International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-2 Issue-6, August 2013

Mobility in Name-Based Home Ad Hoc Networks

Tharinda Nishantha Vidanagama, Hidenori Nakazato

Abstract*— Wireless mobile ad hoc networks can be used as a means to ease and support life in an ordinary house. As future home appliances will have many useful built-in functions, a communication network that allows a user to access these built-in functions and to control the appliances is highly desirable. This paper proposes algorithms for clustering and message routing to manage mobility in a name-based home ad hoc network. The nodes are given names such as "living room TV", "kitchen oven", etc. for identification. This paper discusses the performance of three routing schemes for communication of nodes with mobility. The proposed algorithms handle mobility efficiently while ensuring a high accuracy on message delivery with a low number of control messages.

Index Terms*— Ad hoc network, Name-Based addressing and routing, cluster, routing cache, mobility.

I. INTRODUCTION

Wireless mobile communication devices have become common in everyday life. Wireless ad hoc networks can be used to extend their capabilities to provide convenience in daily life. The main attractions of ad hoc networks are their dynamicity, lack of infrastructure and the vast number of application possibilities.

Recently the number of wireless devices in the home environment has also risen. This trend continues to introduce wireless capabilities to household appliances and sensors as well. Most devices at a household remains at one point in the house such as television, refrigerator etc., where as some devices such as smart phones, laptops etc. has no fixed point. Creating a network of such devices at home would greatly assist the users to control and monitor their household environment.

Most new devices are fitted with technologies such as Wi-Fi, but it is possible that there are other devices with different short range technologies such as Bluetooth or infrared. Smaller sensor like devices may be powered by a limited power source. If the Wi-Fi access point is at a distance, transmitting at high power will reduce the lifetime of devices that use limited power sources. It is possible that a single Wi-Fi access point may not cover the entire household. The dependence on central access point(s) also has the risk of entire (partial) network failure. The home environment may consist of various nodes with many different functions and capabilities. Hence the heterogeneity of the nodes needs to be considered when forming a network as the transmission distance, processing power and memory etc. of the appliances and sensors may vary.

This paper proposes a futuristic home environment where household appliances and sensors with wireless communication modules are connected via an ad hoc network. Long-range networks such as Wi-Fi are also considered as being a part of the entire network. This network will allow users to access and control home appliances and sensors from anywhere in the network through other appliances or sensors (hereafter referred as nodes). Such a network is cost effective as no extra equipment is required and it also supports heterogeneity of nodes.

To provide ease of usage of the network, the use of everyday names such as "*kitchen oven, bedroom light*" etc. is proposed to identify the nodes. A solution such as IP with DNS is possible, but it would require the user to initially map the description or keywords to IP addresses when the network is setup or when a new device is added or removed from the house. Also additional hardware may be required.

In the proposed system for example, during setup of the home network a user walks to a location in his household with a device which has a user interface such as a smart-phone or laptop. Through this device the user is able to see a list of compatible wireless devices whose default names are given by the manufacturer. The user may also change some node names as preferred. Then the user will assign a common location name through his device to the other nodes in that location by multiply selecting them from the list. The user only needs to follow the same simple steps in other locations in the house to setup the home network. Also nodes added later will join the network automatically without any user intervention. Such an easily setup home network will provide convenience and support life in an ordinary house.

This paper discusses the formation of the clusters based on their location names and the formation of the name-based ad hoc network. The paper continues to discuss how mobility is managed in the network. Three routing strategies are also proposed and their performance is evaluated.

This paper is organized as follows. The following section 2 discusses some of the related works. Section 3 describes the network model and node naming. Sections 4 and 5 introduce the cluster organization and the network organization respectively. The routing protocol is explained in section 6 and section 7 presents the evaluation of the proposed methods through simulation. The latter section 8 concludes the paper.

II. RELATED WORK

A cluster based network is created using three conceptual data structures by all nodes in CBRP [1]. The *Neighbor Table* (NT) is used to store their bidirectionally linked neighbor nodes.

Manuscript published on 30 August 2013.
***** Correspondence Author (s)
Tharinda Nishantha Vidanagama*, Graduate School of Global Information and Telecommunication Studies, Waseda University, Tokyo, Japan.
Hidenori Nakazato, Graduate School of Global Information and Telecommunication Studies, Waseda University, Tokyo, Japan.

The *Cluster Adjacency Table* (CAT) is used to store the neighboring *Cluster Head* (CH) nodes. A CHs role is to provide the routing for all the member nodes in that cluster. Also a *Two-hop Topology Database* is built using periodical broadcasts of NT, that stores complete network topology information at most two-hops away. A form of source routing is used in both [1], [2] in order to discover routes. Routing overhead increases as the number of nodes increase, and the route must be discovered before any data is sent. Also in [2], the routes are also maintained with periodic routing updates. Reference [3] requires that all the nodes have similar high performance capabilities to build a routing table and nodes use Dijkstra's algorithm so that each node could find a shortest path to a destination.

A publisher/subscriber communication model for wireless mobile ad hoc networks is described in [4]. But in this proposal the amount of data kept for routing purpose is high, as much as five table structures. A similar approach is also used in [5], but uses "Named-Data" where data name prefixes are flooded and cached in the network. Data requests are also flooded in the network and any node with the requested data may respond. The data requester can select a suitable responder to receive the data. Intermediate nodes may cache the data itself for other requests. In both [4] and [5] when the publishers, subscribers and types of events are large, the caching requirement of the nodes also becomes high.

