International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 21 (2017) pp.11409-11416

Cooperative Caching in IEEE802.15.4 based WSNs

Piyush Charan[1], Tahsin Usmani[2], Rajeev Paulus[3] and Syed Hasan Saeed[4]

[1]Assistant Professor, Department of Electronics and Communication Engineering, Integral University, Lucknow, Uttar Pradesh, India.
[1]Orcid Id: 0000-0002-8949-4056

[2,4] Professor, Department of Electronics and Communication Engineering, Integral University, Lucknow, Uttar Pradesh, India.
[2,4]Orcid Id: 0000-0002-3296-3069, 0000-0002-8194-5203

[3]Assistant Professor, Department of Electronics and Communication Engineering, Sam Higginbottom University of Agriculture, Technology and Sciences, Allahabad, Uttar Pradesh, India.
[3]Orcid Id: 0000-0003-4441-3619

Abstract:

Wireless sensor networks (WSNs) based on the IEEE 802.15.4 MAC and PHY standards are a recent trend in the market. Due to its low power consumption characteristics and low data rate, it gained a lot of attention. However, for large networks, spreading of overheads over the entire network is still a problem. The energy consumption can be reduced by minimizing overhead by combining new collaborative caching schemes and providing data with minimal latency, thereby reducing energy consumption. This article explores the possibility of improving energy efficiency by consolidating caching strategies.

Keywords: Cooperative Caching, IEEE 802.15.4, ZigBee, WSNs, Energy Efficiency, AODV

INTRODUCTION

With the wireless personal area network (WPAN) received a huge response, the wireless network community has been looking for new ways to enable wireless connectivity to extend to newer dimensions and explore a wide range of applications.

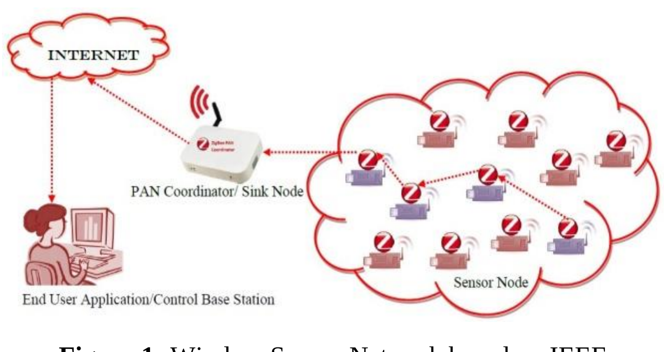

Figure 1: Wireless Sensor Network based on IEEE 802.15.4/ZigBee Devices

Optimization of total link costs can lead to efficient sensor network design [1], where power and quality of service (QoS) are two main attributes that should be controlled, and researchers must focus on energy consumption and high routing efficiency.

ZigBee is based on the IEEE 802.15.4 standard is a low-rate wireless personal area network (LR-WPAN) device and the most important source of energy consumption in such devices is the transmission and reception, monitoring, idle listening, collisions and overheads of data packets brought together at the sink node (multiple packets due to unicast or multicast).

According to Narottam Chand [2], supplying continuous data to the receiver in the case of uninterrupted communication is a challenge in designing a large-scale ubiquitous sensor network. Caching is one of the most effective ways to improve performance, and in recent years the cache size has witnessed significant growth in various computing systems.

COOPERATIVE DATA CACHING

In a wireless sensor network, cooperative caching helps wireless motes coordinate and share cached data, reduce communication / link costs, and take advantage of the overall cache of cooperative sensors. IEEE 802.15.4 based sensor nodes are accompanied by flash memory for frequently accessed data items. The data that is cached in the flash satisfies the node's own request while satisfying the data request passed from the other nodes. When data is lost, the node first searches for data in its zone / cluster and then forwards the request to the next node on the path towards the data source. The process of cache admission control (CAC) depends on the distance criteria of the nodes from the sink node and gives high priority to nodes near the sink. Caching plays an important role in reducing the number of communications from the sensor node (SN) to the number of received communications by caching the useful or frequently accessed data.

Caching is a technology that provides faster data access in any computing system. With the discovery of the cache, the accessibility of the data has increased because it stores the data required in the future and can be retrieved simultaneously with the access request. The cache [8] also has an impact in the wireless sensor network.

