distributed framework for correlated datagathering in sensor networks," *IEEE Trans. Veh. Technol.*, vol. 57, no. 1,pp. 578–593, Jan. 2008.
13. He, J. Chen, D. K. Yau, and Y. Sun, "Cross-layer optimization of correlated data gathering inwireless sensor networks," *IEEE Trans.MobileComput.*, vol. 11, no. 11, pp. 1678–1691, Nov. 2012.
14. G. Ferrari, M. Martalo, and A. Abrardo, "Information fusion in wireless sensor networks with source correlation," *Inf. Fusion*, vol. 15, pp. 80–89,2014.
15. F. Gielow, G. Jakllari, M. Nogueira, and A. Santos, "Data similarity aware dynamic node clustering in wireless sensor networks," *Ad Hoc Networks*, vol. 24, pp. 29–45, 2015.
16. J. Niu and Z. Deng, "Distributed self-learning scheduling approach for wireless sensor network," *Ad Hoc Networks*, vol. 11, pp. 1276–1286, 2013.
17. M. Mihaylov,Y. L. Borgne,K. Tuyls, andA.Nowe, "Reinforcement learning for self-organizing wake-up scheduling in wireless sensor networks," in *Proc. 3rd Int. Conf. Agents Artif. Intell.*, Rome, Italy, 2011, pp. 382–396.
18. G. Anastasi, M. Conti, and M. D. Francesco, "Extending the lifetime of wireless sensor networks through adaptive sleep," *IEEE Trans. Ind. Informat.*, vol. 5, no. 3, pp. 351–365, Aug. 2009.
19. H. Byun and J. Yu, "Adaptive duty cycle control with queue managementin wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 12,no. 6, pp. 1214–1225, Jun. 2013.
20. H. S. AbdelSalam and S. Olariu, "Toward adaptive sleep schedules for balancing energy consumption in wireless sensor networks," *IEEE Trans. Comput.*, vol. 61, no. 10, pp. 1443–1458, Oct. 2012.
21. I. F. Akyildiz, S. Weilian, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communication Magazine*, vol. 12, pp. 102–114, August 2002.
22. M. C. Vuran, O¨zgr B. Akan, and I. F. Akyildiz, "Spatio-temporal correlation: theory and applications for wireless sensor networks," *Computer Networks*, vol. 45, no. 3, pp. 245–259, 2004.
23. M.C. Vuran, I.F. Akyildiz, Spatial correlation-based collaborative medium access control in wireless sensornetworks, submitted, December 2003.
24. W. Ye, J. Heidemann, D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks", in " Proceedings of the INFOCOM " , 2002 vol. 3, pp. 1567–1576