III. NETWORK MODEL

In the home environment the appliances and sensors can be considered as a group depending on their location in the house, such as dining room, kitchen and bedroom, etc. In such an environment, a cluster-based network can be implemented with its advantages such as lower control overhead [1], [3].

A cluster-based ad hoc network is developed forming a backbone topology without loops (Fig. 1). Nodes in the network become a CH, a Gateway (GW), or an Ordinary Node (ON) after joining the network [6]. The following assumptions are made in the design of the network.

1. *Every node is able to measure the signal strength of a received transmission*: There exists a function in every node, which provides the routing layer, a numerical value of the reception signal strength.

2. *The signal strength between two stationary nodes are unaffected by external events*: Events such as opening and closing of doors or windows etc. in the physical path between the two nodes has insignificant or no effect on the propagation of wireless signals between the two nodes.

3. *All clusters are able to connect to one or more neighbor-ing clusters such that isolated clusters do not exist.*

4. *After the formation of the network, the network topology does not change as long as the structure of the house remains the same*: The clusters are formed depending on the location of the nodes. These locations are actual locations in the house based on the structure of the house. Thus the formed topology will remain the same as long as the structure of the house does not change.

A. Node Naming

A fixed name such as model etc. is given by the device manufacturer by default or it can be modified by user for better clarity. For example, *television*, *light*, *smoke sensor* etc are given as fixed names depending on the type or functionality of the appliance or sensor where duplication of the names is allowed. A few of the nodes in each location of the house are given names including their location in the home environment at the initiation of the network. The rest of the nodes or new comer nodes will discover its location name from its neighboring nodes (details are discussed in section 4.3).

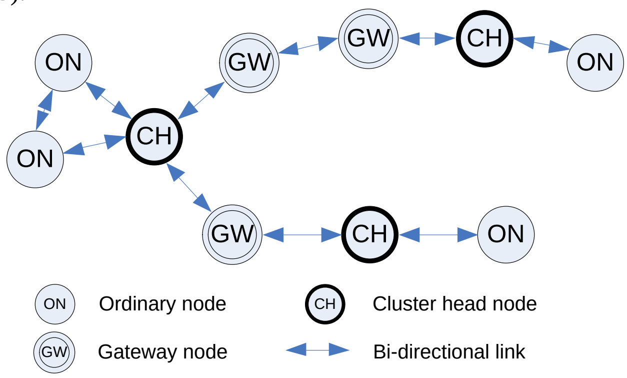

Fig. 1 Sample network

The location name is given to the nodes as a prefix to its fixed name such as; *kitchen oven, living room light, bedroom television* etc. This naming approach closely relates to *class-based identifiers* [7]. In this use of natural language, language, a location is identified from the beginning words and becomes more specific within that particular area by the next words. The location name should also be assigned as uniquely as possible to avoid message misdirection and delivery failures.

IV. CLUSTER ORGANIZATION

This section introduces the conceptual data structures and types of messages used in the paper. The location name discovery and formation of the clusters are also discussed.

A. Conceptual Data Structures

Three types of conceptual data structures are used to assist the formation of the clusters and the network as well as routing. The NT [1] is modified to include two additional fields i.e. *Priority* and the reception *Signal strength* of the neighbor's message. The format of NT is shown in Table I. Also the proposed algorithms require only CH and GW nodes to store their neighboring clusters in CAT [1]. The format is shown in Table II. The NT and CAT entries also have expiration times. The message arrival time is measured by a node's internal clock and kept as a *time stamp* with each entry. A memory cache organized using special methods called *Routing Cache* (RC) [8], [9] is used as a routing table. It holds the names of nodes that are beyond a node's neighboring clusters. Methods used to organize the RC also allow it to aggregate entries that have certain degree of similarity and it is used only by the nodes that participate in the routing. A RC is allocated to each bidirectional link (*Directional RC*) to a node participating in the routing to increase routing accuracy. The RC memory is proportionally divided according to the number of entries in each directional RC, termed *Proportionally-sized Directional RC* (*PDRC*) [9]. The directional RCs are stored in another conceptual data structure called *Routing Cache Table* (RCT) [9].

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

The RC is used here in dynamic network settings where it is used only by CH and GW nodes. A RC is allocated to each bidirectional link to a CH or GW node. The RCT will store the neighbor CH or GW node name (that has a directional RC for that direction) and the RC itself as shown in Table III.

The memory size used for RCT by the nodes is fixed before the network is setup as nodes contain limited memory [10]. During the network initiation, the process of allocating memory for each direction is performed dynamically but the total RC size is fixed. After initiation the RC would remain unchanged if the network topology also remains unchanged. It is also possible that the formed network may contain routing loops. Messages may be forwarded continuously without reaching the destination. The use of PDRC also prevents the formation of routing loops. Before inserting a candidate entry into any directional RC, all the directional RCs are searched for a match. If a match is found in any of the directional RCs the candidate entry is not inserted. This ensures that only a single path is available at a time between two nodes.