Providing constant data to the sink in a continuous correspondence is the primary task of implementing a large-scale sensor system. Information routing [3,4], data compression [7] and intra-network data aggregation [5,6] have been studied in recent years in WSN. Through the incorporation of cooperative caching in wireless sensor networks, we have tried to solve the problem of efficient data dissemination to a certain extent. The best implementation of the cache can reduce network traffic and improve data availability.

RELATED WORK

Many cache-related scenarios have been proposed in the past. Jin Bao Li et.al [9] proposed a multi-sink sensor network cache scheme. The sensor network forms a set of network trees for a particular receiver node/sink. The combination of these trees forms a common subtree and selects the root of the common subtree as the data cache node to reduce the communication cost.

J. Xu et al. [10] proposed a latency cache scheme that expected that the same cluster of data becomes available within the threshold and then aggregates it with packets from the lower cluster and sends it later to the sink, thereby minimizing the number packets traversing in the network. Md. A. Rahman et al. [11] proposed an efficient caching mechanism that, by coordinating the data between base stations and sensors develops expectations of data changes and data loss.

T.P. Sharma et al. [12] proposed a cooperative cache scheme that used the cooperation between the various SNs in the defined region. In addition to its own local storage, a node uses memory of nodes from other clusters around it to form a larger cache called the Cumulative Cache. In this case, a token-based cache admission control (CAC) scheme is designed in which the node carrying the token is admissible to replace and cache the data item. The main drawback of the proposed model is the generation of more packet overheads so as to maintain the token and rotate it as and when required. Later, N. Dimokas et al [13,14] identified various targets that need to be optimized, such as consumption of energy, access delay, and the copies of data to be stored in different locations. The drawback of the scheme is that the node importance (NI) index takes into account the neighborhood of a particular node. Therefore, finding the cost of NI for all nodes consumes energy, thereby reducing the life of the sensor network.

P. Kumar and N. Chauhan et al. [15] proposed a novel proactive method to cache the data in MANET of mobile node. In this method, all nodes leaving a particular zone will broadcast the "LEAVE" message to all neighboring nodes. The data to be cached is determined by the zone administrator based on the cache information table (CIT) of the leaving node. They also explained that good cooperative cache management technology for MANETs should address the following issues:

1. A Cache Discovery Algorithm that efficiently retrieves and passes the requested data item from a neighboring node.

2. A Cache Admission Control Algorithm is required that decides which data is cached to be used in future. In cooperative caching, this decision not only depends on the needs of caching node, but also considers the needs of other nodes.

3. When the cache space is not enough to cache the new cache data, there should be a cache replacement algorithm to replace the cached data items.

4. A cache consistency algorithm to ensure that the cached data items are updated based on their Time-to-Live (TTL) values.

