P2P Data Management in Mobile Wireless Sensor Network

SYEDA NIDA SAHAR*, FAISAL KARIM SHAIKH**, AND IMRAN ALI JOKHIO***

RECEIVED ON 02.01.2013 ACCEPTED ON 20.03.2013

ABSTRACT

The rapid growth in wireless technologies has made wireless communication an important source for transporting data across different domains. In the same way, there are possibilities of many potential applications that can be deployed using WSNs (Wireless Sensor Networks). However, very limited applications are deployed in real life due to the uncertainty and dynamics of the environment and scare resources. This makes data management in WSN a challenging area to find an approach that suits its characteristics. Currently, the trend is to find efficient data management schemes using evolving technologies, i.e. P2P (Peer-to-Peer) systems. Many P2P approaches have been applied in WSNs to carry out the data management due to similarities between WSN and P2P. With the similarities, there are differences too that makes P2P protocols inefficient in WSNs. Furthermore, to increase the efficiency and to exploit the delay tolerant nature of WSNs, where ever possible, the mobile WSNs are gaining importance. Thus, creating a three dimensional problem space to consider, i.e. mobility, WSNs and P2P. In this paper, an efficient algorithm is proposed for data management using P2P techniques for mobile WSNs. The real world implementation and deployment of proposed algorithm is also presented.

Key Words: **Mobile Wireless Sensor Networks, Peer-to-Peer, Data Transport and Management, Static Network, Mobile Network.**

1. INTRODUCTION

Data management is a process of developing data architecture and procedures that deal with data and are executed on a regular basis. If the data is properly managed it will be easily accessed by others in a reliable manner. The data management is important for every type of network either it is wired or wireless. The data management in wired networks has been reached their maturity during past decades [1]. In contrast, data management in evolving networks, i.e. P2P, MANETs (Mobile Ad Hoc Networks) and WSNs are still uncharted terrain [1]. Recent works have witnessed an increasing interest in the field of WSNs for collecting and processing data (data management) in a distributed manner from either static or mobile scenarios [2].

Data management in WSNs is different from traditional database systems because of its low capabilities or limited resources [3]. But traditional approaches are still

* Post-Graduate Student, Department of Telecommunication Engineering, Mehran University of Engineering & Technology, Jamshoro.
** Associate Professor, Department of Telecommunication Engineering, Mehran University of Engineering & Technology, Jamshoro.
*** Associate Professor, Department of Software Engineering, Mehran University of Engineering & Technology, Jamshoro.

being used in WSN especially P2P approaches because of certain similarities between WSN and P2P systems, e.g. each node in the network shares other node's resources, does not have permanent connectivity, has equal responsibilities and is fully independent regarding its own resources, has separate ID and behaves as a router when forwarding the data [4]. Along with similarities there are also differences with P2P systems such as scare resources, which cause P2P data management protocols to inefficiently utilize the resources of WSNs. The advantage of applying P2P technology in WSN is to exploit the decentralized approaches, which reduces the number of transmissions for data retrieval and replication process. The replication process assures reliability in case of node failures which is frequent in WSNs. To make use of these advantages and to overcome the problem of inefficient use of resources many protocols are developed and tested [3-5]. Unfortunately, these protocols are developed only for static networks and provide no support for the mobility.

The data management in mobile networks is a more challenging than static networks. In MWSNs (Mobile Wireless Sensor Networks), a subset of nodes changes their location in order to collect and disseminate the data instead of routing the data in order to efficiently utilize the network recourses [6]. This paper explores how to support mobile nodes in an established static network which is using P2P approaches for data management. Furthermore, we present the implementation and deployment of the proposed algorithms in real world scenario for the proof of concept.

Let us consider a scenario of smart hospital where the environment is monitored and controlled by the WSN. The hospital rooms and corridors are equipped with static wireless sensor nodes and the data inside the network is managed by P2P system similar to [4,7]. The patients and their data plays major role in clinical support system. The patients are generally mobile and their movement cannot be restricted in order to send data towards the base station. The major issue here is how to integrate mobile patients with the existing WSN such that the data gather by the patients (e.g. heart rate, blood pressure, etc.) can be transported to the base station. This paper presents algorithms to detect the presence of WSN and transferring the data to the network. After receiving the data WSN stores this data with P2P technique in static nodes and if any exception occurs the data will be transported to the base station using P2P techniques [4-5].

The rest of the paper is organized as follows. Section 2 briefly describes system models and mobility model. Section 3 explores the related work and Section 4 elaborate the VCP. Section 5 presents the proposed algorithms for data management in MWSNs. Implementation and deployment of proposed scheme is discussed in Section 6. Section 7 comprises of result and the Section 8 concludes our work.

2. MODELS

In this section we first describe the considered system model then briefly elaborate the mobility model.

2.1 System Model

In a finite network, consider a WSN has N static nodes and M mobile nodes numbered [0 to N-1] and [N to M] respectively, where node numbered 0 is a sink. Each node has sensors and except sink each node has small memory, less computational resources, limited radio communication and less power. Whereas, sink node has more capabilities than other nodes in the network, e.g. continuous power supplies, more memory and high computational resources. The static and mobile network senses data, hash it and store it at a static node having virtual position closest to hashed value. These nodes also respond to a request that is generated by the base station. The mobile nodes send data towards the base station through the static nodes if any exception occurs.

2.2 Mobility Model

The mobile nodes inside WSN can move either in a random, predictable or fixed fashion. In random mobility node can move anywhere in the network and its movement is unpredictable. For predictable mobility pattern the movement of the node can be predicted. Whereas, for fixed mobility pattern the movement of a node is known in advance. Irrespective of mobility pattern our proposed system can be deployed in all scenarios but for proof of concept we consider fixed mobility pattern.

3. RELATED WORK

In P2P, no single node controls the network and each node is a peer to all other computers in the network. There is no centralized management in this approach. P2P has provided much flexibility to applications and solved most of the problems of WSN [8].

