An efficient caching technique for wireless sensor network based on Clustering and data aggregation

Mohammed A. Mohammed, Ahmed I. Saleh, Mohammed F. Al Rahmawy

ABSTRACT : Wireless sensor networks(WSNs) are used nowadays on a large scale for the low cost and efficiency in building many applications. These networks have limited resources i.e limited memory, battery power (energy) and insufficient bandwidth these limitations make these networks suffer from many problems e.g non data availability and accessibility, long latency and frequent network failure. Several techniques exit to overcome these problems and to increase the efficiency. One of these techniques is the cooperative caching for sensed data; this is a significant process to store the frequently accessed data items in the WSNs. This paper Suggests a novel technique for caching data in wireless network called Caching Group Scheme (CGS) that is based on cooperative caching to cache data. The simulation results show that this new technique is more efficient than other existing techniques in terms of byte hit ratio, time to query response (latency) and energy consumption .

Keywords: wireless sensor network, cooperative caching, cache discovery, cache admission control, cache replacement policy.

1- INTRODUCTION

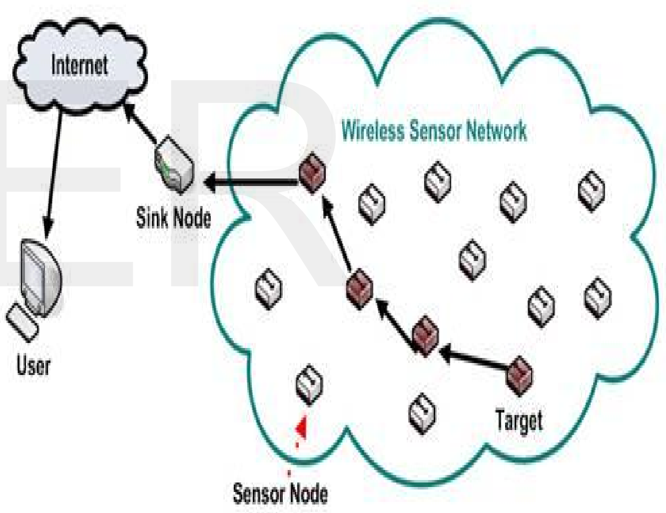
Wireless sensor networks(WSNs) is a new technology which is expected to dominate in the future due to their data acquisition and data processing abilities [1]. Advances the technologies of digital electronics and computer networks have made revolution developing wireless sensor networks(WSNs), in the last few years. This type of networks sparked interest quickly for many reasons First ; due to the low cost for building these networks, as sensor wireless networks (WSNs) consist of a large number of small and low cost sensors that act as network nodes. These nodes make a self-regulating process to form a network that uses a wireless system to communicate with each other, Second; due to the large number of applications that can use this type of networks such as in surveillance applications, disaster relief, smart buildings, medical care, tracking on the battlefield and the weather forecasting .

A sensor node is the basic unit in these networks; Wireless Sensor Networks (WSNs) comprises different elements, including low cost sensors, processors, flash memories and transceiver. All the components of each individual WSNs node/element are integrated by the micro-electro-mechanical system (MEMS) into a single substrate known as wireless sensor node (SN) .

•Mohammed A. Mohammed is currently pursuing masters degree program in computer engineering and systems in Mansoura University ,Egypt, E-mail: malzuharey@gmail.com

• Ahmed I. Saleh ,Assistant prof in computer engineering and systems dept. Faculty of Engineering, Mansoura University, Egypt, E-mail: aisaleh@yahoo.com

•Mohammed F. Al Rahmawy ,Ph.D. in computer science dept. in Faculty of computer and Information, Mansoura University, Egypt , E-mail: mrahmawy@gmail.com



Fig(1) wireless sensor network[2]

Sensor nodes exchange the information using wireless medium (e.g. radio frequencies, infrared)[2]. However there are some of the drawbacks or challenges of WSNs; in the forefront of those challenges is the need for efficient energy utilization of the batteries of those devices (nodes) as the batteries used in these devices have limited power and size because the devices(nodes) or networks are always deployed in inaccessible and inhospitable environments. This is a big challenge as majority of WSN applications are data centric, i.e they require transmission of sensed data to a sink node situated inside/outside the sensing field, through single/multi hop transmission. Once the data is collected at a sink, it preprocessed as per requirement of the application and further transmitted to the end user via Internet or some other wireless/wired media[3]; this leads to a large number of transmit/receive operations that increase the energy consumption. Failing to satisfy the energy requirements of WSNs nodes leads to inefficient data dissemination and non-availability of data and data loss during transmission and reception; resulting in a decrease of the overall performances and network lifetime(the time until the first/last node in the network depletes its energy or the time).Due to these challenges; we proposed in this work a set of novel efficient techniques and protocols for caching and routing the sensed data in WSNs.

2-BACKGROUND AND RELATED WORK

Caching techniques are efficient solution for increasing the performance in data communication in both wired and wireless networks [4].Since wireless sensor networks have constrained resources (bandwidth, battery power, memory size and computational capacity), they have difficulties in storing large amount of sensed data; therefore, data caching is very important challenging in WSNs .Efficient Caching enhance the performance of wireless sensor networks as, caching frequently accessed data items not only reduces the average query latency, but also saves wireless bandwidth and extends the battery lifetime in such environments. There are many exiting caching techniques; each one of these techniques has its own benefits and limitations ; therefore, many researches have been conducted to enhance the caching techniques and to overcome their limitations in order to enhance the WSNs efficiency.
These researches exploiting data caching either in some intermediate nodes; or at a location nearer to the sink[9] . In the following we discuss some of these techniques:

2-1 TRADITIONAL CACHING

The traditional way of resolving a data request(query) is to check the local cache first and send the request to the data source after each local cache misses. This scheme is not effective because it increases latency and the energy consumption [3].

2-2 ZONE COOPERATIVE CACHE

In Zone Cooperative (ZC)techniques the set of one-hop neighbors in the transmission range forms a zone, all the nodes in this zone forms a cooperative cache. The origin of each data item is held by a particular data source act as server. Each member node inthe zone stores the frequently accessed data items it accesses in its own cache. The data items in the cache satisfy not only the node's own requests but also the data requests passing through it from other nodes in the same zone [5][12] .

2-3 GLOBAL CLUSTER COOPERATIVE CACHING

The Global Cluster Cooperative (GCC) caching partitions the whole network into equal size clusters based on the geographical network proximity. Individual nodes within each cluster interact and cooperate with each other, which enhances the system performance in a the overall network. In each cluster area, a "super" node is selected to act as Cache State Node (CSN)or a cluster head, which is responsible for maintaining the global cache state (GCS) information of different clusters in the network domain. GCS for a network is the list of data items along with their TTL stored in its cache[5].

