CHAPTER 2

BACKGROUND

Data replication schemes that are projected in distributed systems are to address the issue of data availability and the performance. Availability of data refers to persistence of data at all times even in the failure of a few replicas. Performance refers to reduced latency and load balance by distributing the replica to several servers that are nearer to clients. Nevertheless, these conventional approaches will not be appropriate for WADNET since they have to address some additional constraints that arise in the ad hoc network environment.

To develop a replication mechanism in ad hoc networks, three significant issues to be considered are network partitioning, energy consumption and scalability. However, replication mechanisms proposed in the literature both for MANET and WSN addresses any one of the issues or combination of any two issues stated above. Hence, the feasibility of addressing all the three issues is yet to analyze and implementing the mechanisms in real network deployment still remains an open issue. This chapter confers meticulously on the subject of the related works that exist for data replication and discuss the demerits and restrictions in existing approaches thereby mentioning the need for an enhanced replication model that can circumvent the drawbacks of existing approaches.

2.1 RELATED WORKS

We have broadly classified the replication schemes in WADNET as:

- Ad hoc-unaware Data Replication Schemes in MANET.
• Ad hoc-aware Data Replication Schemes in MANET.

• Other Data Replication Techniques in MANET.

• Ad hoc-unaware Data Centric Storage in WSN.

• Ad hoc-aware Multi-Replication Data Centric Storage in WSN.

2.1.1 Ad hoc-unaware Data Replication Protocol in MANET

The methods proposed in this category give importance in increasing data availability and do not deal with the issues that are related to network partitioning, energy efficiency and scalability.

2.1.1.1 Flooding Based Replication Protocols

The very first approach to replicate data items in ad hoc networks is to flood the information in the network about the data items [48, 49]. This approach cannot be practicable as it cannot adapt to topology changes that occur in the network. Hence, the variant of flooding called as selective flooding [50, 51] was proposed to reduce the communication cost by forwarding the messages that contain information about data items only to a subset of neighbors. However in general, the flooding mechanisms are not appropriate for large scale dynamic WADNET.

2.1.1.2 Quorum Based Replication Protocols

Data management in MANET is considered as a major issue, therefore, several approaches were proposed by researchers that were focused only on the data management issues in MANET. In particular, more interest was shown on quorum based replication protocols [52–58]. The concept of quorum systems is considered to be a renowned technique implemented in distributed systems. A quorum is defined as a subset of servers and classified as write quorum and read quorum. Quorums are either fixed or determined randomly.

The significant trade-off of this mechanism is the larger the quorum size higher the update cost and larger the number of nodes in the intersection of quorums in ad hoc networks. Conversely, making use of the quorum system has several advantages: First,
due to the fact of intersection of two quorums, there exists one replica in read quorum with latest version. Second, by selecting a subset of servers and performing updates and query requests, energy consumption of other mobile hosts is curtailed. Third, the load of the replica server is well balanced by allocating the replica evenly in the network.

Many of the quorum based replication protocols proposed in ad hoc networks select the topology based routing protocols to update and query data packets. Inspite of its advantages, the quorum based systems has some limitations. They incur high complexity due to topology changes, also, if updates and query occur frequently, establishing a route often becomes an overhead. In addition, there is no guarantee that the data is available at all times and cannot ensure that the accessed data is consistent at all times. This is due to the fact that frequent network partitioning and node failure due to energy depletion may lead to non-intersection of write quorums and read quorums.

### 2.1.1.3 Dynamic Adaptive Replica Allocation Algorithms

The work proposed in [59] discuss about a distributed dynamic adaptive replica allocation algorithm for MANET. In this scheme, communication cost becomes the most important factor which persuades the performance of data access in an ad hoc environment. The authors have proposed an algorithm known as adaptive replica allocation algorithm (ARAM) initially and extended ARAM (EARAM) . The algorithm APARA is executed periodically and independently. Replica decision is made by performing three tests namely Expansion-Test, Relinquish-Test and Switch-Test. The tests are executed as follows:

- **Expansion-Test:** For the neighbor $u$ of the replica node $s$ and $u$ is not in $F$, the condition of expansion is satisfied (which means that when a replica is expanded to $u$, the decrease of the access cost is greater than the increase of the update cost, thus the total communication cost declines), a replica is expanded to $u^*$ and $F' = F \in u^*$.

- **Relinquishment-Test:** If the number of update request received by replica node $s$ from other replicas is larger than that of the read and write requests received
by s from itself and those non-replica nodes, then s requires relinquishing the replica.

- **Switch-Test:** The switch test will allocate the replica to the neighbor node which receives more read and writes requests. Then \( u' \) becomes a replica node while s will not be a replica node anymore.

The limitation of the above mentioned algorithm is that it does not adhere to topology changes, hence an extended algorithm is proposed and is named as EARAM-SN (The EARAM Algorithm Based On The Stable Neighbor). This algorithm identifies the relatively stable neighbors of replica nodes in a distributed way and spread out replicas only to stable neighbors. Besides, in EARAM-SN algorithm the access requests are collected only from stable neighbors while expanding or relinquishing the replica. Neighborhood stability is determined by finding the distance between two nodes with the aid of GPS.