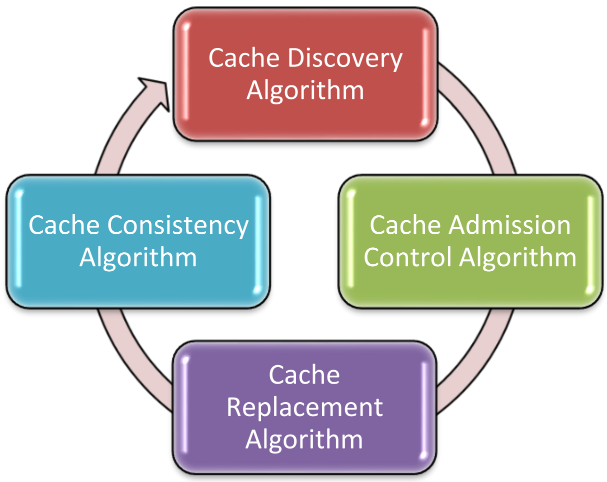

Figure 2: Requisites for Cooperative Cache Management

SYSTEM ENVIRONMENT

We consider a Wireless Sensor Network consisting of Sensor Nodes (SNs) that are based on the IEEE 802.15.4/ZigBee devices and are capable of sensing physical data while interacting with the environment. Based on the type of transmission these networks are classified as operating in either i). Push Mode, or ii). Pull Mode. In push mode, the sensor nodes are sensing data all the time and once the data is aggregated it is pushed towards the Sink node or the PAN Coordinator which later is responsible for taking Control decisions for the system. In such a system, each sensor node itself is a Source of information. Whereas, in pull mode the data is sensed only when the Requester/Sink node expresses an interest for a particular data at an instant. Thus we see that Caching may not be much helpful in Push Mode because all the sensors are sensing data and all the neighboring nodes are sources of information. The adhoc on demand distance vector (AODV) is one such on demand routing protocol that makes all the nodes to work in Pull Mode in which all routes are discovered only when needed and are maintained as long as they are being used [16]. A data request initiated by a sink is forwarded hop-by-hop along the routing path along with a beacon signal until it reaches the source and then the source sends back the requested data. The beacon signal makes all the radios in its routing path to wake-up and acknowledge the request made by the Sink/Coordinator. Caching in such a system improves the overall system performance by reducing the number of packet overheads in requesting information from neighboring nodes and the data availability is increased.

WSN consists of sensor nodes which comprises of limited cache storage e.g. for multimedia data, then cooperative caching may also be useful for sharing the cached data among the neighboring nodes. Since WSNs comprises of a group of motes communicating via omni-directional antennas within the same transmission range. Thus, a WSN topology is typically represented by a graph G= (S, E) where S is the set of sensor nodes SN_1, SN_2, …., and $E \subseteq S \times S$ is the set of links between nodes. The existence of a link (SN_i, SN_j) ∈ E also means (SN_j, SN_i) ∈ E which means that if a link exist between node 'i' and 'j' then the same exists between nodes 'j' and 'i'. Also nodes SN_i and SN_j are within the transmission range of each other, and are called one-hop neighbors. The set of one-hop neighbors of a node SN_i is denoted by SN_i^1 and forms a zone.

Following assumptions have also been made in this System Environment:

- The communication links between Sensor Nodes are bidirectional, and the sensors communicate via multi-hop transmissions.
- WSN is uniform, that is, all sensor nodes have the same computational and communication capabilities.
- Each sensor node uses some positioning methods to know its geographic coordinates (x, y).
- Since we assume that there is limited cache storage and once the cache space of a node is full then the node will select some data items to remove from the cache, when it has to cache new data.
- Each node in the zone will maintain a Table of Cache Information denoted as TCI. This TCI will contain *n* elements where *n* is the number of the data items.

There will be four entries related to each node which will be as follows:

The first entry is *d.avail* that shows whether *d* is locally cached at node *X*. This is binary type value and is *TRUE* if data is locally available.

The second entry is *d.neighnode* and shows which neighbor node has cached *d*.

The third entry is *d.acount* which is maintained in order to count how many times *d* is cached by neighbor nodes of node *X* after *d* is cached by node *X*.

The final entry is *d.ttl* shows the *TTL* (time-to-live) value that after how much time *d* is expired. This value is assigned by the data server. The entries of TCI are summarized in table 1.

Table 1: Entries of the Table of Cache information (TCI)

Sl. No.	Entry Name	Meaning	Initial Value
1	*d.avail*	shows whether *d* is locally cached at the requesting node	FALSE
2	*d.neighnode*	shows which neighbor node has cached *d*	null
3	*d.acount*	shows how many times *d* is cached by neighbors node of node *A* after *d* is cached by node *A*	zero
4	d.ttl	shows after how much time *d* is expired.	assigned by the data server

Before forwarding the request for data to an intermediate one hop neighbor in its route path, each node will check if it contains the required data in its local cache and if it has the data then it will send directly and stop the forwarding else it will forward the request to the one hop neighbor that it knows has cached the data item.

CACHING IN COOPERATIVE ZONES (CCZ)

Caching in Cooperative Zones (CCZ) is an adaptive technique for data retrieval in WSNs based on IEEE802.15.4.

In CCZ caching technique, it is beneficial for the SN to share cached data with its one hop neighbors located in the zone. SNs that are part of a zone of a given node forms a cooperative cache system for that particular node since the link cost or the cost of communication with them is low both in terms of energy consumption and data exchange. Fig. 3

shows the behavior of CCZ caching strategy for a data request. For each request, one of the following four Scenarios holds:

Scenario 1: A Local hit is said to have occurred when a copy of the requested data item is found in the local cache of the sensor node. If the data item is valid, it is used to serve the query and no further cooperation is required.

Scenario 2: The Zone hit is encountered when the requested data item is found in the cache of one or more single hop neighbors of the requester. Messages exchanged within the domain of the requested sensor node (local zone) during the cache discovery.

Scenario 3: It is said that a Remote Hit occurs when data belonging to a region other than the local zone of the requester is found along the route path towards the data source.