In the same way, VCP (Virtual Cord Protocol) inspired by structured P2P technology-provides efficient data management using the routing techniques with DHT services [3-5]. It uses virtual positions for data transport, where each node maintains a successor, a predecessor and a routing table to store information of its neighbors. Each node in the network can have a maximum of two virtual neighbors and can broadcast a message to update the other nodes about its current status and to assign virtual positions to the new nodes. All the nodes are responsible to hold information of their single hop neighbors only, so the functionality of each node would not be affected by the size of the network. In [4-5] VCP has been implemented on mica2 sensor nodes and its applicability in real world scenario has been shown. The major drawback with VCP and related approaches [1] is that they do not support mobility.

MWSN, as its name suggests, has the presence of mobile nodes in the network [9]. MWSN has advantages over static WSN as it causes improved coverage, superior channel capacity and enhanced target tracking [9]. This evolving technology can be applied for land [10], ocean [11] and air [12] exploration and monitoring, Habitant Monitoring [13], automobile applications [14] and also on other scenarios. MWSN can derive their coordination with other nodes either through the dedicated coordinates or through the communication with other nodes in the network [2]. Recently, data management techniques have been proposed for MWSN [2]. In these techniques data are sent either periodically or on demand. But still research is going on for efficient data management in MWSN. The main drawback of these techniques is that they do not exploit P2P advantages and their major focus is on exploiting mobility only.

4. VIRTUAL CORD PROTOCOL

The VCP provides underlay routing between nodes in the network. VCP has two basic operations one is to store and second is to retrieve. It assigns virtual positions to each node in the network. Each node maintains a routing table to store information of its neighbors in addition it also maintains its successor and predecessor. All nodes are responsible to hold information about a portion of the network so the functionality of each node does not effected by number of nodes in the network.

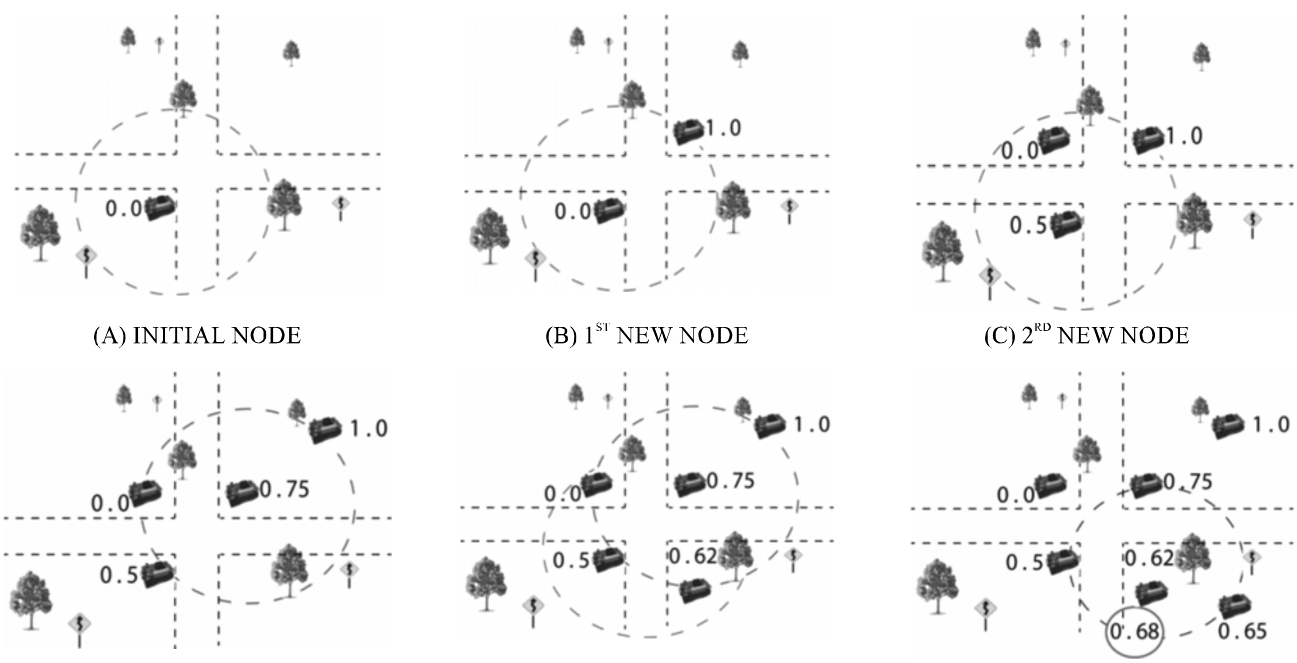
4.1 VCP Joining Process to Build a Network

Each node in the network has a separate virtual position. These virtual positions are assigned by a node that is already part of the network. To start the network one node is preprogrammed as initial node and assigned a virtual position 0 or Start(S). Each node that becomes part of the network starts broadcasting HELLO message to update the other nodes about its status and to assign the virtual positions to the new nodes joining the network. If a new node added in the network receives a HELLO message, it sends a request to the old node to set its virtual position in the network. The assignment of the virtual positions can be divided into four types (Fig. 1):

(i) If new node added in the network is second node it will be assigned virtual position 1 or End (E). Now all virtual position will be between these two end points.

(ii) After the second node joins the network, if a new node receives a HELLO message from either S or E, the old node (S or E) assigns its own virtual position to the new node and gets a new virtual position that is obtained as if its old virtual position is S, it will be assigned a new virtual position between its old virtual position (S) and its successor whereas if its old virtual position is E, it will be assigned a new virtual position between its old virtual position (E) and its predecessor.