### 2.1.1.4 Effective Replica Allocation Algorithm in MANET

Hara et.al [60] has proposed three allocation schemes which diverge in emphasis that is set on access frequency and network topology. The allocation schemes are named as SAF (Static Access Frequency), DAFN (Dynamic Access Frequency and Neighborhood) and DCG (Dynamic Connectivity based Grouping). These schemes are described as follows:

- **Static Access frequency (SAF):** In the SAF method, the objective is to allocate replica in the mobile host in descending order of access frequency. The mobile hosts do not need to exchange information with each other for replica allocation. No duplication removal takes place as they are not aware of existence of replica with their neighbor. The efficiency of this method is the allocation of replica takes place with low overhead and low traffic. On the other hand, since each mobile host allocates replicas based on only the access frequencies to data items, mobile hosts with the same access characteristics allocate the same replicas. Therefore,
the SAF method gives low data accessibility when many mobile hosts have the same or similar access characteristics.

- **Dynamic Access Frequency Neighborhood (DAFN):** The objective of this method is to eliminate duplicate of replicas that exist between the connected mobile hosts. At a relocation period, each mobile host broadcasts its node identifier and information on access frequencies to data items. On completion of broadcasts, from the received host identifiers, every host will come to know its connected mobile hosts. Each mobile host at the beginning determines the allocation of replicas based on the SAF method. In each set of mobile hosts which are connected to each other, when there is duplication of a data item (original or replica) between two neighboring mobile hosts, and if one of them is the original, the host which holds the replica of data changes it to replica of another replica. If both nodes hold a replica of the data, the host whose access frequency of the data item is lower than the other one replaces the existing replica with another replica of data. When changing the replica, among data items whose replicas are not allocated on either of the two hosts, a new data item is selected for replication where the access frequency of this item is the highest among the possible items. Compared to SAF, data accessibility is expected to be higher in DAFN method, but does not completely get rid of replica duplication among neighboring hosts because it performs the elimination process only for once by scanning the network based on the breadth first search.

- **Dynamic Connectivity Based Grouping (DCG):** The DCG method distributes replicas in larger groups of mobile hosts than the DAFN method that shares replicas among neighboring hosts. In order to share replicas successfully, each group should be stable, i.e., the group should not be easily separated due to changes of network topology. From this perspective, the DCG method creates groups of mobile hosts that are bi-connected components in a network. A bi-connected component is defined as a maximum partial graph which remains connected (not divided) even if a random node in the graph is deleted. By grouping mobile hosts as a bi-connected component, the group is said to be more stable even if one mobile host disappears from the network or one link
is disconnected in the group, and thus it is considered that the group has high stability.

The DCG method is executed at every relocation period. In each group, an access frequency of the group for each data item is calculated as a summation of access frequencies of mobile hosts in the group to the item. This calculation is done by the mobile host with the lowest identifier. Based on the order of the access frequencies of the group, replicas of data items are allocated to all mobile hosts in the group until the memory space of the host becomes full. The replicas of data items which are held as originals by mobile hosts in the group are not allocated. Each replica is allocated at a mobile host whose access frequency of the data item is the highest among hosts and has free memory space to create it.

2.1.1.5 Two Phase Replication Algorithm

A distributed and decentralized algorithm for dynamic data replication has been proposed in [61] named as the two phase replication approach (TPRA). The aim of this replication is to ensure high data availability. Two algorithms were proposed and termed as Primary replication algorithm and Dynamic replication algorithm. In primary replication, the node that wish to replicate its data, distributes its data to a node that are three hops away and the new replica holder further distributes the data to a node that is three hops away from it. This process continues till no nodes are found in the network.

In dynamic replication, the replica is redistributed based on a parameter called as access rate. If a node accesses a data frequently, the replica of the data is redistributed to it in order to save the energy and reduces the traffic. This node then notifies the other nodes in the range that it has been designated as a new data server. If the access rate of a node is low then the client have to access data from the server that are $k > 1$ hops away.

Although this method distributes the replica uniformly, the replica distribution is based only on the location of primary server which could not be a fair approach in a situation where a client may take several hops to reach a server. The distance estimation between the servers is three hops away which is considered to be a weak assumption.
and cannot be a good estimation for large scale or dense networks because this may lead to increase the number of replicas. Moreover, in dynamic replication, redistribution of replica is based only on the access rate of the data from a client and the other issues that exhibit the dynamic behavior of a node are not considered.

2.1.1.6 Balancing the Trade-off between Query Delay and Data Availability

The work proposed by [62], discusses new data replication techniques to balance the trade-offs between data availability and query delay. They have proposed four replication schemes named as, the greedy data replication scheme, one to one optimization (OTOO), Reliable Neighbor scheme (RN) and Reliable Grouping scheme (RG). In greedy scheme, each node picks the data item with large access frequency and small data size. This scheme does not consider the cooperation between the neighboring nodes and therefore its performance may be limited.