3- PROPOSED CACHING STRATEGY

The proposed strategy is an efficient cooperative caching technique, which aims to reduce both the energy consumption and the time delay when fetching the requested data by using an enhanced grouping technique inside each cluster to form a Caching Group Scheme (CGS). Which is a region around node that utilizes other nodes' storages to form large cumulative cache (CMC) that is distributed across a set of one-hop node neighbors and within the transmission range of each other, and it use a cache discovery algorithm that identifies the location of the requested data item by using broadcast messages.

3.1- NETWORK MODEL

To apply the proposed caching scheme in WSN, we assume the following:

-The network has a large scale and the communication links are bidirectional.

- The network is divided into clusters of nodes, Where each node has a cluster head. The task of a cluster head in the proposed model is routing data from its cluster or through it to another cluster head, or to sink node. Also it connects its cluster with the other clusters in the network to exchange information with each other.

- A unique identifier is assigned to each cluster head (CH_id) , node (N_id) and to each data item (data_ Item_id).

- Each cluster head contains a small table to store the (CH_Id, data_ item_id, TTL)information and any node in group contains two tables self table and group table .

- In the following ,we discuss how the network is established into clusters as proposed in our model and how the messages are routed in this network .

3.1.1 NETWORK SETUP

The process of setting up the network in the proposed strategy is divided into two main stages: the cluster formation stage and the master node selection stage. This section explains these two stages as follows .

Cluster Formation Stage.

-The network is divided first into overlapping virtual clusters ,where each cluster has a square shape .To determine the dimensions of each cluster . The points are made up in the form of a pairs consisting of the intersection of two axes, i.e the X-axis and y-axis. We assume initial value of X-axis and y-axis and we start divide from left lower.

- Then the process of selecting the cluster head begins among the nodes of each individual cluster. by using Node Degree and Energy-Aware routing protocol (NDEA) as follow see figure(2) :

1- Each node in the cluster broadcast a helloID_msg to its neighbor nodes in the same cluster this message contains the ID of the node .

2- Each node has an ID table is built to hold the received message .

3- The residual energy of each node is checked , if it is less than a pre-specified energy threshold min(E), the node is declared to a member node ,otherwise, it is considered candidate cluster head[6] .

4- The cluster head is elected from the candidate nodes by choosing the node that has the highest maximum performance index W, where the performance index W(n) of each individual candidate node is calculated using equation [1] that combines node connectivity level ,and memory size for this candidate node.

$$W(n) = \omega_1 f(n) + \omega_2 g(n) + \omega_3 h(n) \qquad (1)$$

where ω_1, ω_2 and ω_3 are weights specified by the network administrator as required per application to specify the relative importance of the node connectivity level f(n), residual energy g(n) and memory size h(n) of a certain node n respectively.

The cluster head election process is processed using a distributed algorithm, where each individual candidate node broadcasts its performance index in a helloweight_msg to the other candidate nodes. Then, the performance index in each candidate node is compared with the performance index values received in this node from the other candidate nodes. The candidate node that finds its performance index bigger than the received performance indices declares itself as a cluster head by sending a broadcast message with its ID to all other nodes in the cluster; otherwise, it is considered a member node [5].

5- In case two nodes or more have the same highest performance index value, the conflict is solved by choosing the node with the lowest ID value; this can be discovered when an candidate cluster head node receives a message contain id of other candidate cluster head nodes. the node compares its ID with the ID(s) received, if it founds its ID has low value its broadcast cluster head declaration message ; otherwise, it keeps itself as a cluster member.

6- In order to build route to sink nodes, the cluster head CH sends its ID to sink node through CHs on the path to setup routing tree.

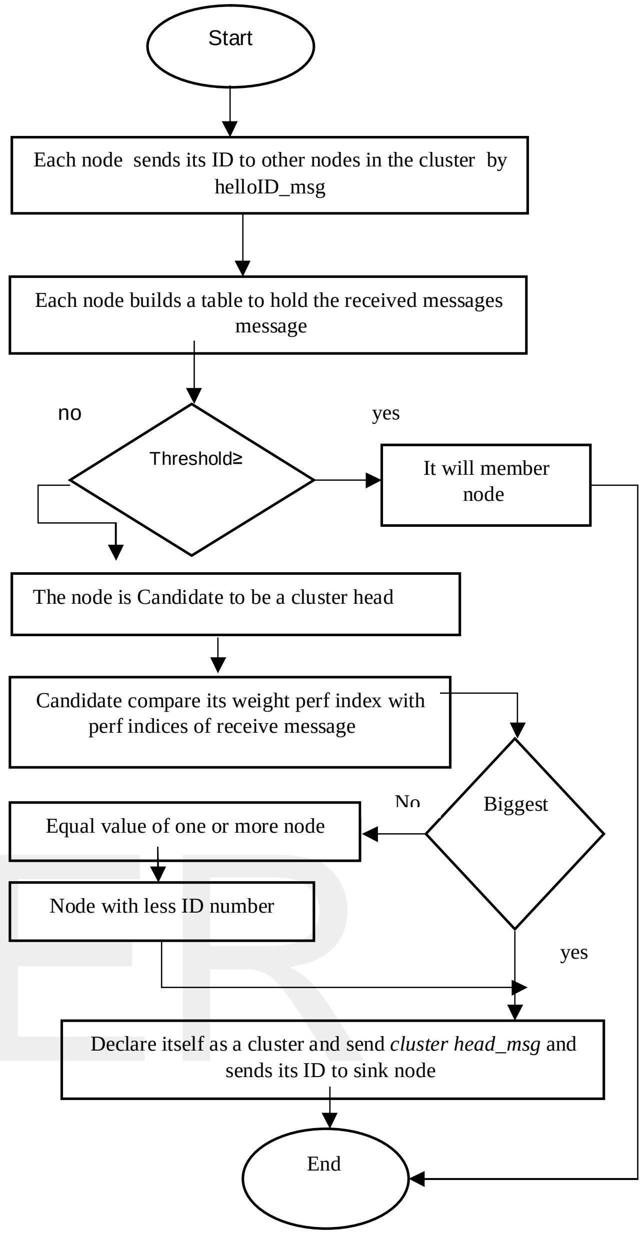


Figure (2) Flow Chart of Cluster Formation stage

Master Node Selection Stage.

The process of selecting a master node is a main contribution in this research. The aim of master node selection stage is to form a caching group from nodes around it within its transmission range, and management of caching process in this region (caching group) and trans and receive information from cluster head in its cluster. The steps of selecting master node process are; see figure (3) :-

1- The candidate nodes other than the cluster head send their performance index value to the cluster head .
2- cluster head selects from the received values the node that has the highest performance index to become a master node in the cluster; then, it sends a request message to this node which accepts in turn this request message.
3- The selected master node broadcasts a message to all other nodes in its cluster to declare it- self as a master node; this message has the master node's ID.
4- The selected master node form caching group by sending a caching control message to each node in its one-hop neighbor to utilize cache of these node as a big cache (cumulative cache)[4] .
5- master node reschedules to assign timeslots for the sensor nodes to transmit and receive message from remaining nodes (outside caching group) to cache sensed data in caching group and then to cluster head to arrive to sink node .