Scenario 4: A Global hit is said to have encountered when none of the nodes could have helped then the requested datum is served by the data center or the Source Node.

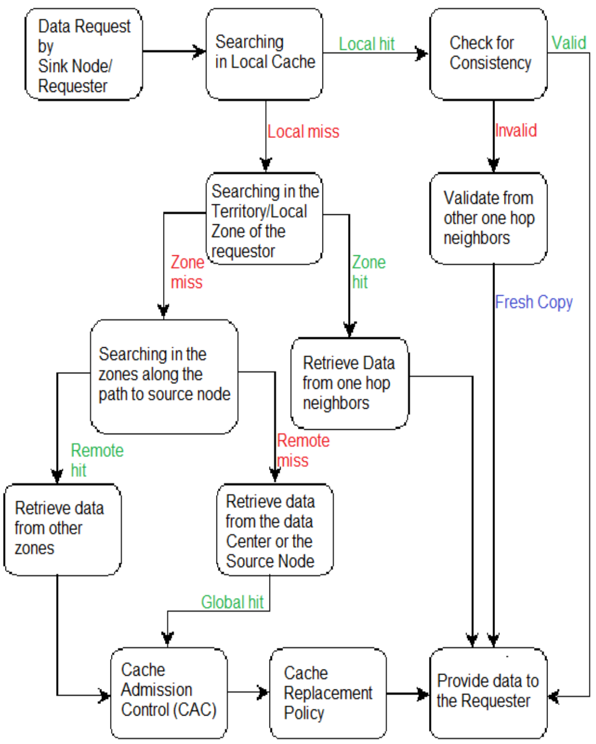

Figure 3: A typical data request being administered by the CCZ Protocol

Cache Discovery Process

Caching in Cooperative Zones (CCZ) uses an algorithm for Cache Discovery where in the node which has cached the requested datum is located. Once a data request is initiated by a sink node or PAN coordinator, it first searches the data in its own cache. If the data is found in the local cache then its *d.avail* is found to be'1'or TRUE. Then the data is used by the Requester for further processing. Such type of data retrieval is called a Local hit. If a local cache miss is encountered then the request is forwarded to one hop neighbor of the Requester or the local zone of the requester. In case the zone cache is also missed then the request for the required data is forwarded towards the Source Node and if the data is found while routing in any cluster as indicated by the CCZ algorithm, it will be returned to the requester, such type of retrieval is called a remote hit. Otherwise the data is fetched and returned by the source node, which is called a Global hit.

Cache Admission Control

The Cache Admission Control is responsible for making a decision of whether to cache a particular data item at a Sensor Node or not. An efficient cache management ensures caching a data item near to the Sink Node/PAN Coordinator or the Requester so that it is easily accessible. So, Caching also depends on the Distance Function of a node from the Requester, which is calculated as:

$$\delta_f = 1 - \frac{D_i}{S} \qquad \ldots\ldots\ldots(1)$$

Where, $\delta_f \rightarrow$ represents the Distance Function of a node from the Requester/Sink

$D_i \rightarrow$ is the distance between node N_i from the Requester/Sink

$S \rightarrow$ is the Network Size or total no. of Nodes in the Network

A particular data item cached is based on the Distance Function, δ_f. If the $\delta_f > Threshold$, then the data should be cached in that particular node. If the $\delta_f < Threshold$, then the data should not be cached. To avoid multiple cached data items a time window concept is incorporated which will not cache the data if the cluster has cached value in the time window t_w. When a data passing through a node is cached in that particular node then it sets cache_tag=1 in the data packet and the field *d.neighnode* will change its value to the Node Number of the cached node.

Cache Consistency

The Cache Consistency ensures that sensor nodes are served with valid and fresh data and no expired data is served for data requests. Two commonly used cache consistency models are the weak consistency and the strong consistency models. Since WSNs are characterized by multihop environment, limited bandwidth and energy constraints so the weak consistency model is more suitable. The CCZ caching approach uses a simple weak consistency model base on the

Time-to-Live (TTL), in which a SN a cache copy up-to-date if its TTL has not expired. Once the TTL of a datum expires, the node considers for the replacement of the cached data. When a fresh copy of the same data passes by a Sensor Node, the SN updates its TTL and a cached data item is refreshed.