(iii) If a new node does not receive HELLO message from S or E, but receives HELLO messages from the two intermediate old nodes that are adjacent



(A) INITIAL NODE (B) 1ST NEW NODE (C) 2RD NEW NODE

(D) 3TH NEW NODE (E) 4TH NEW NODE (F) 5TH NEW NODE

FIG. 1. VCP JOINING PROCESS

to each other (successor and predecessor to each other), then the new node will be assigned a virtual position between these two old nodes.

(iv) If a new node receives a HELLO message from only one intermediate old node, then old node creates a virtual position between its own virtual position and its successor. The new node will now be assigned a virtual position between this newly created virtual position and the old node's own virtual position. This new node is called a virtual node. This node can neither assign a virtual position to the other new nodes nor can it participate in the data transport process.

4.2 VCP Routing

An example scenario of the VCP is shown in Fig. 2. Each node is assigned a virtual position within the range 0-1. If node 0.5 originates data P, P will be hashed and suppose the result of the hashed function is 1.0. Node 0.5 will search in its routing table for the nearest node towards host node 1.0 and find node 0.75 as a nearest node. Node 0.5 will send data P to node 0.75.Node 0.75 will receive the packet and search for nearest neighbors in its routing table. Node 0.75 will find node 1.0 in its routing table and directly send data P to node 1.0.Node 1.0 will receive the data P and store in its memory.

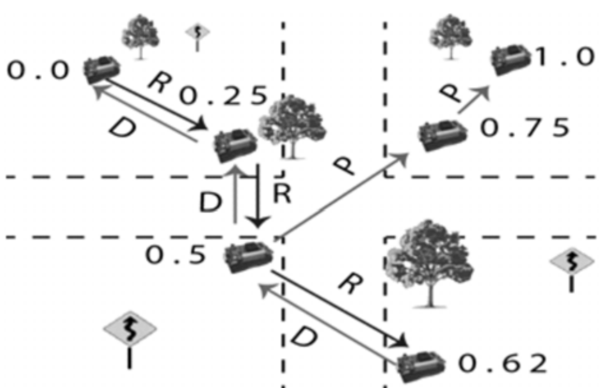


FIG. 2. VCP DATA STORAGE AND RETRIEVE PROCESSES

Now suppose node 0 wants a data D and its hashed value is 0.6. Node 0 will search in its routing table and sends request to nearest node 0.25. Node 0.25 searches in its routing table and find the nearest node 0.5 towards host node. Node 0.31 forwards requests to node 0.5.Node 0.5 receives request and find node 0.62 in its routing table. Node 0.5 directly forwards request to node 0.62. Now node 0.6 receives a request and sends data D to requested node 0 by following the path Node 0.6 ? Node 0.5 ? Node 0.25 ? Node 0.

5. DATA MANAGEMENT IN MWSN

This section presents the proposed algorithms for data management in MWSNs.

5.1 Overview

VCP is a static network that uses P2P approach for data management in WSN. In proposed system, the VCP static network has been extended by adding mobile nodes in the network. The proposed system consists of both the static nodes and mobile nodes. All nodes uses P2P approach for data storage process and data retrieve process. All static nodes will be programmed with VCP code, i.e. all static nodes will be assigned virtual positions for communication, if a node senses a data this data will be hashed and stored at a node having virtual position closest to the hashed value and if a node requires a data the request will be sent to host node only. All mobile nodes are identified by their unique source addresses. If a mobile node will sense a data this data will be hashed and stored at a static node having virtual position closest to hashed value. The base station can retrieve any specific data from the network by sending request to a host node and it can communicate with any node in the network.

5.2 Network Construction

The network starts with a preprogrammed initial static node having a virtual position either S (Start Position) or E (End Position). With the first node, the network will be started and it will start sensing. If a new static node will be added in the network it will be assigned a virtual position according to VCP joining process as mentioned in [15-16]. Each mobile node in the network starts its assigned function, e.g. sensing, hashing and storing data to host node, as soon as it found a static network around it. Each node in the network either it is mobile or static maintains its routing table of single hop neighbors as routing will be done using single hop neighbors only. If the base station requires a data, the data will be hashed and request will be sent to a host node. If the base station wants to communicate with specific node in the network then query will be sent to particular node either mobile or static.

5.3 Mobile Node Data Transport and Management

Data transports in mobile nodes are carried out using single hop neighbors. If source and destination nodes are not in direct communication then request will be sent to a node having virtual position closest to the destination node. This procedure continues until destination is reached.

The data storage process of mobile node is shown in Fig. 3. To explain the data storage process it is assumed that mobile node senses data D from point 1 to point 3 and its hashed value is 0.5. The data storage process of mobile node from point 1 to point 3 (as shown in Fig. 3) is explained below:

At point 1, mobile node is in communication range of only node 0.75. The Mobile node will send data to node 0.75 (line 1-5 in Alg.1 and line 1-3 in Alg. 2). Node 0.75 will