In the OTOO scheme, each mobile node only cooperates with at most one neighbor and calculates the combined access frequency to decide which data to replicate. If the probability of link failure between two nodes is less and if the neighbor contains the frequently accessed data, the node is less likely to replicate this data. Otherwise it replicates the data locally.

The next scheme is the RN scheme which further increases the cooperation to replicate data for a number of neighboring nodes. Each node checks its neighbor reliability by calculating the link failure probability. If the probability of link failure is less, then the node finds the neighbor data-interest and replicates them.

To further increase the cooperation of nodes, RG scheme has been proposed, in which replicas are shared in large groups. This scheme selects the most suitable data items and places them in the best node. The node selection is based on the function that considers access delay between the query node and the nearest replication node. This paper focuses on allocation of frequently accessed data near to the clients, but has not considered the stability of a node to place the replica.
2.1.2 Ad hoc-aware Data Replication in MANET

The protocols that are mobility aware or power aware are called ad hoc aware data replication protocols. The following sections discuss various mechanisms that fall under these two categories:

2.1.2.1 Partition-aware Data Replication in MANET

Partition-aware protocols can predict the occurrence of network partitioning and replicate data items beforehand.

2.1.2.1.1 Partition Detection in Mobile Ad hoc Networks

The replication mechanism discussed in [63,64] define a metric to evaluate the link quality between two nodes and find a reliable disjoint path between the client and the server. If no disjoint path occurs it is concluded that partition will occur and service replication will be triggered. To find a reliable path the authors have proposed a theoretical approach in calculating the reliability metric denoted as $R_k(v,w)$. Here $v$ is assumed to be a client and $w$ is assumed to be the server. The metric first determines the set of $k$-sub-optimal between $v$ and $w$ and denoted as $SOP_k(v,w)$ with $k > 0$. A path $P$ is $k$-sub-optimal path if and only if

- Path $P$ is free for loops
- Number of hops in $P$ denoted as $|p| < d(v,w) + k$ where $d(v,w)$ is the least possible distance between $v$ and $w$.

Then subset of $SOP(v,w)$ that contained disjoint paths is found and denoted as $DSP_k(v,w)$. Then the robustness of the path between the nodes $v$ and $w$ is given by:

$$R_k(v,w) = \max_{C \in DSP_k(v,w)} \left\{ \sum_{p \in C} \frac{1}{|p|} \right\}. \quad (2.1)$$

To make this approach practical, three broadcast based protocols were proposed to compute the robustness of the link between the client and the server, These approaches are named as efficient broadcast based protocol (mode 0), stormy broadcast based protocol (mode 1) and Directed response based protocol. The operation of these protocols are explained below:
• **Efficient Broadcast Based Protocol (mode 0)**

1. $v$ sends a link evaluation request message to $w$ using a routed message;

2. $w$ replies with a link evaluation reply message using a flooding protocol with a TTL of $d(v, w) + k$ ($k$ depends on the $DSP_k(v, w)$ set);

3. Each node forwarding the messages appends its ID in the message in order to construct the path used by the message;

4. $v$ receives the responses and check it with the paths it has already stored. If this path is disjoint of all the others, the node stores it. Otherwise, the path is discarded (as this path comes later to the node, it is supposed to be longer);

5. When a given time out has elapsed, $v$ can evaluate $R_k(v, w)$.

• **Stormy Broadcast Based Protocol (mode 1)**

1. $v$ sends a link evaluation request message to $w$ using an efficient broadcast protocol. Every node $u$ receiving the packet stores $d(v, u)$;

2. $w$ sends a message containing a set of sequence numbers to all its neighbors. When receiving it, a node randomly selects a sequence number in the set and sends a link evaluation request broadcast;

3. Every node $u$ receiving this packet forwards it if $|p(w, u)| + d(u, v) \leq d(v, w) + k$ and if it has not forwarded a packet with the same sequence number before;

4. Each node forwarding the messages appends its ID in the message in order to construct the path used by the message;

5. $v$ receives the responses and stores only disjoint paths (by the same way as mode 0);

6. When a given time has elapsed, $v$ can evaluate $R_k(v, w)$.

• **Directed response based protocol (mode 2)**

This protocol is proposed to prevent the downside of the previous protocols where the entire network is flooded with packet messages. By simulating this protocol
the authors have proved that this protocol performs better than the other two protocols.

These techniques discussed above predicted the network partition by computing the robustness path between the server and the client. However, a lot of computation is incurred in finding the reliable path between the client and the server. Moreover, this mechanism is not aware of how much energy is consumed to flood the query and get the reply from the server to compute the robustness link. Hence, this method cannot be attractive in power constraint networks and for time critical applications.