Cache Replacement Policy

The CCZ is a cooperative caching technique that enables data replacement in the cache once the TTL value of the data expires. Hence we reach to a conclusion that the Cache Replacement Factor (CRF_i) for a data item d_i at a sensor node N_i depends on the following three factors:

Popularity $\left(P_{d_i}\right)$: The probability of a data item, d_i being accessed by a host represents the Popularity of that data, which is computed as:

$$P_{d_i} = \frac{d.acount_i}{\sum_{k=1}^{N} d.acount_k} \quad \dots\dots(2)$$

Where, $d.acount_i \rightarrow$ is the mean access count of a data item d_i being accessed by a host node.

Distance(D_i): is the distance between the caching node N_i and the Requester/Sink Node and is the measure of the number of hops between Requester Node and the caching node.

Consistency (TTL_i): Sometimes the cached data is not accessed at all by the requester in such a case to maintain the consistency of the system the Time-to-Live value of a data item is considered. When the TTL_ivalue of a data is expired the data item will be removed from the Cache.

Based on these attributes the Cache Replacement Factor (CRF_i) for a data item d_i is computed as:

$$(CRF_i) = P_{d_i} \cdot D_i \cdot TTL_i \quad \dots\dots(3)$$

The more the CRF_i value is the more is the utility value of the data item d_i to be kept in the cache. The cache will get up-to-date with fresh data once the CRF_i value of data d_i is found the least. The CCZ algorithm uses a heuristic approach that removes a cached data item, d_i having least CRF_i until free cache space is sufficient enough to accommodate fresh data that is ready to be cached.

SIMULATION AND ANALYSIS

Simulation Model

The Simulation of the proposed CCZ algorithm is carried out on NS-2(version 2.32). The routing protocol used is AODV[1,16, 17] to route the data traffic in the Wireless Sensor Network based IEEE802.15.4 MAC and PHY protocol and the free space propagation model as the radio propagation model. The number of nodes is 16, deployed in a sensor field region of 100*100 m^2. The wireless bandwidth is 250kbps that represents the maximum amount or bits of data that can be transferred in a time period, normally in one second.

We have simulated AODV with CCZ caching technique in two analytical models where in we consider the nodes to be connected in STAR connection and in the other network model; the nodes are connected in a Grid.

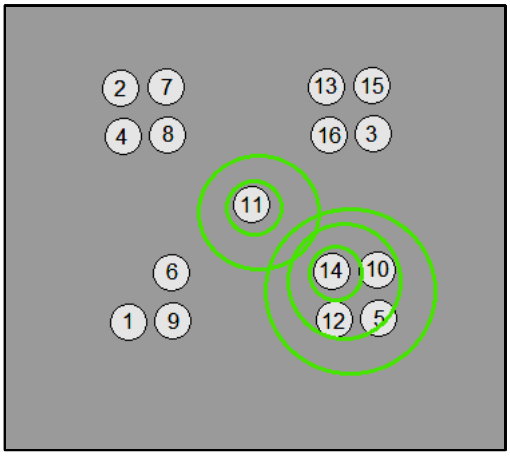

Figure 4: Star based Network Model

In Star (Cluster based) network, 16 sensor nodes based on IEEE 802.15.4 PHY standard are considered which are all Full Function devices (FFDs) and are accompanied with a fixed amount of cache memory to store the sensed data as shown in figure 4.

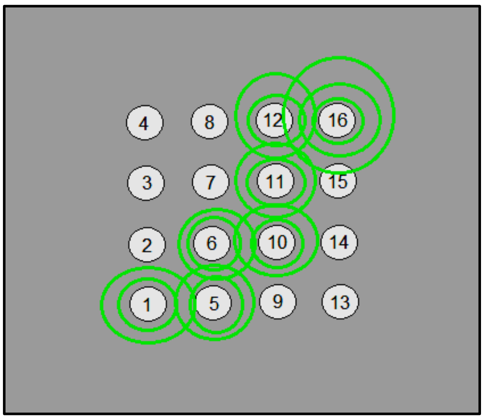

Figure 5: Grid based Network Model

In Grid based network, sixteen sensor nodes are placed equidistant in an area of 100m *100m, and all devices are IEEE 802.15.4 PHY and MAC compliant, as shown in figure 5. All devices are associated with some fixed amount of cache to store and update it with fresh data.