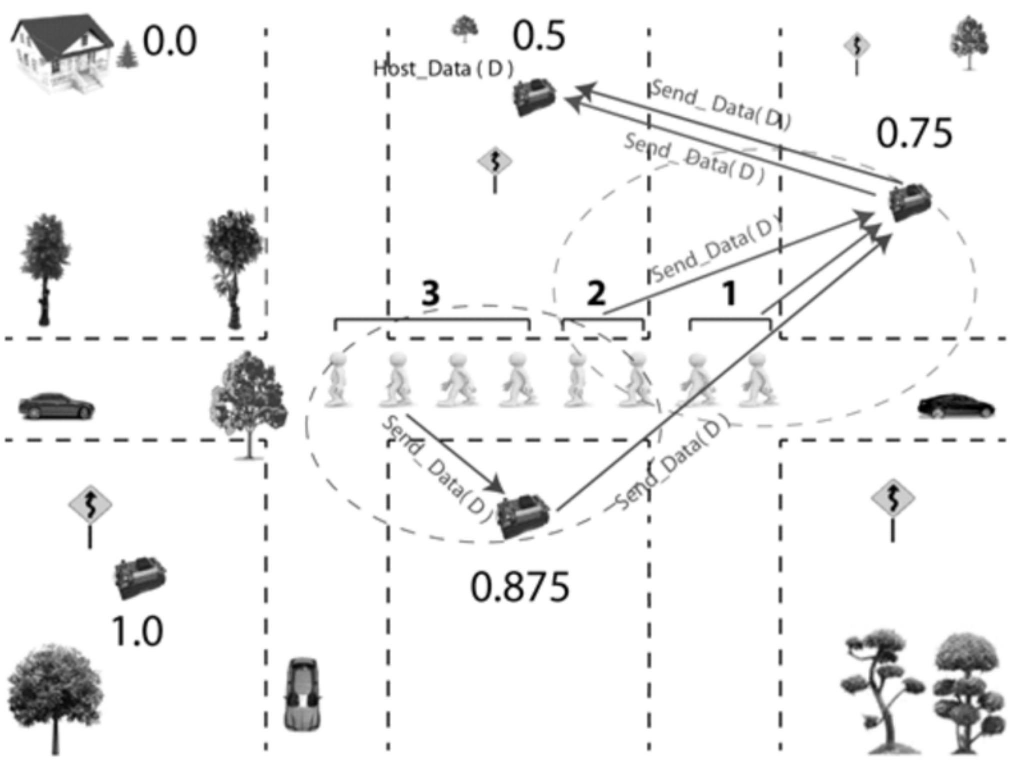


FIG. 3. MOBILE NODE DATA STORAGE PROCESS

search in its routing table and finds node 0.5 that is equal to hashed value 0.5. Node 0.75 will send data to node 0.5 for storing it (line 7-11 Alg. 2). The node 0.5 receives the data and stores it in its memory (line 7-14 Alg. 2).The node 0.5 is host of data D.

At point 2, mobile node is in communication range of two nodes, node 0.75 and 0.875, but since node 0.75 is nearest to node 0.5 it will send data to node 0.75 (line 1-5 in Alg.1 and line 1-3 in Alg. 2). Node 0.75 performs in same way as mentioned above in point 1.

At point 3, mobile node is in communication range of only 0.875. Now mobile node will communicate with node 0.875 to transport data (line 1-5 in Alg.1 and line 1-3 in Alg. 2). The data D will be transported using path node 0.875 $\rightarrow$ node 0.75 node $\rightarrow$ 0.5.

The important terms used in algorithms are listed below:

NT = Neighbor Table of a node or an array of neighbor table
NT[MIN] = minimum number in sorted table (NT[1])
NT[MAX] = maximum number in sorted table(NT[LAST])

```
1.    if NT != null then
2.        Sense Data
3.        Hashed_Value = Hash(Data)
4.        Send(Hashed_Value, Data) (see Alg. 2)
5.    end if
```

Algorithm 1 Sense data from environment

```
1.    if Hashed_Value<NT[MIN] then
2.        SendData(NT[MIN, Data])
3.    end if
4.    if (Hashed _Value>NT[MAX]) then
5.        SendData(NT[MAX], Data)
6.    end if
7.    if Hashed_Value > = NT[MIN] and Hashed_Value < = NT[MAX]
8.        for(int i=1; i<= MAX; i++)
9.            if(NT[i]<=  Hashed_Value) then
10.               if ( i == 1 && Pos != NT[i])
11.               SendData (NT[i], Data)
12.       else
13.               store(Data)
14.       end if
15.       if ( i>1)
16.               N1 = NT[i] - Hashed_Value
17.             N2 = Hashed_Value - NT[i-1]
18.             if( N1 <= N2 )
19.               if ( NT[i] != Pos)
20.                   SendData(NT[i], Data)
21.             else
22.                   store(Data)
23.                     end if
24.             else
25.               if (NT[i-1] != Pos)
26.               SendData(NT[i-1], Data)
27.                 end if
28.                 else
29.               store(Data)
30.                       exit for
31.                 end if
32.     end for
33. end if
```

Algorithm 2: Sending data and hashed value.

The data retrieve process is shown in Fig. 4. The base station can retrieve a particular data from the network and all or part of record of a nMWSNsode either mobile or static. Suppose the base station requires data D. The base station will hash this data and gets hash value 0.5. As node 0.5 is in direct communication with the base station, it will send request of data D to node 0.5 as shown with blue color in Fig. 4. After receiving request node 0.5 send response to the base station as shown with brown color in Fig. 4. This procedure is for a particular data in the network. Now suppose the base station want to communicate directly to the mobile node. The base station has information about current position of mobile node. Let mobile node be at point 2. The base station sends request to mobile node following the route node 0.5 $\rightarrow$ node 0.75 $\rightarrow$ mobile node. The mobile node responses to the request of the base station following the route node 0.75 $\rightarrow$ node 0.5 $\rightarrow$ node 0.0 (the base station) as shown in Fig. 4.

5.4 Mobile Node Table Update Process

To route a message from source to destination accurately each node must update its routing table. The routing table update process of static nodes is explained in [3-5]. In this paper we will discuss routing table update process of mobile node only that is required because of two reasons: first, to update the current virtual position of its neighbors and their existence in the network. Second, as it is a mobile node its neighbors will be changed continuously.

The routing table update process of mobile node is shown in Fig. 5 whereas its Pseudocode is explained in Alg 3 and Alg 4. Each node broadcasts a HELLO message to update its current status to other nodes in the network.

At point 1, mobile node is receiving a HELLO message of node 0.75 and updates its routing table with node 0.75 (line 1-3 in Alg. 3 and line 1-4 Alg. 4).

