MACHINE-TO-MACHINE SMART GRID SYNCHRONIZATION FOR SUSTAINABLE ENERGY

Rashid. A. Saeed¹, Mohammad². K. Hasan, Rania A. Mokhtar¹, Aisha A. Hashim²

¹Department of Electronics Engineering, Faculty of Engineering, Sudan University of Science and Technology (SUST), Khartoum, Sudan
²Department of Electrical and Computer Engineering, Faculty of Engineering, International Islamic University Malaysia (IIUM), Kuala Lumpur, Malaysia.

Machine-to-Machine (M2M) communication is a recently developed technology in data communication network which extends to indoor coverage for serving smart grid and intelligent grid network. Synchronization is one of the most significant technical challenges in M2M networks to guarantee an acceptable clock offset and frequency error (skew), which leads to severe interference between machines, where all are working in same frequency under a licensed spectrum. Global Positioning System (GPS) antenna cannot synchronize M2M networks because of lack of signal strength in indoor environment. High-precision crystal oscillators can solve the synchronization timing problem, but they are often too expensive for machine devices. In addition, machines can also interfere with each other if multiple units are in close proximity. In some cases, conflicting machines may simultaneously raise transmit power in order to improve signal quality, thereby creating some interference. In other words, when machine transmits at high power, it reduces the capacity of the mBS. Furthermore, in a flat M2M structured network using IEEE 1588 synchronization algorithm creates clock offset and skew which results inaccurate synchronization. However, in this paper a new concept of cluster-based time synchronization scheme is proposed in order to reduce clock offset and skew to achieve precise synchronization among neighboring machines. The results show that the proposed scheme is able to reduce the offset and skew significantly by the number of synchronization process. Therefore, it can be strongly recommend that proposed scheme achieved better synchronization accuracy than the existing schemes.

Keywords: Cluster; M2M Synchronization; M2M Network; Intra-cluster; Inter-cluster.

INTRODUCTION

Present day communication technology has witnessed a paradigm shift with the advent of machine to machine (M2M) or (or machine-type) communications that have immense potential. As a vehicle for wireless communication, M2M networks have become the domain
of interest for the industry and academia. Nowadays intense efforts are being undertaken for standardizing M2M smart grid networks and communication protocols by bodies like 3GPP, ETSI, IEEE, and IETF. Various services and smart grid applications which involve M2M networks and communications could be identified. Those include electric vehicular (EV) and intelligent power distribution design and home automation [1].

These services and applications have completely transformed the way communication systems are defined and believed to operate so far. Sustainable energy through M2M smart grid has opened mass markets and several business opportunities among the green energy fraternity. From the research pursued till date, it could be predicted that the commercialization of M2M smart grid networks, on a serious note, may increase the energy utilization efficiency four times greater than the conventional grid. Consequently, M2M smart grid communication can ensure high average revenue per user (ARPU) and good return of investment (ROI) for the energy companies [2].

This paper proposes maturity software model in planning of utilities in smart grid transformation; prioritize the tasks and measuring their progress at every stage where systems such as automated fault isolation and system restoration can be evaluated [3].

Machine-to-Machine (M2M) extends network coverage and delivers high-quality indoor mobile services as well as data with the better cellular network coverage by using third party public broadband network [4]. It has triggered the design and development of new structured cellular standards such as WiMAX (802.16e), the Third Generation Partnership Project’s (3GPP’s) High Speed Packet Access (HSPA) and LTE standards, and 3GPP2’s EVDO and UMB standards [5].

Figure 1 shows the general architecture of M2M network where the consumer installed wireless data access point inside homes, which backhauls data through an IP broadband gateway (DSL/cable/Ethernet/WiMAX) over the Internet to the cellular operator network. The IP broadband network is usually owned and managed by third party and not by the mobile operator.

![Simple M2M architecture](image)

**Figure 1: Simple M2M architecture**

M2M networks uses the backhaul connection to access the clock of the time servers and the IEEE 1588 clock server in order to update the clock of the network. However, there are problems with this process because in this way M2M synchronization can suffer delays resulting from the varying delays introduced by the backhaul which results different packets will take different amounts of time as a result of the nature of packet networks [6].
As a result, synchronization is considered cornerstone for proper working for M2M network. The problematic issue in M2M synchronization is that all the data and control traffics travel through IP broadband network, which is complicated the synchronization. In addition, the conventional algorithms and schemes has the clock offset and skew which degrades the synchronization accuracy and as well as throughput. Unsynchronized network may cause harm interferences and wrong handover dictions.