Table 2: Radio Characteristics

Operation	Energy Dissipated
Transmit Power	31mW
Receive Power	35mW
Idle Power	30mW

In general the IEEE 802.15.4 based sensor node can be in one of these modes: Idle, Transmit Mode, or Receive Mode. In the simulation we set the BO=3 and the SO=2, for which the Duty Cycle comes out to be:

Duty cycle= $2^{-(BO-SO)} = 2^{-(3-2)} = 2^{-1} = 2^{-1} = \frac{1}{2} = 0.5$ or 50%. A sensor node with very low duty cycle goes to sleep mode until a beacon signal arrives again to wake up the radio.

Table 3: Simulation Parameters

Parameter	Default Value	Range	
Number of Nodes	100	1~100	
Number of Data Items	500		
Payload Size	64bytes		
PHY and MAC Layer	IEEE802.15.4		
Channel Frequency		2.4GHz	
Bandwidth (kbps)		250	
Waiting interval (t_w)		10s	
TTL		300s	100~300s
Cache Size (KB)		800	200~1400
Traffic Type		CBR	
Routing Protocol		AODV	
Beacon Order		3	
Superframe Order		2	

The data items are updated at the source nodes. The source serves the requests on First-Come-First-Serve (FCFS) Basis. Once the node sends the data item to a sink node the TTL value of the data is also piggybacked along with it to the requester/sink. As soon as the TTL expires, the sensor node has to update its cache space with new refreshed data either from the source or from other nodes (which have maintained the data in its cache) before serving the query.

The radio parameters and the simulation parameters are illustrated in table 2 and table 3 respectively.

Performance Metrics

The following three performance metrics have been evaluated:

Average Query Latency$\left(T_{q_{avg}}\right)$: The query latency $'T_q'$ can be defined as the time interval between a query sent by a requester and the response received back by the requester/sink. The average query latency is the query latency $'T_q'$ averaged over all the generated queries.

Byte Hit Ratio(B): The byte hit ratio B is defined as the ratio of total bytes of data retrieved from the cache to the total number of requested data bytes by the sink/requester node. The byte hit ratio B includes bytes retrieved from a local hit known as a Local Byte Hit(B_{local}); bytes retrieved from a zone hit known as a Zone Byte Hit(B_{zone}) and the bytes retrieved from a remote hit known as a Remote Byte Hit(B_{remote}).It is to be noted that bytes retrieved from a Global hit is a freshly sensed data and cannot be considered to have been retrieved from the cache.

Total Energy Consumption(E_{total}): Total energy consumption is defined as the algebraic sum of energy consumed by each IEEE 802.15.4/ ZigBee sensor mote during transmission, reception, and sleep modes. For simulation purpose, we consider the unit of energy consumption in mWh.

$$E_{total} = E_{transmit} + E_{receive} + E_{sleep} \qquad \ldots\ldots(4)$$

RESULTS

Figure 6 shows the effect of varying cache sizes on the Average Query Latency for both the analytical network models that is Star/ Cluster connected Network and the Grid Based Network. The simulation reveals that the average query latency$\left(T_{q_{avg}}\right)$ decreases with increase in Cache Size as more number of requests is satisfied.

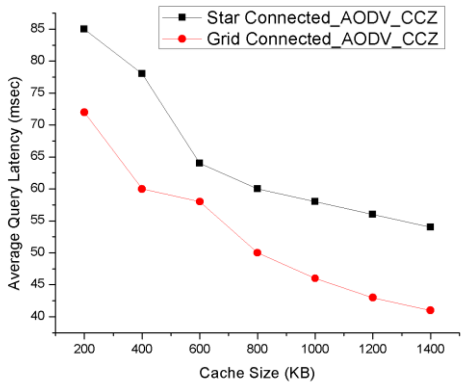

Figure 6: Average Query Latency with different Cache Size

This is due to the fact that more required data items can be found in the local cache as the cache size increases. The average query latency in the Grid/Peer-to-Peer Connected Network is less as compared to the Star/Cluster Based Network. Thus, on an all the performance of CCZ caching algorithm in Grid Connected Network outperforms than the Star Connected Network.

Figure 7 shows the effect of varying Cache Size on the Byte Hit Ratio. Both the schemes that are Star Connected Network and the Grid Connected Network exhibit better byte hit ratio with increasing cache size. When the cache size is small then data is contributed more by zone hit and remote hit but as soon as we increase the cache size the contribution by local hits become significant.