B. *Types of Messages*

There are 2 types of messages used in the network and they use multiple fields from Table IV. *Control* messages are used to configure the network and update the routing information of nodes. The formats of the control messages are given in Table V. For any control message, an empty "Dst" field allows all receiving nodes to process the control message.

After the formation of the network, *Data_1*, *Data_2* messages are sent appropriately. The Data_2 message contains additional fields to control its propagation. The destination node also sends an *Acknowledgement* (Ack) message back to the source node. For a data source/destination pair, each Data/Ack message also uses sequence numbers for identification. The format of the Data/Ack messages is given in Table VI.

C. *Location Name Discovery and Cluster Formation*

Two nodes are considered to be connected when they have a bi-directional link between them. Connected nodes can form clusters based on a common property. Nodes in the network have different capabilities and varying resources. If the lowest ID [11] is selected as the CH node, and the lower ID was assigned to a less resourceful node, it may restrict network performance. Other schemes such as "*highest connectivity*" [11], [12] may have the same effect. This paper uses an approach similar to the '*lowest ID cluster algorithm*' [11] to select the CH nodes. Rather than the *ID* [11] scheme the *priorities* of the nodes are used to select CH nodes. In this proposal the priority (*P*) for each node should be calculated by itself, based on its resources and capabilities. For example, the priority calculating function is defined as,

$$P = c \times m \times r\left(\frac{1 - 1/(f+1)}{1+d}\right) \qquad (1)$$

where c is the maximum processing capability, m is maximum memory capacity, r is maximum transmission power, f is full power capacity and d is the rate of power depletion at c, m and r.

Some nodes have direct power supply from a wall socket. The power available to such nodes is considered unlimited. If a node N is powered by an unlimited power source, its full power capacity is considered as infinite ($f \to \infty$) and there is no power depletion ($d = 0$). Hence the priority of N with

TABLE I - NEIGHBOR TABLE

Key	Description
Name*	Name of the node (including location)
Status	Status of the node (Undecided/ON/GW/CH)
Time stamp	Last message received time
Priority	Priority of the node
Signal strength	Received signal strength

TABLE II - CLUSTER ADJACENCY TABLE

Key	Description
CH name*	Neighbor CH node name
GW name	First hop GW node name to this cluster
Time stamp	Last Hello received time from first hop GW node

TABLE III - ROUTING CACHE TABLE

Key	Description
Name*	Name of first hop CH/GW node in the direction of RC
RC	Directional RC associated with the link
Number of entries	Number of entries in the RC

* - denotes the primary key

TABLE IV - FIELDS USED IN CONTROL/DATA MESSAGES

Field	Description
1. Nm	Name of the sender
2. Dsc	Type of message
3. Dst	Message destination node name
4. Stat	Status of the sender (Undecided/ON/GW/CH)
5. Prt	Priority of the sender node
6. CAT	CAT updates of the sender (CH, GW nodes)
7. RCT	RCT of the sender (CH,GW nodes)
8. SeqNo	Message sequence number
9. ReqCH	GW Requested CH name
10.PrNm	Next predicted name
11.PSnd	Previously forwarded node's name
12.TTL	Time to live – lifetime of a message in hops
13.NRcv	Next receiving node's name
14.Hops	Number of hops traveled
15.InitNm	Name of data sender without Nxt. receiver
16.STm	Sent time without Nxt. Receiver by InitNm
17.Data	Data carried by the message

TABLE V - CONTROL MESSAGE FORMAT

Message Name	Fields
Hello, Reply	Nm, Dsc, Dst, Stat, Prt
new_GW_Req	Nm, Dsc, Dst, ReqCH
new_GW_OK	Nm, Dsc, Dst, CAT, RCT
Update	Nm, Dsc, Stat, CAT, RCT
Block	Nm, Dsc, SeqNo

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

TABLE VI - DATA/ACK MESSAGE FORMAT

Message Name	Fields
Data_1	Nm, PrNm, SeqNo, Dsc, Dst, PSnd, TTL, NRcv, Data
Data_2	Nm, PrNm, SeqNo, Dsc, Dst, PSnd, TTL, Hops, InitNm, STm, Data
Ack	Nm, PrNm, SeqNo, Dsc, Dst, PSnd, TTL, NRcv

Algorithm-1

1. Discover neighbors using Hello and Reply messages.
2. If there are no CH neighbors and this node has the highest priority of all Undecided neighbors, calculate location name with (2) and become CH.
3. If there are CH neighbors, set location name prefix as same as the CH with the highest signal strength and become an ON of that cluster.

Algorithm-2

1. If an ON node receives a Hello message from another CH node other that its home CH node, and there are no GW nodes between the two CH nodes, become a GW node.
2. If an ON receives a Hello message from another ON from a different cluster and if there are no GW nodes between the two clusters, both ON nodes become GW nodes.
3. Send Update message if CH/GW node add/remove CAT entries or add RCT entries or if the status of the node is changed.
4. Receive Update messages and update NT CAT and RCT

unlimited power is calculated from (1) as,

$$P^N_{Unlimited} = c \times m \times r \, .$$

If N is powered by a limited power source such as a battery, both f and d will have positive values (f, $d > 0$) and both values are finite. Hence the following portion of (1) will always have a value less than one.

$$0 < f, d < \infty \;=>\; \left(\frac{1 - 1/(f+1)}{1+d} \right) < 1 \, .$$

Hence, $P^N_{Limited} < P^N_{Unlimited}$; where $P^N_{Limited}$ is the priority of N powered by a limited power source. This priority calculating function common to all nodes will assure that highest resourceful nodes become CH nodes.