The time synchronization problem in M2M network generally involves two steps. Firstly, synchronizing the nodes in the network at a specific absolute time through adjusting the clock offset among the nodes. Secondly, clock’s skew which includes correcting the clock frequency relative to a certain standard frequency [7]. To address time synchronization problem in M2M network, in this paper, we proposed cluster-based time synchronization scheme in order to reduce the clock offsets, skew and increase the synchronization accuracy.

The performance of the clock synchronization in M2M network is based on two important metrics which are clock offset and clock skew.

a. Clock Offset

Clock offset is addressed as the clock differences of M2M nodes between all neighbor nodes. Four main reasons exist for which the nodes represent different times in their respective clocks which are transmission time, access time, propagation time, and receive time. The synchronization process at nodes can begin in different times [8].

b. Clock Skew

Clock skew is the imperfections in the quartz crystal due to several environmental conditions namely temperature (clock drift). The quartz crystals at each of these nodes can possibly run at a little different frequency, which results in the clock values to slowly diverge (clock skew) or the frequency of the clocks to transform variably over a certain period [9]. In actual fact, the ultimate reason behind the clock offset is the clock’s drifting away due to the effect of clock skew. The long-term reliability of synchronization is ensured through the adjustment of clock skew. Therefore, estimating the difference of clock frequencies between two nodes increases synchronization accuracy and guarantees long-term reliability.

CLUSTERING APPROACH

Clustering is a standard approach for achieving well-organized and scalable precise synchronization performance in M2M networks [6]. Also, Clustering is an essential mechanism that efficiently provides information for M2M nodes and improves the clock synchronization process [13, 17]. It is not only reduces the central coordination but also reduces the overhead which will increase the throughput for the overall network. Clustering can be based on criteria like clock time of M2M nodes, distance of each node, and network size. Above all, in order to achieve a better clustering performance, overhead should be minimized, allowing M2M nodes to join and leave without perturbing the membership of the cluster while preserving current cluster structure as much as possible [18].

Clustering in M2M network can be considered as the virtual partitioning of random nodes in the flat structure or distributed network structure. Such nodes are regarded as neighbors when all neighboring nodes are located within their transmission range and set up a bidirectional link between them.
Clustering scheme can be categorized into two parts which are mainly Cluster Head (CH) election and cluster formation. Typical algorithms for clustering in the flat structure or distributed network structure are known as one-hop clustering and multi-hop clustering algorithms [17]. In the one-hop clustering, every member node is at most 1-hop distance away from a central node which is addressed as CH. Thus, all member nodes remain at most two hops distance away from each other within a cluster category. On the contrary, in multi-hop clustering [13], the management of neighboring nodes to the CH is performed by allowing the nodes to be presented at most \(d\)-hop distance away from each other to form a cluster.

A typical M2M network structure consists of flat and hierarchical structures as shown in Figure 2.

![Figure 2: Flat structured and hierarchical structure for M2M network [17]](image)

A group of clustering algorithms have been developed in the wireless sensor network which emphasizes some specific parameters such as speed and direction, mobility, energy, position, and the number of neighbors for CH selection [19].

These algorithms are advantageous in some cases. However, some negative aspects are found as a high computational overhead to execute the clustering algorithm. The highest-degree algorithm is referred as connectivity-based algorithm. This is basically founded on the degree of nodes which can be assumed to be the number of neighbors for a given node [20]. M2M nodes broadcast their Identifier (ID) only when the election procedure is required. In line with the number of received IDs, each node itself computes its degree and CH is selected on the basis of maximum degree. However, the main shortcomings of this algorithm are found with the frequent variation of degree of a node. When the degree of a node varies, then the CHs do not act as CHs for a lengthier period. Additionally, when the number of nodes in a cluster keeps increasing, then reduced throughput is observed which ultimately degrades system performance.