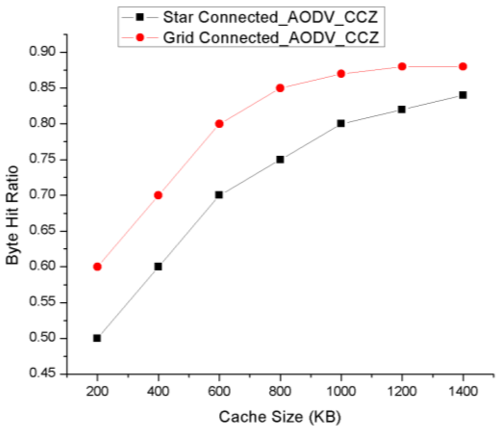

Figure 7. Byte Hit Ratio with different Cache Size

This is because more number of data items are found in the local cache as the cache gets larger. The local byte hit ratio increases with the increase in cache size because with larger cache storage more data can be cached locally. According to the simulation results the Byte Hit Ratio of Grid Connected Network is better than that of the Star/Cluster Connected Network.

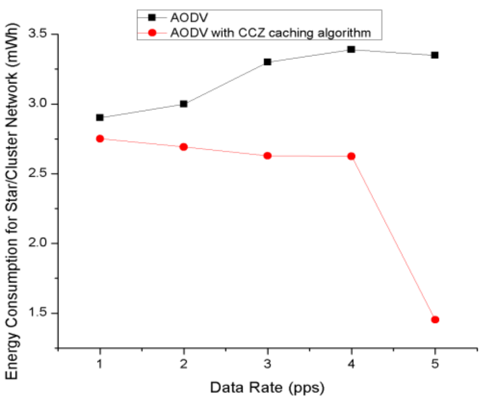

Figure 8: Energy Consumption in Star/Cluster Network

Figure 8 shows the results of total energy consumption by nodes in the network connected in Star (Cluster) connection as shown in figure 4. The energy consumption by the nodes is evaluated when only AODV is evaluated and also for the evaluation of AODV protocol with CCZ cooperative caching algorithm. In figure 8, for different data rate that is packets per second the energy consumption in nodes is evaluated when the simulation is conducted with AODV and figure 9 estimates the energy consumption when the AODV routing protocol is combined with the CCZ caching algorithm. So we see that when data rate is 5 packets per second then the energy consumption by nodes is less when AODV is combined with the caching algorithm; it is because the data is retrieved by the requester from a nearby caching node.

For a Star/Cluster network, the average energy consumption by nodes with only AODV routing algorithm was 3.19mWh. And when AODV routing protocol is used with the CCZ caching algorithm then the average energy consumption is 2.43mWh.

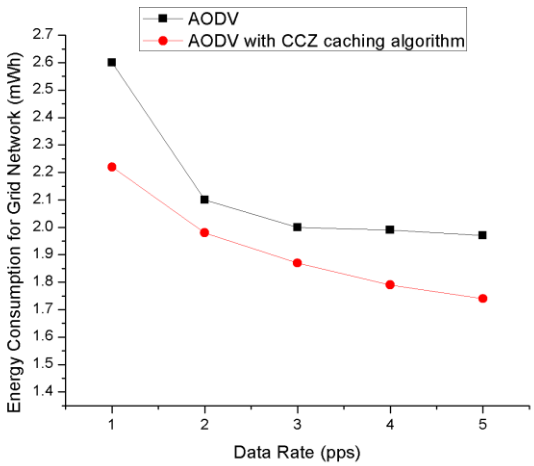

Figure 9: Energy Consumption in Grid Network

For a Grid Network, the average energy consumption by nodes with only AODV routing algorithm was 2.13mWh. And when AODV routing protocol is used with the CCZ caching algorithm then the average energy consumption is 1.92mWh.

CONCLUSION

In this paper two analytical models in which the nodes are organized as cluster (star) and grid (peer-to-peer) based networks are analyzed in beacon-enabled mode in which the Beacon Order is 3 and the Superframe Order is 2. The duty cycle for such a system is 50% and the performance of the proposed two network models are evaluated with AODV routing protocol along with a Cooperative Caching algorithm known as the Caching in Cooperative Zones (CCZ). This cooperative caching technique supports efficient data dissemination and query processing. An effective cache management scheme is proposed that comprises of cache discovery, cache admission control, cache replacement policy and the TTL based cache consistency model. The CCZ algorithm ensures that a query is served from the nearest cache or source. Simulation results show that the CCZ caching scheme performs better in grid based or peer-to-peer network model than the cluster based or star network model.