At the initiation stage, the user will give a location name to at least one node in each of the locations. This location name assignment is used as a trigger to start the cluster formation process. Nodes initially joining the network whose status in the network is yet unknown are *Undecided* nodes. An Undecided node N joining the network, which is unaware of its location (i.e. it does not have a location name), will wait without sending any messages. When the user sets the location name of N or if N receives a *Hello* message from a node with a location name, N starts sending periodical Hello messages (The term "*hello_timer*" is used to specify the time duration between two consecutive Hello messages). All nodes with or without any location name that received this message send a *Reply* message back to N. N will use the information in the Reply message to insert an entry into its NT. N will collect any Reply messages for a certain period (The term "*initiation_timer*" is used to specify the duration a node waits for any Reply messages). After this period if N has the highest priority of all Undecided neighbors, N calculates the total received signal strength for each received location name and chooses the location name with the highest total signal strength as its own location name (The term "*network location*" is used to identify a group of nodes that have the same location name). For example, consider that N receives reply messages from nodes in x number of different network locations. For each network location L_i (i=1,2,3,…,x), N receives reply messages from n_i number of nodes with signal strength s_{ij} (j=1,2,3,…,n_i). For each network location L_i calculate the total signal strength as,

$$S_{tot}(L_i) = \sum_{j=1}^{n_i} (s_{ij}) \, . \qquad (2)$$

Choose the network location with the maximum total signal strength $L_{\max} = \max[S_{tot}(L_i)]$ as N's own network location and set it as a prefix to its name to complete the naming process. Afterwards N sends a Hello message as a CH. Nodes that doesn't have the highest priority in its neighborhood will wait for a Hello message from a CH node.

In addition if N joins the network after the cluster formation and receives reply messages from several CH nodes, N also chooses to name itself in the same network location as the CH with the highest reception signal strength and become an ON of that cluster. Node N will remain as an Undecided node until at least one bidirectionally linked neighbor is found. This procedure is terminated after a node changes its status from Undecided to CH or ON. Algorithm-1 summarizes the cluster formation and naming of the nodes.

V. NETWORK ORGANIZATION

This section introduces the formation of the network with the discovery of adjacent clusters. Also the management of mobile nodes and how they share location names during mobility are also described.

A. Network Formation

After the initial formation of the individual clusters the CH nodes need to discover the neighboring CH nodes in order to form a network. Two neighboring clusters can establish a bidirectional connection through one or two ON nodes. These ONs will become GWs when the two clusters are connected. This paper proposes the use of the difference in location names to identify each cluster separately. Any node from one cluster is able to identify whether another node is from the same cluster or not, based on the location name. This feature is used to form GW nodes and hence form the network using Algorith-2. An ON (N) may hear a message from another CH node (CH_2) which is not its *home CH* node. First N will try to establish a bidirectional link with CH_2. If N succeeds, it sends a *new_GW_Req* message to its home CH. If the home CH does not already contain a GW for CH_2, it sends a *new_GW_OK* message back to N and N becomes a GW and sends an *Update* message. CH_2 also receives this Update message and any new information is inserted to NT, CAT and RCT. Here the two CH nodes are two hops away from each other. N may also find out about CH_2 from another ON (N_2) which is a member of the cluster of CH_2. Here N will try to establish a connection with N_2. If N is successful, the above process is repeated and both N and N_2 become GWs. Here the two CHs are three hops away from each other.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

When GW or CH nodes add/remove CAT entries or add RCT entries, the next periodically sent Hello message is replaced by an Update message. Depending on which data structure changed, the CAT and/or RCT fields may be omitted from the Update message. The receivers update their CAT, NT and RCT and if any information is changed; it creates a new Update messages that is sent to its neighbors instead of the next periodical Hello message. This process propagates the new routing information and the entire network is updated. This propagation of new cluster information only occurs at the initiation of the network. The movement of a GW or CH node may also cause some Update messages to be sent during the reselection of new GW/CH node which will only cause the neighbor GW/CH nodes to send Update messages.

B. Mobility management

Mobile nodes move randomly within the network. At the initiation of the network some highly resourceful mobile nodes may be selected as CH nodes. If the mobile CH moves away from the current network location, another CH needs to be selected. The mobile CH may move into another network location where a CH already exists. Changing the CH nodes often creates many control messages and data communication may also be interrupted. Therefore CH nodes should change as less frequently as possible. Some existing rules for CH mobility [1], [13] are summarized in Table VII. The modified rules in Table VII are implemented in Algorithm-3.

When the status of a node changes often, control messages are generated to notify its neighbors of the new status. In order to minimize the number of control messages, the *Rule 4* [1], [13] in Table VII is changed to minimize the number of status changes of a node. If a node N is moving away from its home cluster, but it is still connected with at least a GW node from either its previous cluster or any new cluster, N forwards any messages to this GW node. But N will not change its status. Even though N does not have a bidirectional link with a CH, it is preferable to keep the status of N as ON to lower the control message overhead. This is implemented in Algorthm-3. If a GW is mobile and a replacement node is unavailable to take the role as GW between the two clusters, the RCT also becomes invalid. The RCT can be rebuilt by requesting the RCTs of the neighbor GW/CH nodes periodically or in a reactive manner when the loss of a GW node is detected.