At point 2, mobile node is receiving a HELLO message of another node 0.875 and adds this node in its routing table (line 1-3 in Alg. 3 and line 5-8 Alg. 4).

```
1.    if SN ⊄ NT
2.        call  ADD(SN) (See Alg 4)
3.    end if
```

Algorithm 3: Received a HELLO message from a node SN

```
1.    if  ( NT = Null) then
2.        Add SN in NT[MIN]
3.        MAX = MIN
4.    end if
5.    if SN>NT[MAX]
6.        Add SN in NT[MAX+1]
7.        MAX = MAX + 1
8.    end if
9.    if SN<NT[MIN]
10.       for ( int i = MAX; i>= MIN; i--)
11.           NT[i+1 ] = NT[i]
12.       end for
13.       NT[MIN] = SN
14.   end if
15.   if (SN > NT[MIN] & SN< NT[MAX]) then
```

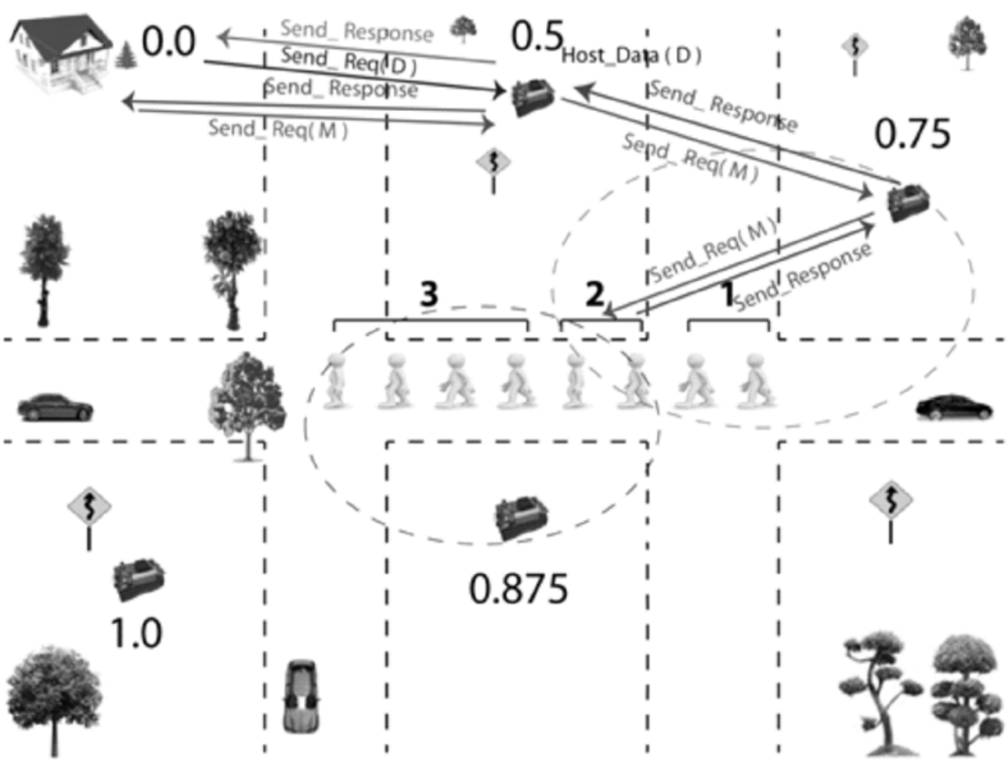


FIG. 4. MOBILE NODE DATA PROCESS

```
        for (int i = MIN; i<=MAX ; i++)
            if(SN< NT[i] ) then
                for (int j = MAX+1; j> = i ; j--)
                    NT[j] = NT[j-1]
                end for
                NT[i] = SN
            exit for
            end if
        end for
    end if
```

Algorithm 4: Adding process for a sensor node.

At point 3, mobile node is out of communication range of node 0.75 and a HELLO message of node 0.75 is not received by the mobile node. The Mobile node will remove node 0.75 from its routing table as shown in Fig 5 (line 1-3 & line 7-12 in Alg. 5).

```
1.   if NT[i] HELLO message not received in 1 second
2.       w++
3.   end if
4.   if NT[i] HELLO message Received
5.       w=0
6.   end if
7.   if (w=4)
8.       for (j=i ; j<= MAX; j++)
9.           NT[j] = NT[i+1]
10.  end for
11.  MAX = MAX - 1
12.  end if
```

Algorithm 5: Remove Neighbor from network table.

6. REAL WORLD DEPLOYMENT SETUP

In this section, real world deployment setup is discussed. First, hardware and software tools are discussed then deployment scenario is explained. In last implementation and execution details are presented.

6.1 Hardware and Software Tools

Six mica2 sensor nodes have been used for the implementation of VCP with the mobile node scenario. Five sensor nodes have MTS301 sensor boards and one sensor node is attached to MIB510 programming board to make the base station. TinyOS-1.1.15 has been used for programming the mica2 sensor nodes.