2.1.2.1.2 Stability of Radio Links

Hara et al. [65] have introduced the concept of the stability of radio links to allocate replicas on connected mobile host in the whole network. The initial distribution of replica takes place by using a static access frequency method (SAF). After applying SAF, replicas of every pair of neighboring nodes are compared to eliminate duplication. The authors have proposed three more methods known as DAFSN-S1, DAFSN-S2 and DCG-S1.

The objective of these methods is to eliminate the duplicates of replicas between two mobile nodes only when the stability of link between these nodes is good. The assumption made with these techniques is that the speed and the direction of movement of all mobile hosts are known in advance. The time at which the two mobile host disconnects can be predictable by monitoring the movements. The authors have denoted the mobile host as $M_i$ and $M_j$ and the disconnection time as $T_{ij}$. The stability of wireless link between two nodes $M_i$ and $M_j$ at time $T$ is represented as $B_{ij}$ and if $T$ is relocation period then $B_{ij}$ is given by

$$B_{ij} = 1 \text{ when } T_{ij} \geq 1; \text{ Otherwise } B_{ij} = \frac{T_{ij}}{T}. \quad (2.2)$$

In DAFSN-S1, for each pair of nodes $M_i$ and $M_j$, the link stability $B_{ij}$ is computed and compared with a threshold. If the link stability $B_{ij}$ is greater than a threshold, the link between the nodes is strong and replicas are compared for each pair of nodes. If duplicate exists and if the data is not original it is replaced by new data whose access
frequency is high compared to other data items. Otherwise if the stability of a link is not strong the node pair is ignored.

In the second method, DAFN S2, a different strategy is proposed to estimate the stability of the link in order to remove the duplication of replicas. An additional parameter $p$ termed as access frequency of data items is included. The probability that this host will access the data even if it is disconnected from the other host is defined as $(1 - B_{ij}) \times p$. If it is below a threshold value $x$, the duplicates will not be eliminated from the hosts. Otherwise, the replica with lower value of the term $(1 - B_{ij}) \times p$ will be removed from the mobile host as it implies that lower the value, the less the probability to access the data item.

The third technique known as DCG S1 considers two mobile hosts to be connected only if it is greater than the threshold. Different bi-connected mobile hosts will form a group and the duplicate replica in the group is eliminated.

These three techniques are aware of network partition and the decision to replicate data items are made at each relocation period. Despite their efficiency in allocating replica, more computation is involved in eliminating the redundancy of replicas which may cause unnecessary delay in accessing the data and consume more energy in removing the replicas.

2.1.2.1.3 Exploring Group Mobility for Replication in a Mobile Environment

The authors in [66] have addressed the problem of replica allocation by exploiting group mobility. The group mobility model used in this scheme is named as reference point group mobility model (RPGM). To avoid flooding scheme, DRAM (Decentralized Replica Allocation with Group Mobility) uses a bottom-up approach without requiring the global network connectivity. Hence DRAM exchanges the motion behavior with selected neighbors. This information exchange is restricted using a predetermined parameter. Then the cluster of mobile nodes is formed with similar motion behavior and data items are allocated. The DRAM scheme is executed periodically to adapt to the network connectivity and the time period is known as relocation period. The major task for each node in relocation period is divided into two phases. Allocation
unit construction phase and replica allocation phase

**• Allocation Unit Construction Phase**

The term allocation unit is defined as set of the mobile data that store shared data and care is taken so that no duplicate data exist until all the shared data is stored in allocation unit. To form an allocation unit each mobile node broadcast a *info* message to its neighbor. If the distance between these nodes is within the TTL specified in the *info* message, the neighbor broadcasts this *info* message to its neighbor, else discards the *info* message. Therefore, only the neighbor nodes that are in regions of the smaller distances will receive the *info* message and it is referred as broadcast zone. The existence of TTL is to avoid blind flooding of messages.

The node with the lowest id will be elected as ZONE_MASTER and other nodes become as ZONE_MEMBERs in broadcast zone. Each ZONE_MASTER then clusters its member that has a similar mobility pattern. To maintain the cluster during the relocation period, each member sends a status message to its master. If the member and master are disconnected then member finds a new master and associates with it. Similarly, if the master does not receive a status message from the member, it removes those members from the list.

**• Replica allocation phase**

The replica allocation data items are allocated to different mobile hosts based on the access frequency and the allocation units that are formed in the previous phase. The algorithm works as follows. The allocation weight of the data item in the allocation unit $C_x$ for timestamp $k$ is the expected number of data access from $C_x$ before the next update of $D_j$ at timestamp $U_j - k$ and can be expressed as follows:

$$w^x_j = f_j^x * (U_j - k) \text{ where } f_j^x = \sum_{M_i \in C_x} f_{ij}.$$  \hspace{1cm} (2.3)

All data items are allocated in $C_x$ according to their allocation weights in $C_x$ in descending order. Let the allocation candidate set of $D_j$ in $C_x$ be a set of mobile hosts in $C_x$ where each mobile host in the set is of available storage and with at
least one path to the owner of $D_j$. If the candidate set of $D_j$ in $C_x$ is not empty, $D_j$ will be allocated to $M_i$, where $f_{ij}$ known as access frequency should be largest among all mobile hosts in the allocation candidate set of $D_j$. Otherwise, $D_j$ will be skipped. The allocation process completes if all mobile hosts in $C_x$ are full.