A Fuzzy Relevance-based Cluster head selection Algorithm (FRCA) in [17] has been developed with the aim of cluster selection and cluster formation with the entity of CH, node, and Intermediate Node (IN). This algorithm is capable of overcoming the limitations of previously developed algorithms such as Low-Energy Adaptive Clustering Hierarchy (LEACH), Weighted-based Adaptive Clustering Algorithm (WACA) [21]. Additionally, FRCA algorithm performs well in case of these algorithms within Wireless Sensor Network (WSN). However, LEACH and WACA algorithms are not suitable for M2M network. This is because, the considered parameters is for dynamic nodes and depends on the mobility, distance, received signal strength indicator, average power and cost. It can be summarized that in FRCA, the number of CH is decreasing if the number of node keeps increasing. However, this is not practical for M2M neighbor network environment. In contrast for the M2M network, the scenario will be quite different where CH keeps increasing with the
increase of number of node. In addition, for FRCA, there are too many CHs in clustering results difficulties in controlling clusters. For instance, 100 nodes generate 4 clusters and the processing rate is not satisfactory as well. According to the FRCA, it is compared with other algorithms such as Cluster Based Routing Protocol (CBRP) [21], WACA, and Scenario-based Clustering Algorithm for Mobile ad hoc networks (SCAM) [22].

In this comparison, it can be summarized that the number of CHs is fully dependent on the fuzzy relevance degree whereas in the M2M clustering environment it is really difficult to follow this. However, the ultimate goal for M2M clustering is time synchronization and for that parameters can be chosen as time and distance for M2M environment.

**PROPOSED CLUSTER-BASED TIME SYNCHRONIZATION SCHEME**

Due to shortcomings of the current protocols and schemes, the novel concept of cluster-based time synchronization strategy introduced to overcome those problems. The proposed cluster-based time synchronization for M2M neighbor network is illustrated in Figure 3.

![Figure 3: Cluster-based time synchronization strategy](image)

Figure 4 shows the proposed scheme’s steps. Clustering scheme is the initial process divided into two segments which are CH selection and cluster formation that performed based on time and distance. In the intra-cluster and inter-cluster synchronization, two way time exchange techniques are employed to conclude the time synchronization between neighbor nodes and cluster heads through establishing a hierarchical topology or flat structure. Finally, linear least square method [16] is applied to attain a high level of precision.
The proposed scheme targets machines in an environment where machines are located in unplanned and uncoordinated way. For a given set of M2M nodes in a phenomenon area, the goal of the cluster formation is targeted to select a number of CHs such that the total overlapping area of all cluster communication ranges is minimal. It is assumed that the M2M network is largely deployed, and all nodes are stationary once deployed. Each node is aware of its position and can obtain estimated positions of neighbor nodes utilizing wireless links through IEEE1588 or mBS. In the proposed cluster-based time synchronization scheme, Sync broadcast mechanism is applied to conduct the time synchronization between CHs, CMs, and clusters as well. Therefore, the main task is to CH selection from the randomly installed nodes in order to accomplish cluster formation. Firstly, the CH initiates the synchronization process through Sync broadcasting to all CMs. The CMs which have received the synchronization time and receiving time are synchronized with CH node through comparing the time difference. The CH election algorithm is given as pseudo code in Algorithm 1

Algorithm 1: CH election algorithm

<table>
<thead>
<tr>
<th>Abbreviations:</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACHN:</td>
</tr>
<tr>
<td>NN:</td>
</tr>
<tr>
<td>ST:</td>
</tr>
<tr>
<td>UCH:</td>
</tr>
<tr>
<td>NN:</td>
</tr>
<tr>
<td>CM:</td>
</tr>
</tbody>
</table>