1) Location name sharing during mobility

Any node in the network is allowed to move regardless of their status, (even nodes whose location name was assigned manually). The following assumptions are made for location name sharing among mobile nodes.

1: *A network location (L) must contain some nodes including a CH node at all times.* This assumption allows the preservation of the network location name. Even if all nodes in L are mobile, while they are in L they will have the location name given to L. A CH node is also selected from the available mobile nodes.

2: *There are enough connected nodes in L such that the new CH node will still have the same network location name.* Even if the mobile CH node moves out of L, the remaining ON nodes still have the network location name and they will repeat algorithm-1 to choose another CH. After loosing the CH node, if at least two nodes in L are unable to connect to each other, the location name L will be lost. Considering the size of a household location and wireless range of nodes, it is assumed that such a situation is avoided.

TABLE VII - STATUS CHANGING RULES

Rule	Description
1.	A non-CH node x does not challenge an existing CH node y to become the new CH even if x has a higher priority than y and they have a bidirectional link to each other.
2.	When two cluster heads become neighbors for more than a certain duration, the lower priority CH will loose its status as CH and become an ON of the other CH.
3.	When an ON node moves from one cluster to another the status of either CH will not change.
4.	When an ON (N) moves out of its cluster and does not enter into another, N becomes a new CH creating a new cluster.

Algorithm-3

1. If a non-CH x has a bidirectional link to an existing CH y, then x becomes an ON of y.
2. If two CHs have a bidirectional link between them, for more than "CH_wait_timer" duration, use (2) to recalculate their location names and the lower priority CH becomes an ON of the other CH. (CH_wait_timer is the time duration two CH nodes are allowed to have a bidirectional link without changing their status)
3. If a non-CH node N moves out from its home CH and has a connection with a new CH/GW node in another network location L, N changes its name (location name prefix) as same as L. Also if N is a GW it will change its status to ON. If N is an ON its status will remain unchanged.
4. If N is unable to find a connection with a CH or GW node, N becomes an Undecided node and repeats Algorithm-1.

The following strategies 1 and 2 are defined for location name sharing for mobile nodes.

1. *When two CH nodes become bi-directionally connected for more than CH_wait_timer duration, they use (2) to recalculate their location names.*

If a CH_1 is mobile, it gradually moves from the current network location to the next while loosing neighbors from the current network location and adding new neighbors from the next. Before CH_1 challenges CH_2 in the new network location, the majority of CH_1's neighbors would be from the new network location. But only when a bidirectional link is setup between CH_1 and CH_2, CH_1 realizes that it is in a new network location. Both CHs use (2) to recalculate their new network location name. CH_2 will not change its name as its neighbors are still intact. CH_1 changes it's network location name same as CH_2 as most of its neighbors are from CH_2. Therefore even if the role of CH was handed over to the newly arriving CH_1 the entire network location name does not change. When CH_1 and CH_2 are not mobile and even if the majority of neighbors of CH_1 become nodes from CH_2 due to neighbor mobility, CH_1 will not change its network location name because CH_1 and CH_2 are not directly linked. This is implemented in Algorithm-3.

2. *When a non-CH node N moves from one cluster to another, N's location name changes according to its new network location.*

If N has a connection with a GW node from the same cluster, N keeps its network location name intact.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

If N is only able to find a bidirectionally linked GW_2 in another cluster (CH_2) (this implies that N has either arrived in a new cluster or is in the border), N will change its location name as same as GW_2's location name and forward any messages to GW_2. This is also implemented in Algorithm-3.

VI. ROUTING PROTOCOL WITH MOBILITY SUPPORT

This section introduces the routing protocol which includes route discovery, maintenance and mobility support features. Three options used to handle a mobile node's transition from one cluster to another are also described.

A. *Route discovery*

After the network is initialized i.e. the NT, CAT and RCT are populated, Data_1 messages can be sent without any route discovery procedures. For intra-cluster communication, nodes search for the entire destination name in their NT. However all ON nodes in one cluster may not be reachable from every other ON node. In this case the ON node forwards the message to its home CH node. Similarly during inter-cluster communication, the complete routing information about the destination may or may not be stored in NT, CAT or RCT of the sender node. If an ON (only has NT) or GW (has NT, CAT and RCT) node is unable to find a routing mach in its internal data structures, it forwards the message to its home CH node. If the destination is known (i.e. the node has information about the destination in its NT, CAT or RCT) the node forwards the message appropriately by specifying the next hop receiver (NRcv) node name [9], [14]. If the home CH or another CH or GW node in the forwarding path is unable to find a routing match for the destination, the message is dropped.