### 2.1.2.1.4 Data Replication Approaches Satisfying Time Constraints

In the paper proposed by [67], the goal of the data replication technique is to address the issue of data access delay, i.e. to have a minimum number of data servers and access the data in the minimum number of hops. They have introduced three approaches for replicating data named as a centralized approach (CEN), distributed approach with no status information, distributed approach with status information.

In CEN, the algorithm is run by a coordinator to identify the data server in the network. A coordinator can be any host which determines the number of hosts covered by each host in “$k$” hops using controlled flooding message. Any host who covers a maximum number of hosts will be the data server and thereafter the hosts who are in its coverage will not participate in the data server selection. The remaining hosts are arranged in the decreasing order based on the number of hosts they cover. The data server selection process continues until there is no remaining host. This method of allocation reduces the access delay since the client can access the data in at most “$k$” hops. However, it is unsuitable for a dynamic behavior because the data accessibility will become low if the server moves away from its client.

To resolve this, the authors have proposed two other approaches that take care of the dynamic behavior in the network. First method, consist of two phases: Initialization phase and Updating phase. In the initialization phase, each host generates a random number and if the random number is less than a threshold value the host can hold the data items; otherwise they remain as a regular host. In updating phase, if each host moves away from a communication range, they should get the data from the data server who are more than “$k$” hops away and announces itself as data server to the entire host which are in “$k$” hops away from it. The next method is slightly modified to avoid redundancy of servers and maintain a minimum number of servers. To accomplish this, each data server within the range can exchange their member information to other
server. If data servers have same set of members, one data server can give up its role and can become a regular host.

However, all these approaches discussed in this paper have no guarantee on the degree of replicas maintained. Lots of messages are flooded to identify the data servers and to avoid redundancy of servers. Due to these overheads, these approaches cannot perform well in scalable networks.

### 2.1.2.1.5 Partition Prediction Algorithm for Service Replication

Drab et.al [68] has proposed a prediction algorithm that belongs to the pull based type and the network partition is spotted using a TORA [69] routing protocol. The key contributions of this paper are to envisage the occurrence of network partitioning, estimate the apt time to activate the service replication process and replica deployment scheme that decide which node will hold the data item in a particular partition. The TORA protocol classifies links into strong and weak based links based on their residual life time. Any node if it loses its current strong link to reach the server, it invokes TORA to find at least one alternate routing path to reach the server. If no stable route is found, the node concludes that the server is about to be inaccessible. In this state, the node downloads the service from the server to ensure service availability.

To ensure that the service is replicated in one node at future partition, the threshold of a residual lifetime should be sufficient enough to predict the partitioning and to download the copy of the service. The advantage of this method is, it forecasts the future changes in topology and minimize the resource wastage by allowing one node to provide a service in network partition. Despite its advantages, the mechanism involves more computational time identifying the stability of links between the node and the server and in addition it does not address in how many hops the data can be accessed by clients. Hence, this type of mechanism will not be suitable for time critical applications where query latency is a major concern.

### 2.1.2.1.6 Integrated Data Lookup and Replication Scheme

Chen et al. [70, 71] have proposed a data lookup service, in which data availability information is exchanged between nodes of a connected group. Each node periodically broadcasts an advertising
message called ad for each member of the group. The ad message contains a sequence number, node’s capability (free space, remaining power, and processor utilization) and the available data items at the node. Each node, upon receiving the ad message, updates its local Data lookup table. When duplicated data is found in the receiving node and if its address is lower than the advertised node’s address, the local copy of the data is deleted. Each node of the group uses location information provided by GPS receiver connected to a reference node and location information received by other nodes of the group to calculate the movement of each member in the group. This information helps to predict group partitioning and therefore, replicate data items to nodes of the future-separate- partitions.

2.1.2.2 Power-aware Data replication in MANET

The schemes discussed below aims to reduce the power consumption of nodes to increase the lifetime of nodes.

2.1.2.2.1 Weighted Expected Access and Battery

Hara et. al [72] have proposed a replica allocation method, called Weighted Expected Access and Battery (WEAB). The mechanism aims to balance the power consumption among mobile nodes. In the WEAB method, when $M_i$ wants to access a data item $D_{\text{new}}$, which is not available in $M_i$, it performs some operations to decide whether to replicate the new data item in it or not. The operations are as follows:

1. $M_i$ replicates $D_{\text{new}}$ if it has free memory space to create the replica. Otherwise, $M_i$ will flood mobile nodes within $h \geq 1$ hops with a data information query packet. This query packet includes the list of data identifiers of data items held by $M_i$ and $D_{\text{new}}$.