REFERENCES

[1] Charan, P. , Usmani, T. , Paulus, R. and Saeed, S. "Performance Evaluation of AODV Protocol for Energy Consumption and QoS in IEEE 802.15.4 Based Wireless Sensor Network Using QualNet Simulator" *Wireless Sensor Network*, Vol.8 Issue 8, 166-175, 2016. doi: 10.4236/wsn.2016.88014.

[2] Chand, N., "Cooperative Data Caching in WSN", *International Journal of Computer, Electrical, Automation, Control and Information Engineering*, Vol. 6 Issue 3, 287-291, 2012.

[3] A. Tripathi, P. Gupta, Aditya Trivedi and Rahul Kala, "Wireless Sensor Node Placement Using Hybrid Genetic Programming and Genetic Algorithms," *International Journal of Intelligent Information Technologies*, Vol. 7, No. 2, pp. 63-83, 2011.

[4] A. Abbasi and M. Younis, "A Survey on Clustering Algorithms for Wireless Sensor Networks," ACM Journal of Computer Communications, Vol. 30, No. 14-15, pp. 2826-2841, 2007.

[5] T.P. Sharma, R.C. Joshi and M. Misra, "Dual Radio Based Cooperative Caching for Wireless Sensor Networks," IEEE International Conference on Networking, pp. 1-7, 2008.

[6] E. Fasolo, M. Rossi, J. Widmer and M. Zorzi, "In-Network Aggregation Techniques for Wireless Sensor Networks: A Survey," IEEE Wireless Communications, Vol. 14, No. 2, pp. 70-87, 2007.

[7] N. Kimura and S. Latifi, "A Survey on Data Compression in Wireless Sensor Networks," International Conference on Information Technology: Coding and Computing, Vol. 2, pp. 8-13, 2005.

[8] H. B. Jang, A. Kashif, M. Park, and S. W. Chung, "Reliable Cache Memory Design for Sensor Networks.", 3rd International Conference on Convergence and Hybrid Information Technology, Vol. 1, pp.651-656, 2008.

[9] J. Li, S. Li and J. Zhu, "Data Caching Based Queries in Multi-Sink Sensor Networks," International Conference on Mobile Ad-hoc and Sensor Networks, pp. 9-16, 2009.

[10] J. Xu, K. Li, Y. Shen and J. Liu, "An Energy-Efficient Waiting Caching Algorithm in Wireless Sensor Network," International Conference on Embedded and Ubiquitous Computing, Vol. 1, pp. 323 - 329, 2008.

[11] Md. A. Rahman and S. Hussain, "Effective Caching in Wireless Sensor Network," International Conference on Advanced Information Networking and Applications Workshops, Vol. 1, pp. 43-47, 2007.

[12] T.P. Sharma, R.C. Joshi, Manoj Misra, "Cooperative Caching for homogenous wireless sensor networks", International Journal of Communication Networks and Distributed Systmes, Vol. 2, No. 4, 2009.

[13] N. Dimokas, D. Katsaros, L. Tassiulas and Y. Manolopoulos, "High Performance, Low Complexity Cooperative Caching for Wireless Sensor Networks," Springer International Journal of Wireless Networks Vol. 17, No. 3, pp. 717-737, 2011.

[14] N. Dimokas, D. Katsaros and Y. Manolopoulos,"Cooperative Caching in Wireless Multimedia Sensor Networks," *Springer Journal of Mobile Network Applications*, pp. 337-356, 2008.

[15] P. Kumar, N. Chauhan, L.K. Awasthi, Narottam Chand, "Proactive approach for Cooperative Caching in Mobile Adhoc Networks", *International Journal of Computer Science Issues*, Vol. 7, Issue 3, No. 8, pp. 21-27, 2010.

[16] Charan, P., Usmani, T. and Saeed, S.H., "A Comprehensive Study of Various on Demand Routing Protocols for MANETs", *International Journal of Electronics and Communication Engineering*, Vol. 4, pp. 1-12, 2015

[17] C.E. Perkins & E. M. Royer, "Ad hoc on-demand distance vector routing" *Proceedings of IEEE workshop on mobile computing systems and applications*, pp. 90-100, 1999.