3.1.2 - Routing setup

Routing setup describes how data is sensed, aggregated and routed from source node to sink node or base station as follow :-
1- After network is formed the actual data transfer begins . The source node senses the events and stores them in its cache for some time .
2- The source node sends its data to master node to store it in caching group .
3- If the master node has cache space, it stores these data; otherwise it uses cache replacement policy.
4- The master node aggregates the data and removes redundancies from the stored data item.
5-If the data requested by a sink node, it sends a query to corresponding cluster head that routes it to another cluster head on routing path until sink node .
6- These data item, if not used for long time or has least quality, are remove from caching group to accommodate cache to store a new data item as shown in figure(4).

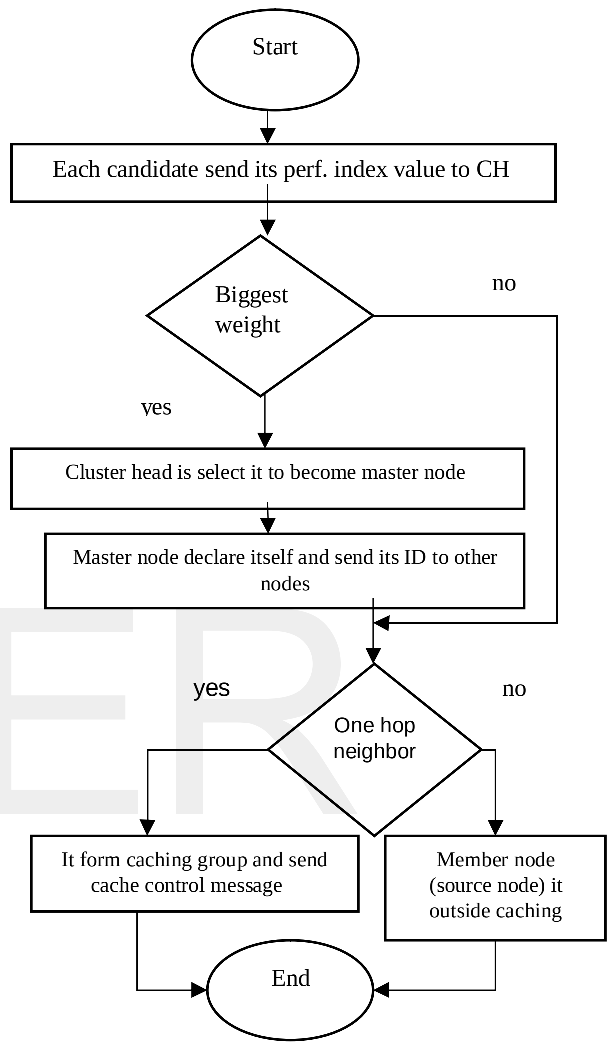


Figure (3) Flow Chart of Master Node Selection Stage

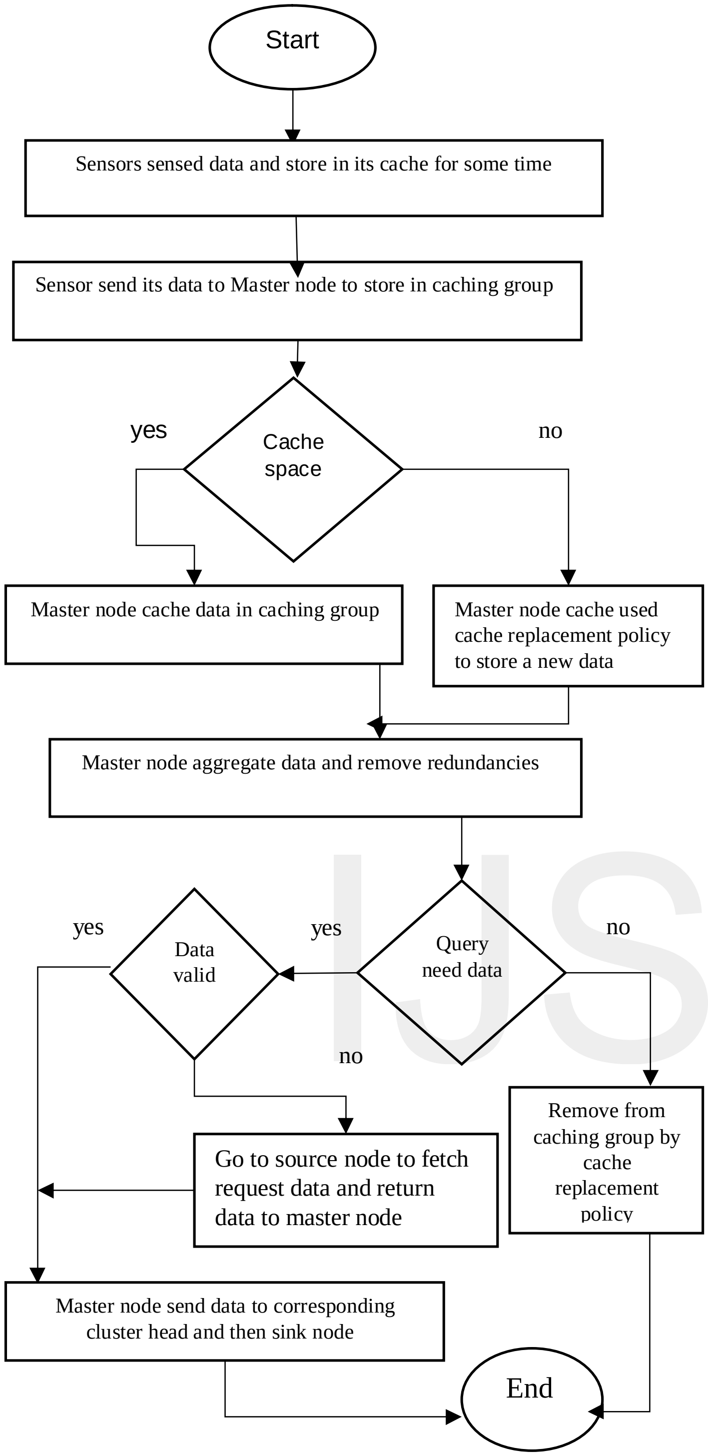


Figure (4) Flow Chart of query resolve

3.2- THE PROPOSED COOPERATIVE CACHING SCHEME

We have proposed an efficient cooperative caching technique which aims to reduce the energy consumption and the time delay of fetching the requests data, by using a group inside each cluster that forms a caching group (CG). The Caching Group (CG) is a region consisting of all nodes around a master node, where there is only one master node for each cluster, this master node is chosen by the cluster head as explained later in section network setup . All the nodes within a caching group form a large cumulative cache (CMC) and all the nodes within the CG are one hop neighbors and within the transmition range of each other .