1. Define method as CH_selection( )
2. Set this node as ACHN.
3. Broadcast to all NN to join as CM and update neighboring table.
4. If all nodes are attached with other CH.
5. Compare latest ST & distance of all nodes with ACHN.
6. If any of the node’s ST is most recent than ACHN and recent node’s distance is smaller than other CH.
7. Attach with the recent CH.
8. Broadcast the new CH information to all neighbors.
9. Set the attached CH as UCH.
10. 
    Call ClusterFormation() method. // algorithm 2
11. Else
12.  Skip attachment with CH.
13.  
    If ACHN is away from one hop but nearest (ST & Distance) to old CH
14.  Join with the CH
15.  Else
16.  
    If ACHN is announced as CH by New Node.
17.  Listen to the new node.
18.  Declare ACHN as IN.
19.  Else
20.  Compare latest ST & distance of all nodes with ACHN.
21.  
    If ST == ∞ for ACHN and all other nodes.
22.  Wait for 1 second.
23.  Else
24.  
    If ST is most recent for any New Node.
25.  Attach with the node as CH.
26.  Broadcast the new CH information to all neighbors.
27.  Set the attached CH as UCH.
28.  Call ClusterFormation() method.
29.  Else
30.  Skip the attachment (reject invitation) with Nodes.
31.  Message to new nodes, Set old CH as current CH.

Once CH selection is done then the next step is to form the cluster. The cluster will be created by using the cluster formation algorithm which is given in pseudo code wise as shown in Algorithm 2.

**Algorithm 2:** Cluster formation algorithm

1. Define method as ClusterFormation()
2. Initialize Cluster as CH
3. If node distance <= 30 meter than the old CH
4. Broadcast to all neighboring nodes to join with CH
5. If number of hop <= 2
6. Discard the invitation message.
7. Announce as IN.
8. Else
9. Send ACK for acceptance.
10. Update node as CM.
11. Else
12. Search for another CH.
13. Send request to CH to update the status of CM.
14. Receive ACK from CH.
15. Announce nodes as CMs.
16. Attach all CMs with CH

In the proposed scheme, once the clustering formation is accomplished, the intra-cluster clock synchronization scheme will be applied for the entire cluster members to synchronize the clocks which will lead to the establishment of a better synchronized network.
Furthermore, the process which is inter-cluster synchronization will be performed with the clusters after selecting the CH, CM, and IN.

1- Intermediate Node (IN) selection

Intermediate node (IN) plays a significant role for the inter-cluster synchronization by selecting the short distanced nodes within the clusters. According to the proposed clustering process, the CH is selected over all neighboring nodes. The selected CH begins clustering process with the CMs. The IN selection can be done through two CHs which is based on the distance and recent sync time. The IN selection procedure is mentioned stepwise according to the flowchart in Figure 5.

**PERFORMANCE EVALUATION AND RESULT ANALYSIS**

The performance evaluation of the proposed scheme is carried out based on these two vital metrics namely clock offset and clock skew in order to achieve precise synchronization. In order to obtain average clock offset as well as the better synchronization accuracy, The performance of the proposed scheme has been evaluated numerically by considering equations 13, 14, 20 and 21 an synchronization processes are considered to be carried out for 25 times with different time difference in standard deviation (σ =10 µs, 15 µs, 20 µs, and 30 µs) for both intra-cluster and inter-cluster scheme. For the clustering scheme algorithm for CH selection, cluster formation and IN selection evaluated by using Matlab. The selective parameters are summarized in Table 1.

![Flowchart of IN selection](image)
Table 1: Parameters setting

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network size</td>
<td>200x200 m², 200x300 m², 100x300 m²</td>
</tr>
<tr>
<td>Transmission range</td>
<td>30 m</td>
</tr>
<tr>
<td>Cluster radius</td>
<td>30-40 m</td>
</tr>
<tr>
<td>Delay difference standard Deviation</td>
<td>10 µs, 15 µs, 20 µs, 30 µs</td>
</tr>
<tr>
<td>Distance of M2M node</td>
<td>0-30 m</td>
</tr>
<tr>
<td>Time intervals</td>
<td>1 sec</td>
</tr>
<tr>
<td>Synchronization process</td>
<td>25 times</td>
</tr>
</tbody>
</table>

1- Result Analysis for Clustering Scheme

In Figure 6 shows the CH selection and cluster formation for different number of nodes. When the CH created, it starts dominating all the CMs and forms the cluster. In some cases, some of the nodes may not join the cluster because of unsatisfactory clock time and distance. In addition, some of the clusters may have an interference with another cluster due to the distance and minimum clock. It can observed that as the number of CH increases, the number of M2M nodes keeps increasing as well.
The number of CHs keeps increasing with the increase of number of nodes and network size (100x300 m$^2$, 200x200 m$^2$, and 300x300 m$^2$) as shown in Figure 7. The results ensure that the ratio of the CH and nodes is satisfactory. In a general view, if the number of nodes keeps increasing, then a crowded network will be created which will be an ultimate burden for the network. This network overhead can be reduced through clustering. In this way, a number of CHs will form clusters with a number of CMs. The CHs will synchronize the CMs in the network. Then, after a while, if number of nodes keeps increasing, then number of CHs will be approximately linear which is satisfactory in terms of message overhead reduction.
2- Result Analysis for Intra-cluster Synchronization Scheme