B. *Route maintenance*

During data communication regardless of mobility, nodes that receive the Data_1 or Ack messages directly from the source or destination nodes will continue to send Hello messages and *refresh* their routing information i.e. NT and CAT entries will also expire periodically. This will allow the mobile nodes to recognize its whereabouts accurately and identify their location name or status. The CH and GW nodes in the forwarding path identify the previous sender node from the PSnd field of the Data_1/Ack message. This will enable the nodes to update the time stamp of NT entries from the Data_1/Ack messages they receive from forwarding neighbors in order to maintain an accurate forwarding path. Nodes that do not hear any Data_1 or Ack messages directly from the source or destination for a certain period of time will stop sending Hello messages and their NT and CAT entries will not expire. A period of time a node remains in this state is referred as *standby period*.

When communication is required between two nodes, the sender does not consider if the routing information in the path to the destination is up-to-date or not. The Data_1 message is forwarded by intermediate nodes according to the available routing information. But the routing information may not be up-to-date. Therefore the Data_1 message will be forwarded to the last known network location of the destination in the network. When the Data_1 message is received by a node (N) that has the destination name in its NT, N will forward the message and wait for an Ack message from the destination node. If the destination node is still in the last known location, it will receive the message and send an Ack message back to N. Receiving the Ack message N will realize that the destination node is within its reach and the Ack message is forwarded back to the sender and also the NT time stamps of GW/CH nodes of the forwarding path becomes renewed by using PSnd field of the Data_1/Ack messages.

C. *Mobility support*

It is also possible that the destination node may have moved during the standby period and it may not be reachable by N. As before N will forward the Data message to the destination and wait for an Ack message. (The term *Ack_wait_timer* is used to specify the period of time that a last forwarding node wait for an Ack message). If the Ack message is not received after an Ack_wait_timer period of time, N will send a Data_2 message (derived from the previous Data_1 message) allowing any receiving GW or CH node to forward the Data_2 message. The GW or CH nodes will wait for a certain period of time before forwarding this Data_2 message (The term

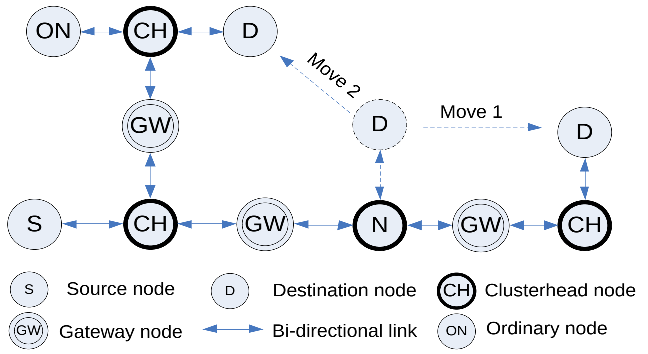

Fig. 2 Example destination movements

Block_wait_timer is used to specify the time period that a GW/CH node wait before forwarding a Data_2 message). The Data_2 messages record the number of hops it has traveled and the time it was sent. The Block_wait_timer duration is specified as "2 $\times$ number of hops of the forwarded Data message $\times$ time for one hop" [15]. This allows a GW/CH node to independently calculate the Block_wait_timer duration. The delay introduced by the Block_wait_timer can be considered as minimal as the maximum number of hops in a home environment is not very large. The forwarded Data_2 messages by neighbors of N are also received by N as well. This will allow N to passively renew the time stamps for GW/CH entries already in NT and establish new links with any new GW/CH nodes that are in the bidirectional communication range.

When the destination node receives the Data-2 message, it will send Ack messages back to the sender as well as N. When the Ack message is received by the source node the routing information for the forwarding path is also updated by the forwarded Ack message, and communication can continue. After N receives the Ack message it will send a Block message which is forwarded by GW or CH nodes if their Block_wait_timer for the received "Nm, Dst, SeqNo" triplet has expired. The Block_wait_timer will allow the waiting GW or CH nodes to receive the Block message before the Data_2 message is forwarded.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

When the Block message is received by a waiting GW or CH node, it will discard both Data_2 and Block messages.

For example the movement of the destination may have created a path between the source and destination which also include N as a forwarding intermediate node (Fig.2 – Move 1). Here N will forward the Ack message to the sender as well as send a Block message. If the new path between the sender and receiver does not include N (Fig. 2 – Move 2) as a forwarding node, N will only send a Block message. Here also N will try to forward the Ack message to the sender node. But the calculated next receiver (NRcv) node to forward the Ack message will be the same as the previous sender node (PSnd) because the network has backbone topology without loops. Therefore N will discard the Ack

Algorithm-4

SENDER:

1: Send Data_1 message and Wait for Ack message

2: If Ack message is received continue to send Data_1 messages

3: If Ack is not received within Ack_wait_timer duration, Data_2 message is sent to the destination.

4: If an Ack message is not received within the second Ack_wait_timer duration the destination is not reachable.

INTERMEDIATE NODE:

1: Process Data_1 message if this node is specified in the message as the next hop receiver

2: If the destination of a received Data_1 message is in NT, forward Data message to destination and start Ack_wait_timer.

3: If the destination of a received Data_1 message is in CAT/RCT, forward message to next hop receiver node.

4: If Data_2 message is received and it has been forwarded previously, update the NT entry of the previous sender (passive update). If the Data_2 message has not been forwarded before, start Block_wait_timer.

5: If Ack_wait_timer expired then Send Data_2 message and mark this node as Initiator.

6: If Block_wait_timer expired then Forward Data_2 message.