The Caching Group uses a cache discovery algorithm to look for the locations of the required data item(s) by broadcasting a query to all the nodes in the sensor network[4]. This caching group acts as a big cache store, because it uses cooperative caching to cache data in the nearest place from the cluster head. So ,each caching group play as cluster helper. Because, it aggregation the data and transfers it to its cluster head. Which reduces the workload of the cluster head, and reduces the time delay of fetching the requested data item. Cooperative caching is more useful as the cached data at sensor node may also be shared by the neighboring nodes in each cluster . In the Caching Group (CG), each node and its one-hop neighbors form a group within transmission range by sending "Hello" messages periodically. In order to utilize the cache space of each node in a group and to exchange the caching status in a group periodically, the master node checks the caching status of group members using information message send to member nodes and the member nodes return or replay message to master node ,this message describe caching status in member nodes called caching control message. The caching control message contains the fields:
{ Cached data id, Timestamp, Remaining available cache space}[6] .Hence, the caching group stores more different data items and increases the data accessibility. In order to record the caching status of group members, each node maintains a self_ table and a group_ table to record caching status of itself and its group members respectively.

-Advantages of the proposed technique (The Caching Group)

I -Reduce the time delay because the cluster head (CH) instead of transferring and receiving messages to / from all nodes (member nodes)in its cluster, it exchange messages only with master node .

II- Reduce the workload of the cluster head .Each caching group works as a big cache store, because it uses cooperative caching to caching data in the nearest place from the cluster head; thereby, these groups play role as a cluster helper . (CG) reduces the workload as it becomes responsible for the fetching and data aggregation and delivering the data to cluster head. The cluster head becomes only responsible for the routing it to the sink node, which reduces the energy consumption at the cluster head (CH only store data that passes through it form other cluster heads to the sink node).

III - Prolong the lifetime of the network, because the transmission and receiving inside the network decreases and by using cooperative caching, it maintain the energy of each nodes in the network thereby the lifetime of the network is prolonged.

3.2.1- Cache Discovery Algorithm

When the query arrives to the CH from the sink node it sends a query to the master node to ask if the requested item exists in the CG ; if the item is found in the CG , the master node sends it as a replay to the cluster head ,which forwards the received item to the sink after inserting its own CH_id in the frame holding the replay. In case the item is not found in the CG ,the cluster head checks its internal routing table to check if this data item passed through it ; If the item is found passed through this cluster head ,the cluster head sends a replay message to the sink telling it that the item is not found and associate with this message the route of the path to the location of the data item extracted from the cluster head .

The details of the cache discovery algorithm is shown in figure (5),When a data request is initiated at a sink, it first looks for the data item in its own cache In case data item is not found or missed it Looks for the data in the CG within its own cluster. If the data item is found the query is immediately served without broadcasting it any further, otherwise, the query is sent to the next cluster. The CH resides within the next cluster searches in its cache if the data is found the query is served . In case the data item is not found in the caching group, it searches in its routing table and checks if such data was routed through it(simply by matching the CH_id of queried data item with CH_id in the respective routing table) .In case the queried item does not match with any of CH_id in the routing table, further broadcast of query is stopped.
This process avoids query broadcast through those routes where data in not residing thus saving considerable amount of energy in the network. Only the cluster head that found data passed through it, sends a broadcast query to the respective cluster head that has required the data item and the CH sends the required data item to the sink node, thereby, the query is resolved . This process avoids query broadcast through those routes where data in not residing thus saving considerable amount of energy in the network. Only the cluster head that found data passed through it, sends a broadcast query to the respective cluster head that has required the data item and the CH sends the required data item to the sink node, thereby, the query is resolved .
In case the data was found in any place on the routing path but it evict because TTL expired, the CH_id and data item_ID of it stays in the routing table that Identifies the routing path to the cluster that has the required data .

3.2.2-Cache Admission Control

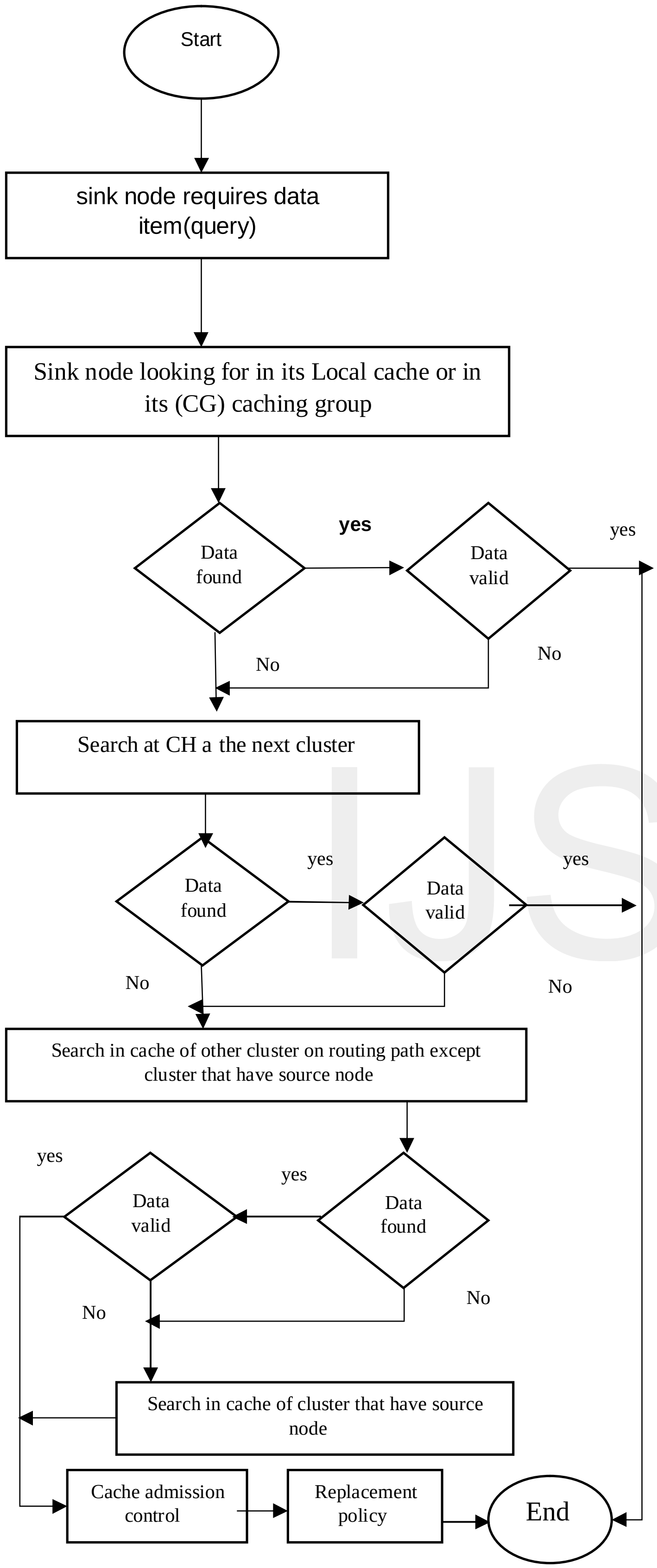
Cache admission control determines whether an incoming data item from another cluster should be stored into the cache of the nodes or not . Inserting a data item into the cache might not always be favorable because incorrect decision can lower the probability of cache hits and also makes poor utilization of the limited storage[1]. replacing a data item that will be accessed soon with an item that will not be accessed in the near future degrades performance. the cache admission control allows a node to cache a data item based on the location of data source or other node that has the requested data such as sink node (distance or number of hops) . Let S be the network size in terms of hop .we calculate the DSi for node ni at a distance Di from sink. So the distance function :