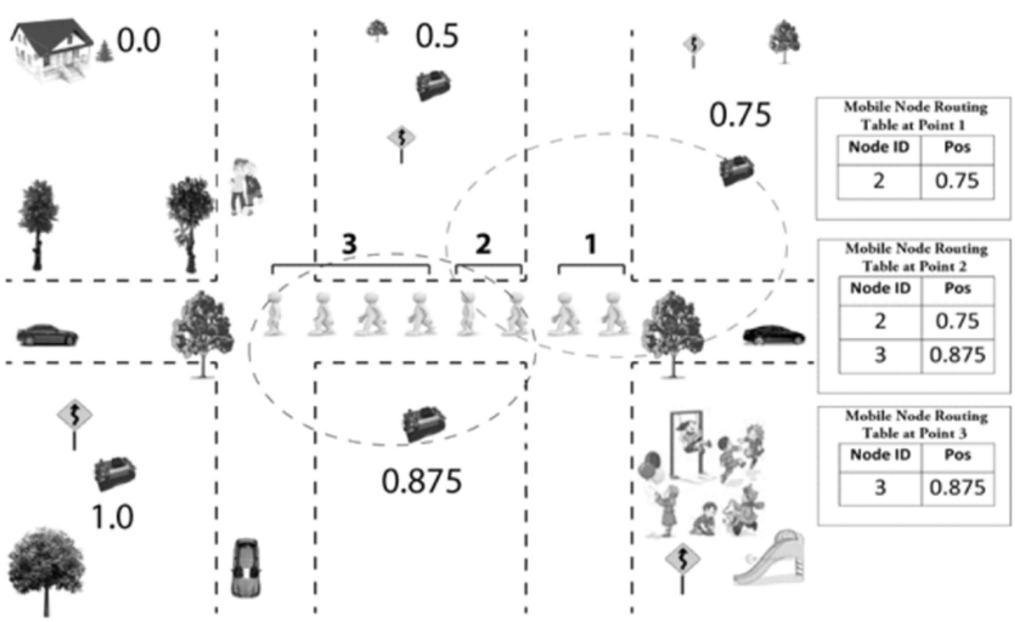


FIG. 5. ROUTING TABLE UPDATE PROCESS OF MOBILE NODE

6.2 Deployment Scenario

Six mica2 sensor nodes have been deployed in a hospital scenario. Five sensor nodes are static and one node is mobile. Among five static nodes, one node is the base station and four are randomly distributed in lobbies. One node is attached to a patient that behaves as a mobile node. Each static node has VCP functionality and senses light and sound reading from environment. For light data, when there is darkness then the lights will automatically on and in brightness lights will automatically off in lobbies. For sound data, when there is a noise, node will send report to the base station to take an action. Mobile node's data will be stored in static nodes using P2P technique. If any exception occurs, the mobile node will send data directly to the base station to take an immediate action.

As shown in Fig. 6 two nodes (node 0.875 and node 1.0) have been placed in lobby of the female ward and two nodes (node 0.75 and node 0.5) have been placed in lobby of the male ward. One node has been placed in information desk that behaves as the base station. Node 0.5 is in direct communication with the base station and is at single hop. Whereas node 0.75 is in communication range of node 0.5, node 0.875 is in communication range of node 0.75 and node 1.0 is in communication range of node 0.875. These nodes are at hop 2, hop 3 and hop 4 respectively. A node attached to a patient is present at the private room of the male ward and he is walking in the lobby.

6.3 Hardware Implementation

6.3.1 Static Network

The unique source address to each node is assigned and the nodes that are not part of the network are assigned a virtual position -1.0. Each node that

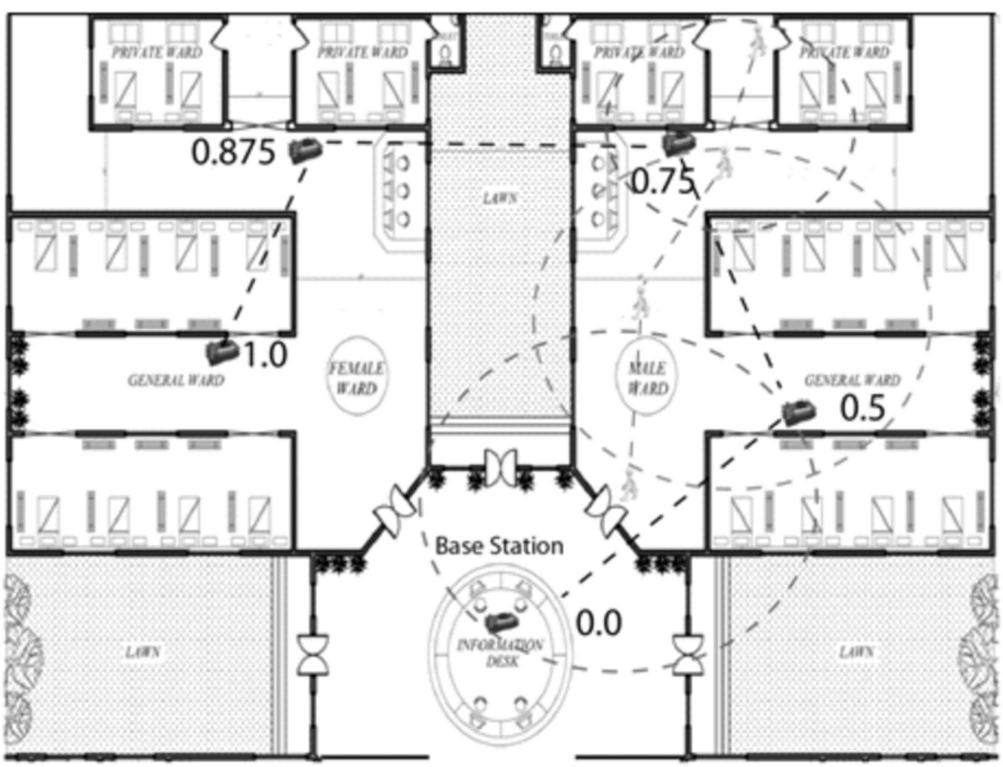


FIG. 6. MOBILITY IN VCP DEPLOYMENT SCENARIO

becomes part of the network starts broadcasting a HELLO message after every second along with the other network functions including sensing light and sound from the environment and maintaining the routing table. A variable SET_POS is used to check the status of the node to determine if the node is busy in setting the virtual position or not. This variable prevents the assignment of same virtual positions to multiple nodes and it also prevents assignment of two different virtual positions to a single node. The virtual positions are allocated between value S and E (0 and 1). Once the virtual positions are assigned, the data transmission is carried out using the single hop neighbors.

6.3.2 Mobile Network

The mobile node starts sensing data after receiving at least a HELLO message from a static node in the network. It senses data after every second and stores to a static node having virtual position closest to hashed value. Mobile node sends report to the base station using static network if any exception occurs.