2. When $M_k$ receives the query packet, it transmits a data information reply packet to $M_i$. The reply packet consists of access frequencies from $M_k$ to data items included in the query packet and flags that signify whether $M_k$ holds these data items or not.
3. If $M_i$ receives reply packets, it calculates the following criterion $\Delta D_{i,j \rightarrow \text{new}}$ for each data item held by $M_i$. In Equation 2.4,

$$
\Delta D_{i,j \rightarrow \text{new}} = \alpha (a_{i,\text{new}} - a_{i,j}) + \frac{P_i}{P_{\text{init}}} \times \beta \left( \frac{A_{i,\text{new}}}{C_{i,\text{new}}} + 1 - \frac{A_{i,j}}{C_{i,j}} \right).
$$

(2.4)

$a_{i,j}$ denotes the access frequency from $M_i$ to $D_j$, $A_{i,j}$ denotes the sum of the access frequencies to $D_j$ from mobile nodes within $h$ hops from $M_i$ excluding those holding $D_j$ and $C_{i,j}$ denotes the number of $D_j$ held by $M_i$ and mobile nodes within $h$ hops. $P_i$ and $P_{\text{init}}$ denote the remaining and initial amounts of $M_i$ battery, respectively. The $\alpha$ and $\beta$ are known as predefined weights.

4. $M_i$ selects $D_j$ among its own data items whose $\Delta D_{i,j \rightarrow \text{new}}$ has the positive maximum value and replaces $D_j$ with $D_n$. If $C_{i,j} = 1$, i.e., $M_i$ is the only node which holds $D_j$, it does not discard $D_j$. If $C_{i,\text{new}} = 0$, i.e., No mobile nodes hold $D_{\text{new}}$, hence $M_i$ replicates $D_n$.

This method balances the power consumption among nodes because data items with high access frequencies are replicated on a large number of nodes. Furthermore, Equation 2.4 prevents nodes with the little remaining amount of batteries from being accessed for data.

2.1.2.2 PADARA  Wang et. al [73] have proposed a Power-Aware Dynamic Adaptive Replica Allocation Algorithm (PADARA) that alters the replica allocation scheme frequently in order to reduce the power consumption and thus extends the network lifetime of a node. The replica allocation scheme for an object $O$ is the set of nodes at which $O$ is replicated. The read requests are served by a chosen replica node and the write requests are propagated from a chosen replica node to all other replicas. PADARA is performed from time to time and autonomously in each replica node.

For each object $O$ held by $m$ and a neighbor $u$ such that $u$ does not hold $O$, node $m$ executes three tests: Replica Expansion, Replica Switch and Replica Contraction. In expansion test, if object $O$ is accessed frequently, $O$ is replicated in $u$ in expansion test. The read request to the data will decrease the power consumption whereas propagation of write requests to replica nodes increases it. The replica switch test
performs replacement of replica from \( m \) to \( u \). Hence read request coming to \( m \) are forwarded to \( u \) thus increasing power consumption. If \( u \) lie closer to many replica nodes power consumption of write requests propagation is reduced. In contraction test, deletion of replica in \( m \) increases power consumption of read requests and decreases power consumption of writing requests. These tests are performed regularly and hence power consumption is conserved which indirectly increases the network lifetime.

2.1.3 Other Data Replication Techniques in MANET

2.1.3.1 Applicability of Eigen Vector Centrality for Replica Allocation

In a recent paper [74], authors have applied the Eigenvector centrality (EVC) principle to determine the best node for replication of shared data. Their goal is to find whether EVC calculations can give good measure for determining the host nodes to replicate data items. The authors have calculated EVC for three types of connectivity matrices - Weighted Connected Matrix, Unweighted Connected Matrix and Weighted Normalized Connectivity matrix. They have proved that using different types of connectivity matrix to calculate EVC does not make much difference in the performance of the network. Also, they have confirmed that choosing a group of higher EVC nodes as replication points is not a good measure because according to the definition of the EVC, if a node has high EVC then the neighbor node also has high EVC. Therefore, the replication points chosen are close to each other at most of the times and hence replicas are not distributed evenly which eventually decreases the data accessibility for large scale networks.

2.1.3.2 Handling Selfishness in Replica Allocation

The work proposed in [75] addresses the problem of selfishness in the context of replica allocation in MANET, i.e., Selfish nodes may not share its own memory space to store replica for the benefit of other nodes. The proposed strategy consists of three parts: 1) Detecting selfish nodes, 2) Building the self centered friendship SCF-tree and 3) allocating replica. Each node in the network detects the selfish nodes based on credit risk scores and then makes its own topology (partial) and builds its own SCF-tree by excluding selfish nodes and finally allocates replica in a distributed manner. Despite
the fact that the authors have proposed replica allocation algorithm in a new dimension, the algorithm incurs a lot of computation in detecting selfish nodes and constructing SCF-tree. Such type of replication is not applicable for time critical applications.

2.1.4 Ad hoc-unaware Data Centric Storage in WSN

The methods proposed under this category focus in selecting single point as storage node or zone head to store the sensed data. Since these methods do not deal with dynamic changes that occur in the network, they are known as adhoc-unaware schemes.