$$DSi=1-Di/S \quad (2)$$

In our proposed scheme ,we used a threshold value = 0.5 .
A particular data item is to be cached or not is based upon distance function DSi. If the distance function DSi > threshold then cache the contents otherwise not .

3.2.3 - Cache consistency

Cache consistency ensures that sensor nodes only access valid states of the data and no stale data is used to serve the queries. Two widely used cache consistency models are the weak consistency model and the strong consistency model[7].Since in the multi-hop environment, limited bandwidth and energy constraints in wireless sensor networks, it is too expensive to maintain strong consistency, the weak consistency model is more used. Our scheme uses a weak consistency model based on the TTL. In this scheme a SN considers a cached copy up-to-date if its TTL has not expired . The node discards a data item if its TTL expire .



Figure(5),Cache Discovery Algorithm

3.2.4 - Cache Replacement Policy

Cache replacement policy is required when a caching node attempts to cache the data item but its cache is full. We proposed a new cache replacement policy as follow :

The master node checks periodically for the data items to be evicted from the cache. We proposed a new method called the Least Quality Index (LQI), this method aims to select the data items that have the least importance for the system. instead of evicting the data items with the least TTL only as done in [3],our proposed method works with two techniques.

First : The master node sends message to all nodes in caching group to check for any data item has the least TTL in cache of each node in caching group, and then nodes replay message to master node. The master node account the Quality Index (QI)value using equation (3) for these data item and compare among them; data item has least value (QI) is evicted from caching group based on following factors :-

- Importance (M) : The entropy is a measure of the importance of the data item (entropy in the information theory is logarithmic measure of the rate of transfer of information in a particular message or number of bits needed to store a particular message). The lower entropy should be preferred for replacement [9] .

- Distance (D) : is measured as the number of hops from a responding node such as source node or cache node to a requesting node (sink node) . Increasing the distance increases the importance of the data item (i) .
- Size of data item (Si) : is the size of a data item . If the size of a data item is bigger than another, it becomes a candidate to be removed from the cache . Based on these factors the Quality Index
(QI) value of data item (i) is calculated using the following expression

$$(QI)i = \frac{Mi\ Di}{Si} \quad (3)$$

The objective is to maximize the Quality index (QI) for the data items kept in the cache. We evict the cached data item (i) that has the least Quality Index until the free cache space is sufficient to accommodate the incoming data .

Second :This way uses a threshold algorithm to determine which data item should be evicted from caching group, this algorithm returns two values that have higher score. In the start the it arranges data items based on size of data item factor ,and then set a threshold (t) value by using an aggregation function which compares between the return values and a threshold (t) value. If these values(return values) greater than or equal to a threshold (t) value, the algorithm is stop, thereby return values of two data items(higher values) and then Enter these values into the equation (3) it account any data item has least Quality Index to evicted it from caching group, otherwise it will continue until this condition is validate, see figure (6) :-

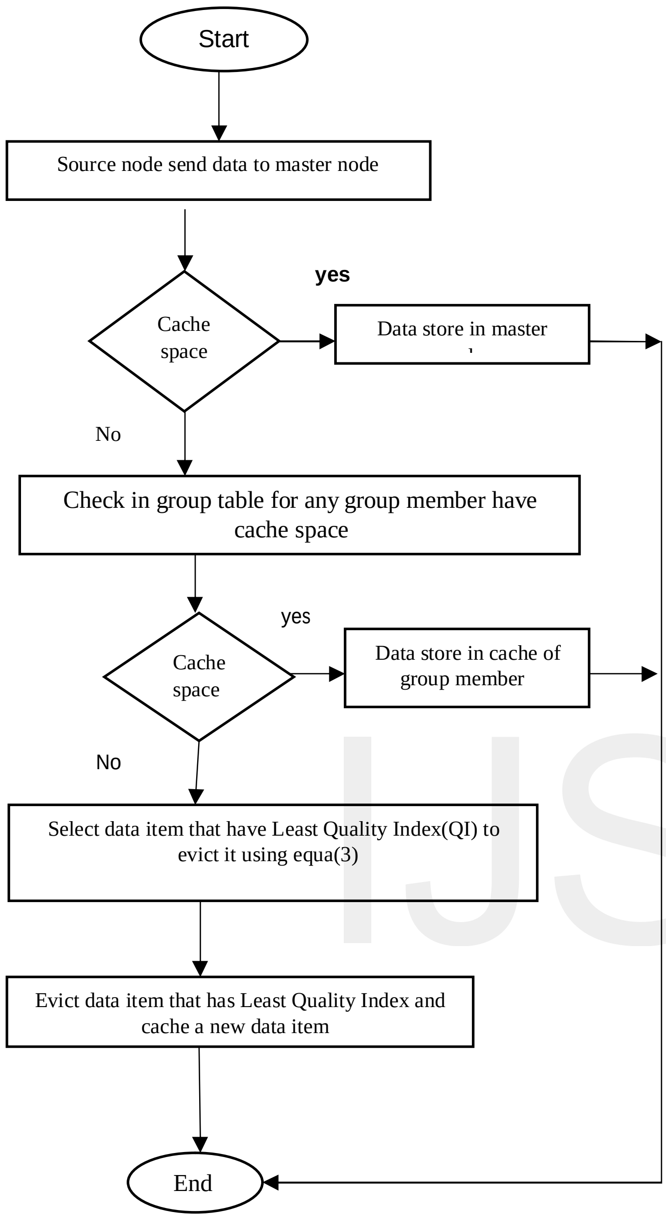


Figure (6) Flow chart of cache replacement policy

4 - PERFORMANCE EVALUATION:

The evaluation is divided into two main parts. The first part compares the proposed CGS scheme with two well know caching schemes CHIC [3], NICoCa [3] in terms of average query latency, byte hit ratio against both cache size and the mean generate query time.