7: If an Ack message is received forward it to the data sender.

8: If an Ack message is received and this is the Initiator node, try to forward Ack message to data sender and send a Block message. If this is not the Initiator node, forward Ack message to data sender or Initiator.

9: When a Block message is received and the Block_wait_timer is expired then forward the Block message. If Block_wait_timer is not expired, stop Block_wait_timer and discard Data_2, Block messages.

RECEIVER:

1: If Data_1 message is received, send Ack message back to data sender.

2: If Data_2 message is received, send Ack message back to sender and Initiator.

message. If a sender node is unable to receive an Ack message within two consecutive Ack_wait_timer periods, the destination is deemed unreachable. This process is summarized in Algorithm-4.

During the communicating node's movement, the transition from one network location to another is the most critical period where message loss can occur. The following 3 routing options are proposed to handle the name changes during mobility.

Option-1: This option disregards the transition period. A source node's name change is notified to the destination by the first Data message sent after the name change. Similarly a destination name change is notified to the source node by the first Ack message sent after the name change. Here the source/destination will not immediately realize that the other party has changed its name and some messages sent with the previous name might be lost.

Option-2: This option considers the transition time before the mobile node changes its name in order to predict its new name. Some existing methods for ad hoc networks [16], [17] require GPS data and other measurements such as speed and direction in order to predict the network behavior. But this proposed prediction method only uses a node's neighborhood location names and reception signal strength. During the transition from one network location to another the mobile node will also gradually add neighbors from the next network location. The mobile node will calculate the total signal strength of each group of neighbors in NT from different network locations using (2). The neighbor group with the 2nd highest total signal strength is predicted as the next network location of the mobile node. The predicted next network location is calculated by both sender and receiver and sent in the Data and Ack messages. If a node in the communication path is unable to find the destination name of the message, it will also look for the predicted name in its NT, CAT or RCT and if found, forwards the message accordingly. However if the calculated 2nd highest total signal strength is not the next network location of the mobile node, message loss will occur.

Option-3: To achieve higher message delivery success regardless of the next network location change, a network location wide forwarding is performed at the source and destination network locations. Any GW/CH node in the same cluster as the source or destination will forward the message. A forwarding GW/CH (N) can recognize if the destination is in the same network location by comparing N's own network location name with that of the received message destination. The GW nodes cover the border between two adjacent clusters where the network location transition also occurs. The forwarding of the Data message by GW nodes also enables the mobile node to receive the message from any border area between the current network location and any adjacent network location.

VII. EVALUATION

A simulation program was developed to evaluate the proposed routing mechanisms and the effect on overhead in the clustering after modification of rule 4 in Table VII. The simulation environment comprised of 100 nodes located in a flat surface of 100×100 units. Each node is given random transmission distances (1, 2, 3 units) and priorities (1~10). Nodes are placed randomly in the simulation environment for each simulation such that random topologies are formed with random number of clusters and random number of nodes in each cluster to represent different households. The location names are also given to a random number of nodes (1~5) at each cluster to represent the user's involvement in location naming process.

For evaluation, mobile nodes were chosen as both the source and destination and they are paired randomly. All mobile source nodes send a Data_1 message every *time tick* (TT) of the simulation clock. This is to allow mobile nodes to send and receive Data even during transition between clusters.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

The simulation duration is 1000 TTs. The mobile nodes move at random rates between 0.1~1 unit per 1TT(If the maximum walking speed is considered as 1m/s, the mobile speed range can be interpreted as 0.1~1m/s). A RCT size of 128 bytes was chosen as it was shown to be sufficient to handle a network size of 100 nodes [10]. Also a unique name for each network location is used. The three routing options are simulated in multiple topologically random networks and the average of the results is presented.

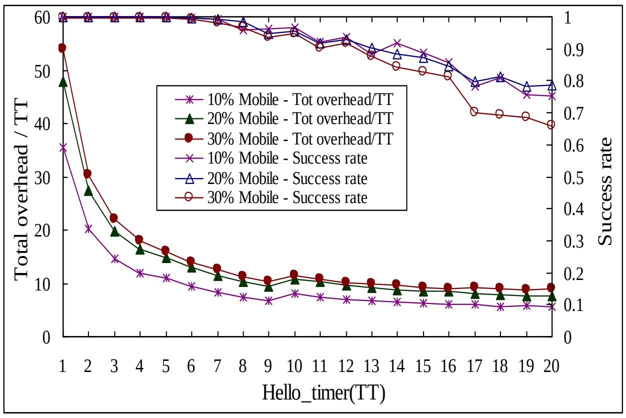

Fig 3. Overhead/TT and Success rate for varying Hello_timer values when refresh rate is 10TT.

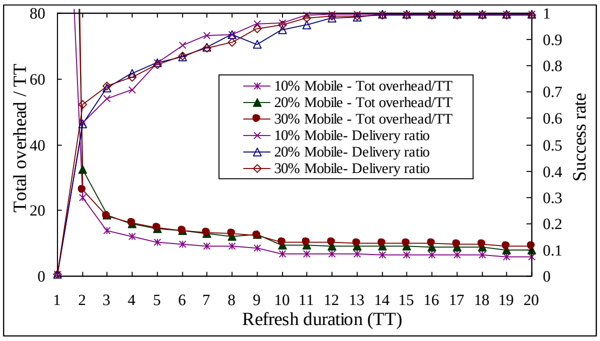

Fig 4. Overhead/TT and Success rate for varying refresh rate values when Hello_timer is 9TT.