2.1.4.1 Similarity Data storage

The objective of Similarity Data Storage (SDS) [76] is to propose a proficient spatial-temporal similarity search scheme for both static and dynamic WSN. The three major challenges identified in this scheme are, efficient data aggregation and query, similarity search for multi-attribute data and spatial temporal search. Therefore, SDS is considered to be an important approach that utilizes spatial-temporal and similarity search functionalities and aims to reduce overhead, energy consumption and search latency.

In SDS, a large-scale WSN field is deployed in a rectangular field. The entire field is divided into small rectangular zones. Each zone has a dedicated sensor node named as zone head. The assumption made in this scheme is that each node in the network is configured with three information elements: (1) Number of zones divided horizontally as $n_x$ and vertically as $n_y$, (2) the zone ID which are assigned within the zone sequentially from left to right and (3) Geographical location. A head node in a zone is responsible for communication with other zones and takes care of aggregation and processing queries. All other nodes inside a zone are connected to the head node to store and retrieve data.

2.1.4.2 Dynamic Load Balancing Storage Scheme

In Dynamic Load Balancing (DLB), [77], authors have pointed out the unbalanced distribution of data among sensors as one of the key constraint for most of the DCS techniques. To deal with this issue, they have proposed a grid-based DLB approach that
depends on two schemes: (1) A cover-up scheme to deal with the problem of a storage node whose memory space is depleted and (2) multi-threshold levels to achieve load balancing in each grid so that all nodes get load balanced. The entire network is divided into a grid with cells of the same size in such a way that all the nodes inside a cell are within one hop distance. Each grid is numbered with positive coordinates \((X, Y)\) called grid IDs. A sensor node can calculate its grid ID \((X, Y)\) using the following equation:

\[
X = \left\lfloor \frac{(X_1 - X_0)}{d} \right\rfloor \quad \text{and} \quad Y = \left\lfloor \frac{(Y_1 - Y_0)}{d} \right\rfloor.
\] (2.5)

Additionally each node has a virtual grid ID and virtual co-ordinates that are initially equal to the actual grid ID and co-ordinates. Initially, each node broadcasts a message within its grid by limiting broadcast protocol to exchange the information to build a 'Grid Node' table. A producer node uses the hash function on the event type to map the event type into a grid and transform the event type into a grid ID using the above equation. The center of the grid is called a grid point and it acts as a data centric storage. The node, after detecting an event, sends a Put packet to the grid ID and uses GPSR [78] to forward this packet to the node closest to the grid point which inturn transmit to grid point.

2.1.5 Ad hoc-aware Multi-Data Replication in WSN

The schemes proposed in this category select multiple replication points to store the data in order to balance the energy consumption. Based on the link quality or residual energy change over of replication nodes take place over time.

2.1.5.1 Sense Swarm

The authors have proposed a scheme [79] known as Sense Swarm (perimeter based), a framework for acquisition and storage of spatial temporal events in mobile sensor networks. Two categories of nodes are introduced in this framework and named as perimeter node and core nodes. The perimeter nodes are used to acquire data and the core node acts as replicas. The contribution of this paper is to divide the proposed work in two phases. The first phase is to partition the network into the perimeter and core nodes using two algorithms known as centralized perimeter algorithm and perimeter algorithm. The second phase of the proposal is to acquire information from
the environment and then replicate to neighboring nodes. The replica nodes are selected based on the voting algorithm.

As a first step, the algorithm discovers an adequate number of neighbors and defines read quorum and write quorum called as \((r, w)\) combinations. The next step of the algorithm is to eliminate the redundant \((r, w)\) combination. The final step is to further prune the \((r, w)\) combination by ranking them. The \((r, w)\) combination with the highest score will be selected to replicate the data item. The focus of this paper is to reduce energy by maintaining high data availability. However the proposed work fails to choose the stable nodes to replicate data. As a result the failure rate of nodes will be considerably higher. Furthermore this scheme incurs a high computational cost in determining the \((r, w)\) combinations and pruning the \((r, w)\) combination to select the replica nodes.

2.1.5.2 Resilient Data Centric Storage Scheme (RDCS)

The method proposed in [80] is an extension of data centric storage to minimize query-retrieval traffic, increases data availability, ensuring event information is not lost even in failure of multiple nodes. The architecture of resilient data centric storage (RDCS) is divided into monitor nodes, replica nodes and normal nodes. The monitor nodes on receiving the query send it to the nearest replica node using the map information list. The replica node acts as a storage node and replies to queries of a particular event type issued by sink nodes. Normal nodes sense or forward to replica nodes.

To balance the energy and increase the lifetime, nodes switch from one mode to another mode. The replica node may degrade to monitor node if its activity coefficient goes below the lower threshold. Similarly, the monitor node upgrades to replica node if its activity coefficient at a particular instant is greater than the upper threshold. The activity coefficient is a composite of residual energy and query traffic.