Then, in the second part , we compare the effect of using CGS and CHIC on both the energy consumption and the packet delivery ratio, using NS2 simulator .

4.1 - Simulation Parameters

In our simulations, 400 sensor nodes are randomly deployed in a square area that has 100*100 dimension. The nodes and the network have the a same initial energy 2 J. Various simulation parameters are listed in Table 1.

Parameters	Default value	Range
Network length	100 meters	50-400 meters
Number of nodes	400	100-500
Initial energy of node	2 Joule	
Data packet size (k)	100 byte	
Threshold distance (d0)	87 meters	
Cache size	800 KB	200-1400 KB
Time of live(TTl) of data	300 sec	
Data rate	10 Kbps	
Transmission range	10 meters	
Zipfian skewness parameter(Z)	0.8	
Mean query generate time	5 sec	2-100 sec

Table (1) simulation parameters

4.2 - Performance Metrics

In the evaluation and the simulation results, we adopted the following metrics:-

•Average Query Latency

The query latency is the time elapsed between the query is sent and the data is received to the sink, and the average query latency(Ta) is the query latency averaged over all the queries[9].

• Byte hit ratio (B)

Byte hit ratio is defined as the ratio of the number of data bytes retrieved from the cache to the total number of requested data bytes. It is used as a measure of the efficiency of cache Management [3].

• Average Energy ratio

The energy metric is taken as the average of energy consumption calculated through the simulation time for the total energy of the network.

•Packet Delivery Ratio (PDR):

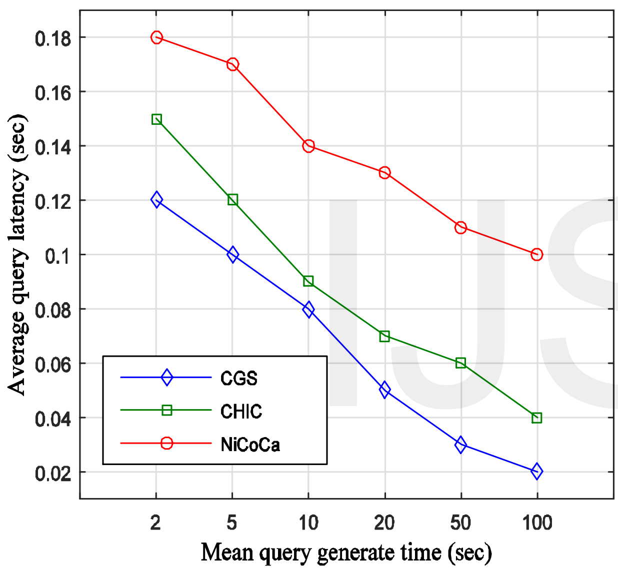
This network performance metric, is defined as the ratio between the number of data packets successfully delivered to the destination and the number of packets transmitted by the source[11].

4.3 - Results

Here we show the effects of mean query generate time and cache size using various performance metrics on our proposed scheme and compare these results with CHIC and NICoCa for WSN .

• Effect of mean query generate time on average query latency

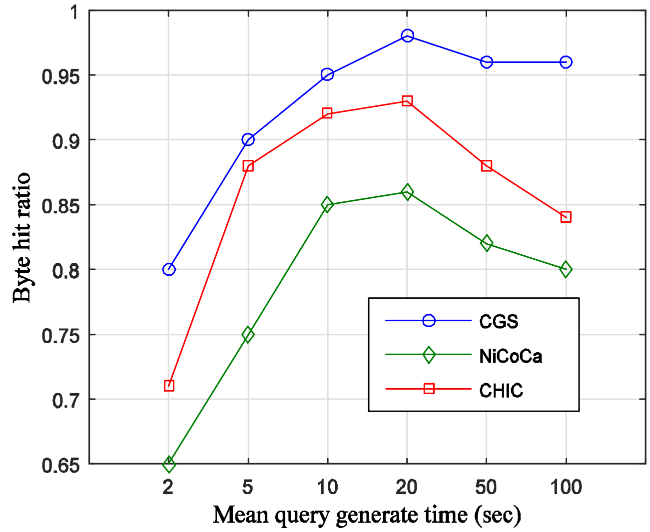
The effect of mean query generate time (Tq) on average query latency (Ta) has been illustrated in Fig(1). Mean query generate time is the time between two consecutive queries averaged over total number of queries[3]. We can be see, from fig(1) that when mean query generate time (Tq) value is very small, more number of queries are generated per unit time, and because a very little time available for settlement of cache to make required data items available. Hence, only the queries that require data items near the sink node may be resolve, while the queries that require far data items are not resolve. When the mean query generate time (Tq) increases, the average query latency (Ta) decreases. The proposed scheme(CG) outperforms other schemes under different mean query generate time (Tq) settings.



Fig(1) Effect of mean query generate time on average query latency

• Effect of mean query generate time on byte hit ratio

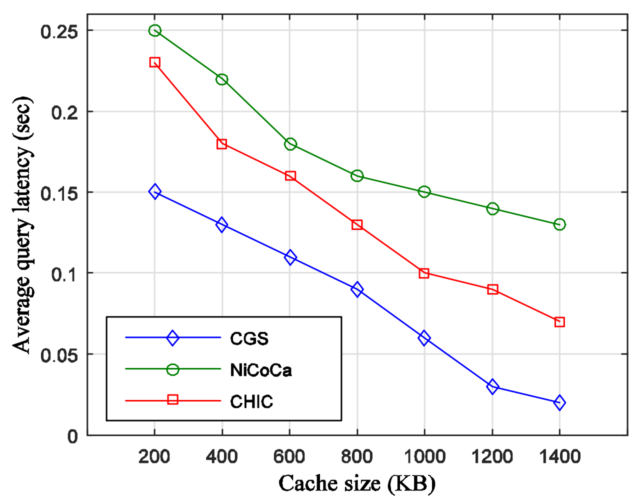
The byte hit ratio increases with increase in (Tq) and byte hit ratio reaches its maximum value around 20 and then again starts reducing. In this case the network has to resolve less queries per unit time, because of the long time between any two consecutive queries, which gives enough time to get or provides frequently queried data item from the cache. This accounts for higher byte hit ratio. However, if Tq is further increased, it does not result in better hit ratio,. The proposed scheme is the best among the other schemes because using caching group (CG) increases cumulative caching in the cluster as shown in fig(2).