7. RESULTS

The performance of the proposed scheme is evaluated with respect to the reliability of the scheme. The reliability is the ratio of number of packets generated by the mobile node to the number of packets received at the base station. The mobile node sends a packet to the base station after every second. The results are obtained by repeating experiments many times.

The Fig. 7 shows the number of packets received at the base station at the interval of one second. From Fig. 7 it is clear that the number of packets received at the base station from hop 1, means when the mobile node is directly communicating with the base station, is approximately 100% (as the base station is supplied with continuous power source, its transceivers performs better than other nodes in the network, i.e. nodes operated on batteries). At hop 2 it is 98% as the packets are first send to node that is in direct communication with the base station or at hop 1 then packets are send to base station by node at hop 1 therefore there is a slight loss of packets. At hop 3 PRR is 92% as packets are first send to node at hop 2 then hop 1 and finally the base station. Therefore, there is a loss of approximately 8% packets.

The mobile node when communicates with its router (first node towards the base station) its percentage of packets received at its first router is approximately 99% and when it directly communicate with the base station its percentage is 100%. For example, if the mobile node is sending a packet/message to node 0.75 as shown in Fig. 6, the number of packets received at node 0.75 will be 99% approximately. The remaining packet loss as shown in the Fig. 7 is due to Mac layer collision as mentioned in our previous work [4].

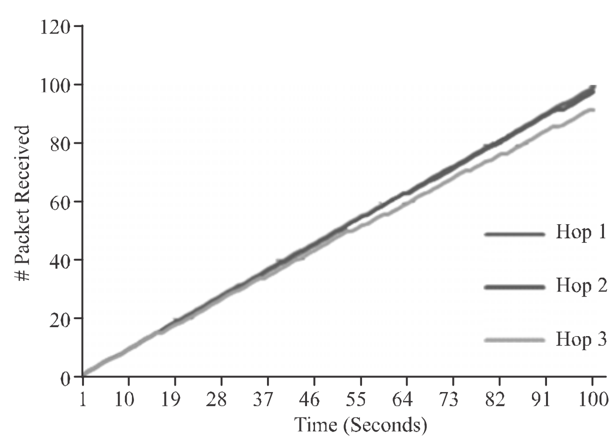


FIG. 7. NUMBER OF PACKETS RECEIVED AT THE BASE STATION VERSUS TIME

The Fig. 7 shows number of packets received at the base station from different hops when a mobile node is moving within communication range of 1st hop, 2nd hop and 3rd hop. But there will be a certain amount of packet loss when mobile node switches from one node to another towards the base station for example in Fig. 6, when the mobile node is near private room it is in communication range of node 0.75. When this node reaches near the kitchen it is out of communication range of node 0.75 and now it has to communicate with the base station through node 0.5 but this information about node 0.75 is not immediately acquired by the mobile node. Because if a HELLO message of a node is not received by the mobile node then it waits for three seconds. If within this period, a HELLO message of that node is not received then mobile node will eliminate this node record from its routing table. But within this period the mobile node tries to communicate with that node and due to this reason packets will be lost. The Fig 8 shows total number of packets received at the base station by the mobile node. As shown in Fig. 8 overall packets received at the base station is 95% and there is a loss between 4-6 packets when mobile node switches from one static node to another. In Fig. 8 shows movement of mobile node from hop 3 to hop1.

The Fig 9 shows the number of hops required to store data of mobile node having specified hashed values. The number of stages shows different location of mobile node when it is in communication range of different static node. At 1st stage, mobile node is in communication range of only node 0.75 as shown in Fig 6. At this stage to store data having hashed value from 0.0 -0.250 (host node 0.0) and from 0.9375-1.0 (host node 1.0) it has to send data at hop 3. To store data having hashed value from 0.251-0.625 (host node 0.5) and from 0.826-0.9375 (host node 0.875) it has to send data at hop 2. To store data having hashed value from 0.626-0.825 it has to send data at hop 1. The same procedure will be applied for 2nd stage (mobile node in communication range of node 0.75 and node 0.5), 3rd stage (mobile node in communication range of node 0.5 only), 4th stage (mobile node in communication range of node 0.5 and node 0.0) and 5th stage (mobile node in communication range of node 0.0). The number of hops required to store data in all these stages is shown in Fig. 9.

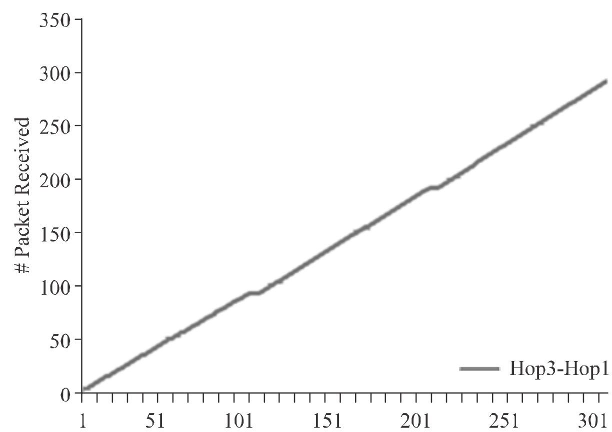


FIG. 8. NUMBER OF PACKETS RECEIVED AT THE BASE STATION VERSUS TIME

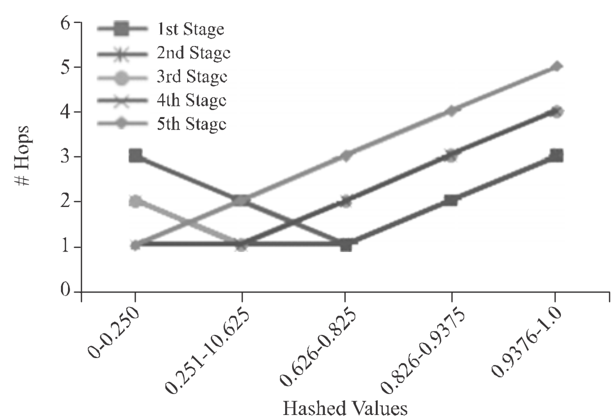


FIG. 9. NUMBER OF HOPS VERSUS HASHED VALUES

8. CONCLUSION

In this paper, we have proposed a P2P based data management scheme for MWSN. In this paper we have shown how mobile nodes data are stored in the VCP static network using P2P approach. The real world deployment of the proposed scheme is also presented to show its applicability in real scenario. The results show good performance in a real world scenario but there is need to improve VCP performance. Its future works involve solving issues in VCP and then implement the same strategy. The drawback of the proposed scheme is that the memories of mobile nodes are not utilizing to store data from other nodes (P2P approach). It is possible to propose a strategy that can utilize the mobile nodes memory like static nodes in the network.