Despite the advantages, the mechanism aims to store data only near to sensing nodes, computational overhead are incurred in identifying monitor nodes and replica nodes. Moreover, every query will traverse the monitor node and then to the replica node to receive the reply which is certainly an additional overhead to access the data.
2.1.5.3 DIDS scheme

The focus of the paper discussed in [81] is to reduce the average query delay and to avoid unnecessary transfer of sensing data and floating control messages to the whole network. Hence, the authors have proposed an important structure called a $k-hop$ connected dominating set to form storage and index node sets and named the mechanism as DIDS scheme. The given network is divided into three layers. The bottom layer consists of sensing nodes to generate raw data. The middle layer consists of storage nodes. The storage nodes in these layers are identified using the Weight function.

The parameters that are included in Weight function are level, node degree, energy level and the node-id. Each node in the network finds its $s-hop$ neighbors and calculates their weight. The neighbor with maximum weight will become the dominator. The sensed data are stored at dominator and the queries are routed to top layer first which consist of index nodes and from there it goes to appropriate storage nodes. The index node contains index information i.e. Location information of storage nodes. To conserve energy, standby replica nodes are selected when the residual energy of existing replica node goes below a threshold. The new data collected will be stored in standby nodes. Regardless of its advantages, there exist some drawbacks in this method. Additional overhead is incurred in maintaining index nodes and the query has to traverse not less than $s+i$ hops to access the data. Besides, replication points do not change over time, instead the changeover occurs only when the energy goes below a threshold. Eventually, this may create routing holes and thus, affecting the routing of the whole network.

2.1.5.4 RANDOM

The authors have proposed a generic replication framework [82] known as RANDOM which uses a set of randomly located replicas that can change dynamically over time. To find the optimal number of replicas, they have proposed and validated a simple model by considering the ratio of consumption intensity and the production intensity. If the production intensity is high, the number of replicas will be reduced. On the other hand, if consumption intensity is high, the number of replicas will increase.
Furthermore, to balance the energy expenditure over time, replication sets will change dynamically. The new set of replication points is determined using a hash function that consists of application, an epoch, a shared time identifier employed to change replica over time and the nodes that are closest to hashed spatial locations. Although the method seems to be simple, flexible and effective in terms of reduction of network traffic, it fails to replicate data in stable nodes by considering the link quality and energy level. Hence, changeover of replication points may occur frequently if the consumer rate increases with production rate.

2.1.5.5 Tug of War

In Tug of War [83], three main contributions were discussed. The first contribution is to provide an analytical model that computes the optimal value of the network depth \( d \) in order to minimize the network traffic. The second contribution is that when the production rate is greater than the consumer rate, the producers store the information to closest replica. Therefore, the consumer retrieves information from all replicas for a particular event type. This type of storage is known as write-one-query-all storage. Conversely, if the consumer rate is greater than the production rate, still the producers forward the data to closest replicating node from which it is copied to all replica nodes and termed as write-all-query-one. Third contribution is that, it proposes a routing mechanism called as combing routing to support grid topology. Though this method reduces the storage cost and query cost, it is restricted only to grid topology.

2.1.5.6 Grid Based Multi Replication Scheme

Ratnasamy et.al. [84, 85] has proposed a grid structured multi-replication system. They assume a square sensor net that is divided into \( 4d \) grid, where \( d \) is defined as network depth, original home node is established by using a hash function and \( 4d - 1 \) mirror replicas are allocated in the same relative position inside each cell of the grid. Replication structure is hierarchical where the producers store the event to closest replica and the consumer queries the top home node which in turn forwards the queries to low level replica recursively. The reply to query traverses the same path toward the sink. Therefore information retrieval based on a hierarchical routing depends on the routing protocol called as GPSR. Eventhough, the method tries to reduce storage cost,
it increases the query cost by making the query to traverse longer communication path and hot spots cannot be fully avoided as all queries still traverse through home nodes to acquire information

2.1.6 Common issues in existing approaches

Most of the data replication methods that exist in the literature for MANET has several issues. The first issue is, these methods fail to maintain minimum replica degree. The second issue is, extra computation is involved in removing duplicates. The third issue is, no strategies are followed to select the appropriate node to hold replica in a dynamic network, hence the number of reallocation of data increases. In addition, in few approaches, replicas are not placed close to the clients, instead, they are distributed randomly and therefore, data access latency is high, communication cost increases which eventually make the replication scheme not suitable for large scale networks.

Similarly, all the existing data replication approaches discussed in sensor networks mainly concentrate on two main issues, i.e., reducing the query traffic, balancing the network by increasing the number of replicas. Very less attention is paid in selecting the appropriate data storage or replica nodes. Moreover, these mechanisms select the replication points very close to the source in order to reduce storage cost. As scalability increases, the sink may traverse multiple hops to access the data and changeover of replication points also occurs frequently which poses an additional overhead. Hence, there is a need for proposing a replication mechanism which should take care of all these issues at a time and ensures that the performance of the network is not degraded.