Fig(2) Effect of mean query generate time on Byte hit ratio

•Effect of cache size on average query latency

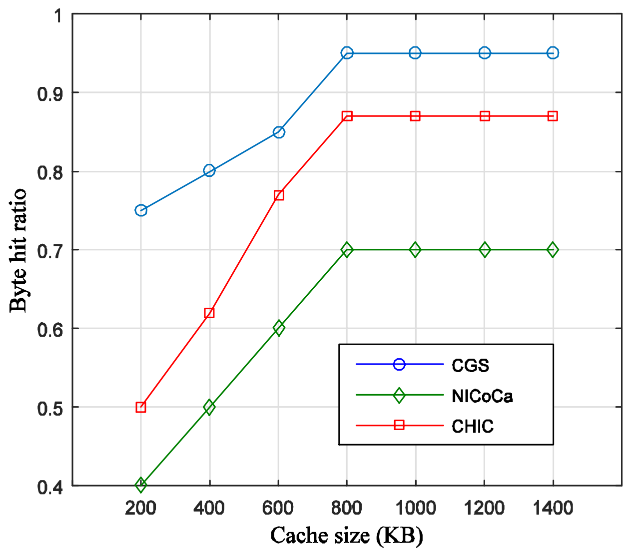
Fig(3) shows the effect of cache size (KB) on the average query latency (Ta), when the cache size is small the average query latency (Ta) increases especially when the required data items are far from sink node. when cache size , increases more number of data items are cached, thereby decreasing the average query latency (Ta). The proposed scheme behaves better than the other scheme because using caching group (CG) increases the cumulative cache and increases the cache size which makes access to the required data faster, leading to reduction in the thereby, which leads to better performance over the other schemes as shown in the fig bellow.



Fig(3) Effect of Cache size on average query latency

• Effect of cache size on byte hit ratio

Fig. (4) indicates, that all the caching scheme gives better byte hit ratio if the cache size is increased. Due to the cumulative caching within the clusters and the better adapting replacement policy, the (CG) scheme outperforms the other schemes for different cache size settings, as the proposed cooperative caching policy retains more useful data in the caches of nodes, thus increasing the overall byte hit ratio. For the proposed scheme (CGS) the increase in the byte hit ratio is clear with the increase in the cache size up to a cache size of 800 KB and then Close at a certain level .

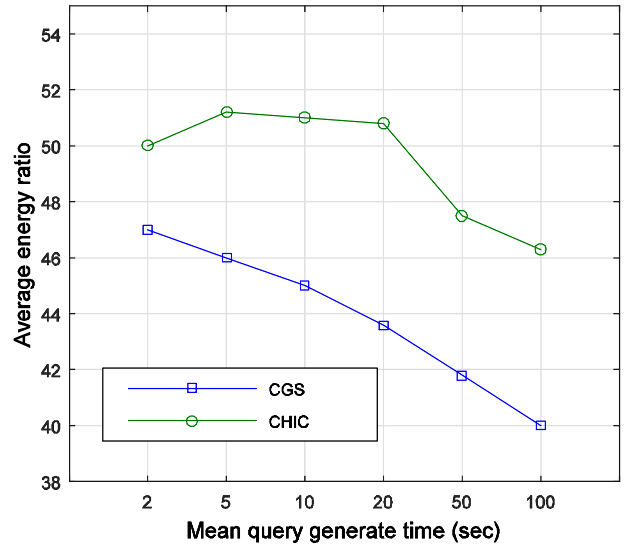


Fig(4)effect of cache size on byte hit ratio

The energy consumption of the nodes is one of the most important factor in evaluation the caching scheme, as the increase in the energy consumption may lead to a network failure or at least to a decrease in the lifetime of the network, and vise versa reducing the energy consumption prolongs the network lifetime and enhanced the performance. Another important metric for the evaluation is the packet delivery ratio, as it measure effectiveness and the performance of the network. In this section we compare the both metrics and study their effects of the both the mean query generate time and cache size on them. We compare these effects in both our scheme CGS and CHIC scheme only as it showed in the previous section that it is more efficient then NICoCa scheme.

• Effect of mean query time on average energy ratio

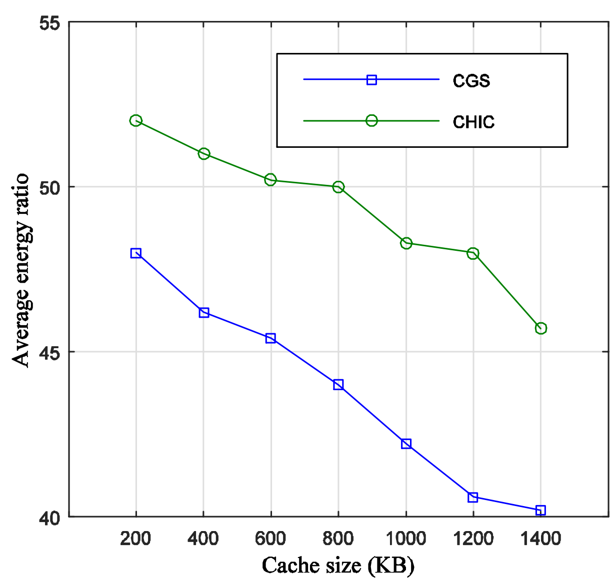
Energy is a very important resource in wireless sensor network because it effects the general performance of network so decreasing in energy consumption increases the lifetime and improves the performance of the network. Fig (5) Illustrates the effect of mean query time on average energy ratio. As seen in the figure, increasing mean query time decreases the energy consumption as increasing the time between two consecutive queries means decreasing the number of queries, so sensors goes in a sleep mode to save their battery power and our scheme (CGS) is better from the other scheme (CHIC), because in our scheme , the used transmit / receive operations are less than the transmit / receive operations in the CHIC scheme. As in CHIC to get a data item , each cluster head sends message to all the nodes within its cluster to ask for the data, while in our scheme, the CGS, the cluster head sends a single message to the master node to ask for the required data item.



Fig(5) Effect of mean query time on average energy ratio

• Effect of cache size on average energy ratio

Fig (6) illustrates the effect of cache size on the average energy ratio, when increasing the cache size of nodes the energy consumption decreases, because the nodes cache more data in their cache, thereby resolve the queries faster and after that go to sleep mode to conserve battery power.Our scheme (CGS) is better from the othor scheme (CHIC) because the CGS scheme provides cooperative caching in nieghboor nodes which provides bigger cache storage and increases the chance of finding the required data item in nieghboor nodes, which reduces the transmit / receive operations within the network, thereby decrasing the energy consumption.