ACKNOWLEDGEMENTS

This research was partially supported by Mehran University of Engineering & Technology, Jamshoro, Pakistan, and National ICT R&D Fund, Ministry of Information Technology, Pakistan under National Grassroots ICT Research Initiative.

REFERENCES

[1] Boncz, P.A., Bonifati, A., Böse, J., Böttcher, S., Kypros, P., Gruenwald, C.L., König-Ries, B., May, W., Mondal, A., Obermeier, S., Ouksel, A., Samaras, G., Sapkota, G., Steinmetz, R., and Viglas, S.D., "P2P, Ad Hoc and Sensor Networks-All the Different or All the Same?", Proceedings of Scalable Data Management in Evolving Networks, Schloss Dagstuhl, Germany, October, 2006.

[2] Chen, M., Kwon, T., Yuan, Y., Leung, V.C.M., "Mobile Agent Based Wireless Sensor Networks", Journal of Computer, Volume 1, No. 1, pp. 14-21, April, 2006.

[3] Awad, A., German, R., and Dressler, F., "P2P-Based Routing and Data Management Using the Virtual Cord Protocol (VCP)", Proceedings of 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing, Poster Session, pp. 443-444, Hong Kong, China, May, 2008.

[4] Awad, A., Sommer, C., German, R., and Dressler, F., "Virtual Cord Protocol (VCP): A Flexible DHT-Like Routing Service for Sensor Networks", Proceedings of 5th IEEE International Conference on Mobile Ad-hoc and Sensor Systems, pp. 133-142, Atlanta, GA, September, 2008.

[5] Awad, A, German, R., and Dressler, F., "Efficient Service Discovery in Sensor Networks using VCP", Proceedings of 15th ACM International Conference on Mobile Computing and Networking, Demo Session, Beijing, China, September, 2009.

[6] Khelil, A., Shaikh, F.K., Ali, A., and Suri, N., "gMAP: Efficient Construction of Global Maps for Mobility Assisted WSNs", Proceedings of the 6th International Conference on Wireless On-Demand Network Systems and Services, pp. 175-182, Snowbird Utah, USA., February, 2009.

[7] Akyildiz, I.F., and Vuran, M.C., "Wireless Sensor Networks", John Wiley and Sons, USA, 2010.

[8] Gutierrez, G., Mejias, B., Roy, P.V., Velasco, D., and Torres, J., "WSN and P2P: A Self-Managing Marriage", Proceedings of Second IEEE Inernational Conference on Self-Adaptive and Self-Organizing System Workshops, pp. 198-201, Venice, Italy, 2008.

[9] Zeinalipour-Yazti, D., and Chrysanthis, P.K., "Mobile Sensor Network Data Management", Book Chapter in the Encyclopedia of Database Systems (EDBS), Ozsu, M., Tamer; and Liu, L., (Editors), [ISBN: 978-0-387-49616-0], 2009.

[10] Dantu, K., Rahimi, M., Shah, H., Babel, S., Dhariwal, A., Sukhatme, G., "Robomote: Enabling Mobility in Sensor Networks", Proceedings of the 4th International Symposium on Information Processing in Sensor Networks, pp. 404-409, California, USA, 2005.

[11] Nittel, S., Trigoni, N., Ferentinos, K., Neville, F., Nural, A., and and Pettigrew, N., "A Drift-Tolerant Model for Data Management in Ocean Sensor Networks", Proceedings of the 6th International Workshop on Data Engineering for Wireless and Mobile Access, pp. 49-58, Beijing, China, 2007.

[12] Allred, J., Hasan, A.B., Panichsakul, S., Pisano, B., Gray, P., Huang J.H., Han, R., Lawrence, D., and Mohseni, K., "SensorFlock: An Airborne Wireless Sensor Network of Micro-Air Vehicles", Proceedings of the 5th International Conference on Embedded Networked Sensor Systems, pp. 117-129, Sydney, Australia, 2007.

[13] Sadler, C., Zhang, P., Martonosi, M., and Lyon, S., "Hardware Design Experiences in ZebraNet", Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, pp. 227-238, Maryland, USA., November, 2004.

[14] Eriksson, J., Girod, L., Hull, B., Newton, R., Madden, S., and Balakrishnan, H., "The Pothole Patrol: Using a Mobile Sensor Network for Road Surface Monitoring", Proceedings of the 6th International Conference on Mobile Systems, Applications And Services, pp. 29-39 Colorado, USA., 2008.

[15] Sahar, S.N., Shaikh, F.K., Jokhio, S.H., "P2P based Data Management WSNs: Experiences and Lessons Learnt from a Real-world Deployment", Mehran University Research Journal of Engineering & Technology, Volume 30, No. 4, Jamshoro, Pakistan, October, 2011.

[16] Sahar, S.N., Shaikh, F.K., Jokhio, S.H., "Peer-to-Peer Wireless Sensor Networks for Efficient Data Transport and Management", Proceedings of the IEEE Student Conference on Engineering Sciences and Technology, Jamshoro, Pkaistan, 2011.