In Figure 8, it is observed that when number of node is 10 and $\sigma$ is 10 µs then after 20 synchronization processes average, the clock offset reaches to 1 µs. However, the clock offset is can be eliminated after 25 processes which ensure high synchronization accuracy. Furthermore, the accuracy becomes degraded for large value of $\sigma$. For instance, Figure 11 suggests that when value of $\sigma$ is 30 and synchronization process is carried out for 10 times, then the average offset increases to approximately 10 µs. This results in an unexpectedly less significant accuracy of synchronization. Meanwhile, with the repetition of synchronization process up to 25 times, the average offset is significantly reduced to 7 µs that guarantees precise synchronization.

![Figure 8: Average clock offset for 50 nodes](image)

![Figure 9: Average clock skew for 250 nodes](image)
Clock skew is another significant performance metric for the intra-cluster synchronization scheme. The frequency accuracy performance is evaluated by measuring the average clock skew with a number of synchronization processes for $\sigma = 10 \, \mu s$, $15 \, \mu s$, $20 \, \mu s$, and $\sigma = 30 \, \mu s$, as shown in Figure 9. In a worst case scenario, when $\sigma$ is $30 \, \mu s$ and number of synchronization processes are 25, and then clock skew is significantly reduced from 8.5 to nearly 0.35 parts per million (ppm). This is how the ultimate aim of obtaining clock skew is achieved with the minimization of frequency error.

The performance of the proposed intra-cluster scheme is compared and benchmarked with the mobile assisted receiver-receiver time synchronization scheme proposed in [16]. The frequency accuracy of proposed scheme is a better precise synchronization than mobile assisted- synchronization scheme which uses least-squares linear regression algorithm.

In mobile assisted scheme, two ways messaging system has been used whereas the proposed scheme uses a hybrid messaging system (both one way and two ways). In consequence, mobile assisted synchronization scheme suffers from timing accuracy as well as message flooding with an increased number of nodes. However, the intra-cluster synchronization scheme fully focuses on these issues and formulates the strategy through clustering process which has already been shown in the result analysis of Figure 8 and 9.

The following configuration in Figure 10 gives overall comparison of offset calculation for the proposed intra-cluster scheme and conventional scheme. The comparison in Figure 10 is composed from the Figure 8 and the conventional clock offsets. It can be seen that the proposed scheme’s average clock offset is better from the initial synchronization process which means the proposed scheme can easily achieve precise synchronization.

Generally, different standards have different timing accuracy and frequency accuracy requirements. For instance, for the real time application, the timing accuracy requirement of Wideband Code Division Multiple Access/ Time Division Duplex (WCDMA/TDD) M2M is 2.5 $\mu s$, which has been obtained straight forwardly through 13 synchronization processes in proposed scheme.

![Figure 10: Comparison of the clock offset between the intra-cluster and conventional synchronization schemes](image)
CONCLUSION

The proposed cluster-based time synchronization scheme for M2M neighbor network has introduced to improve the synchronization accuracy as well as reducing the message overheads. The proposed scheme differs from previously developed IEEE 1588 scheme in the sense of forming clusters among the random number of nodes and applying hybrid (two ways and one way) messaging system to intra-cluster and inter-cluster time synchronization schemes. The scheme involves updating and synchronizing the clocks through exchanging Sync messages within the entire network. The proposed scheme is synchronization scheme is cluster based and can be applied for a large number of nodes in M2M network.

In this paper, it is demonstrated that all the schemes provide satisfactory performance than the conventional synchronization strategy for M2M network with better performance. The analytical results show that the performance of the proposed scheme is better than MS assisted strategy is. Furthermore, the clock offset and skew are minimized which is assured the precise synchronization. Our future research will be to mitigate the interference problems in the cluster level.

REFERENCES