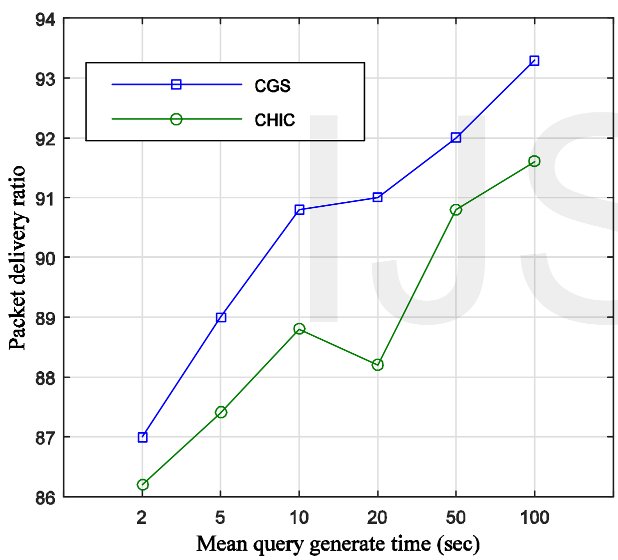


Fig(6) Effect of cache size on average energy ratio

• Effect of mean query generate time on packet delivery ratio

Packet delivery ratio (PDR) is the ratio of the number of data packets successfully delivered to the destination to the number of packets transmitted by the source. It increases when the mean query generate time increases, because the network has enough time to resolve the query (decreasing query latency) and sends more packet from the source to the destination / sink; thereby the packet delivery ratio increases. Our scheme (CGS) is better than the other scheme (CHIC) as illustrated in fig (7) because our scheme does not depend only on the TTL in its cache replacement policy to evict stale data items, but it adopts in its policy other factors such as the data item size, importance and distance from the source.

Also, as our scheme uses cooperative caching of nodes, this gives a better cache hit ratio and less contention due to the reduce no. of message exchange in a group, which means less data packet transmitted by the source which in turn increases the packet delivery ratio.

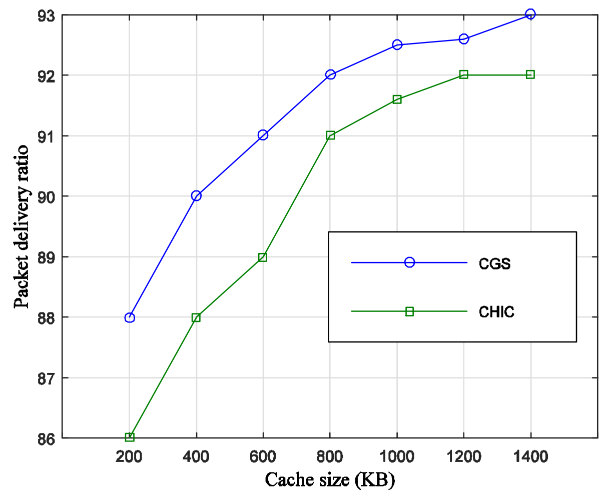


Fig(7) Effect of mean query generate time on packet delivery ratio

• Effect of cache size on packet delivery ratio

Fig (8) illustrates the effect of cache size on the packet delivery ratio, when the cache of senor node increases that means increasing the space of the cache data and increase the packet delivery ratio. Our scheme uses a group of sensors form cumulative cache, this cooperative caching feature increases the total cache and gives better cache hit ratio which leads as stated before to less contention in the network which in turn improves the packet delivery ratio significantly.

Also, using the cache consistency ensures that sensor nodes only access valid states of the data and no stale data to serve the queries and sending successfully packet to destination .



Fig(8) Effect of cache size on packet delivery ratio

5- Conclusion :

A major challenge for WSNs is to effectively use the cache of nodes in the network to cache sensed data or during data collection and query processing. In this paper, we propose a caching group scheme (CGS) for supporting efficient data collection and query processing in WSNs .The scheme can exploit the Cache of all nodes in caching group as one cache in each cluster to form cumulative cache and cluster head act as just router the data to the sink node; which enhance Load distribution in each cluster due to reducing the query latency and energy consumption of the nodes thereby prolong the lifetime of the network. The scheme includes a cache discovery process or cache discovery algorithm that is developed to ensures that a requested data is returned from the nearest cache or source, also it has an admission control to prevent high data replication and minimize the distance between the sink node and the source node, and use a novel replacement policy that helps in improving the byte hit ratio and accessibility. Cache consistency ensures that nodes only use valid data as part of the cache management. Simulation results show that the (CGS) caching scheme performs better in terms of various performance metrics in comparison with CHIC and NICoCa strategies.

References

[1] A Malik, V. K. Nassa , K. Chawla" A Multi-Parameter Scheme to Enhance a Network Life in Wireless Sensor Network's " International Journal of Advanced Research in Computer Science and Software Engineering Volume 4, Issue 8, August 2014

[2] H. Kumar, M. K. Rai " Caching in Wireless Sensor Networks: A Survey" International Journal of Engineering Trends and Technology (IJETT) – Volume 10 Number 11 - Apr 2014 .

[3] Ashok Kumar "Cluster Head Influence based cooperative Caching in Wireless Sensor Networks" Advances in Information Science and Applications - Volume II .

[4] K.Shanmugavadivu , M.Madheswaran "Caching Technique for Improving Data Retrieval Performance in Mobile Ad Hoc Networks" International Journal of Computer Science and Information Technologies, Vol. 1 (4) , 2010, 249-255.

[5] P.Kuppusamy, Dr.K.Thirunavukkarasu, Dr.B.Kalaavathi " A Review of Cooperative Caching Strategies in Mobile Ad Hoc Networks" international Journal of Computer Applications (0975 – 8887) Volume 29 – No.11, September 2011.

[6] W. Luan, Ch. Zhu, Bo Su, and Ch. Pei "An Improved Routing Algorithm on LEACH by Combining Node Degree and Residual Energy for WSNs" Xidian University, China .

[7] Narottam Chand "Cooperative Data Caching in WSN" World Academy of Science, Engineering and Technology Vol:6 2012-03-27.

[8]P.Th. Joy and K. P. Jacob "Cache Replacement Policies for Cooperative Caching in Mobile Ad hoc Networks" Dept. of Computer Science, Cochin University of Science and Technology ,Kochi, Kerala, India.

[9] Narottam Chand "Energy Efficient Cooperative Caching in Wireless Multimedia Sensor Networks" advances in information science and applications - Volume II.

[10] V. Kochher and R. K. Tyagi "A Review of Enhanced Cluster Based Routing Protocol for Mobile Nodes in Wireless Sensor Network" Advance in Electronic and Electric Engineering. ISSN 2231-1297, Volume 4, Number 6 (2014), pp. 629-636 .

[11] M. F. Khan, E. Felemban, S. Qaisar" Performance Analysis on Packet Delivery Ratio and End-to-End Delay of Different Network Topologies in Wireless Sensor Networks (WSNs)" 2013 IEEE 9th International Conference on Mobile Ad-hoc and Sensor Networks.

[12] Narottam Chand, R.C. Joshi and Manoj Misra,"Efficient Cooperative Caching in Ad Hoc Networks", IEEE Transactions, pp.1-8, August 2006.