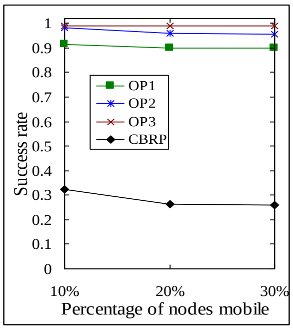

Fig. 5 Percentage of mobile nodes Vs Success rate*

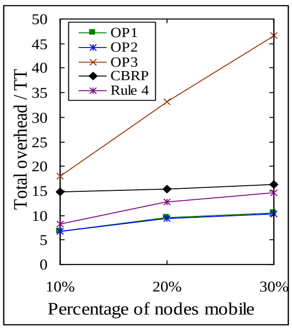

Fig. 6 Percentage of mobile nodes Vs Overhead/TT*

* Refresh rate is 10TT and hello_timer is 9TT

A simulation was conducted to find suitable values for the validity period of NT, CAT entries (*Refresh rate*) and hello_timer duration by varying both values and monitoring the success rate and the overhead per TT. Fig. 3, Fig. 4 shows an instance of this simulation when option-2 was used. Fig.3 shows the success and overhead for varying hello_timer values, when the refresh rate is 10TTs. Fig. 4 shows success and overhead for varying refresh rates when the hello_timer 9TTs. Fig. 3 shows high success rate as well as high overhead for low hello_timer rates. Fig. 4 shows higher success rates for longer refresh durations. When the percentage of mobile nodes was increased, similar success rates were observed with increased overhead. The hello_timer and refresh rates must be chosen such that it gives the highest success rate with the least overhead.

Fig. 5 and Fig. 6 show the observed success rates and the overhead of the 3 routing options when refresh rate is 10TTs and hello_timer is 9TTs. Both CBRP [1] as well as the 3 routing options use 3 conceptual data structures of which two are the same i.e. NT and CAT. Only three data structures are used for the routing purpose (NT, CAT and two-hop topology database for CBRP, RCT for the 3 options) for comparability on equal grounds. Option-3 shows the highest success rate but also it has a high overhead. The overhead calculated for option-3 also includes the data messages broadcast by the CH and GW nodes that are not received by the intended destination. The options-2 shows the least overhead but also with a higher success rate than option-1. All 3 options showed higher success rates compared with CBRP on equal terms. The observed overhead for all options are based on the modified Rule 4 from Table VII implemented in Algorithm-3. When the unmodified Rule 4 was used with Option-2 in Algorithm-3, an increase in the overhead was observed as shown in category "Rule 4" in Fig. 6.

VIII. CONCLUSION

This article presented a clustering scheme for mobile ad hoc networks with name-based addressing. Methods to handle mobile nodes effectively were also discussed using the name-based addressing scheme while minimizing control overhead. Three routing options that can be used for communication among the nodes were also presented. A performance comparison with equal resources against CBRP showed higher success rates by all 3 routing options. Option-3 shows the best success rate but also with the highest overhead. From the 3 options discussed here, option-2 is the most preferable solution in a home environment with resource constrained sensor nodes as it shows a high success rate with the least overhead.

REFERENCES

[1] Jiang M.; et al; Cluster based routing protocol, (CBRP). draft-ietf-manet-cbrp-spec-01.txt, Internet draft, August 1999.

[2] Johnson, D.B.;, "Routing in ad hoc networks of mobile hosts," *Mobile Computing Systems and Applications* (WMCSA), pp.158-163, 8-9 Dec. 1994.

[3] P. Krishna, N.H. Vaidya, M. Chatterjee and D.K. Pradhan; "A cluster-based approach for routing in dynamic networks". ACM SIGCOMM Computer Communication Review 27:49-65, 1997.

[4] Petrovic, M.; Muthusamy, V.; Jacobsen, H.-A., "Content-based routing in mobile ad hoc networks," *Mobile and Ubiquitous Systems: Networking and Services* (MobiQuitous), The Second Annual International Conference on, pp.45,55, 17-21 July 2005.

[5] M. Meisel, V. Pappas and L. Zhang; "Ad hoc networking via named-data" MobiArch '10, Proc. of the 5th ACM intl. workshop, pp 3-8, 978-1-4503-0143-5.

[6] M. Gerla, J. Tzu-Chieh Tsai,; "Multicluster, mobile, multimedia radio network". *ACM- Baltzer Journal of Wireless Networks*, 1: 255-265, 1995.

[7] Shah, R.C.; Rabaey, J.M., "Energy aware routing for low energy ad hoc sensor networks," *Wireless Communications and Networking Conference* (WCNC), IEEE , vol.1, no., pp.350,355 vol.1, 17-21 Mar 2002.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

[8] Vidanagama, T.N.; Nakazato, H.; "Description based routing and addressing in cluster based ad hoc network" *IEICE General Conference*, Mar 2009.
[9] Vidanagama, T.N.; Nakazato, H., "Reduction of Message Misdirection in Description Based Clustered Ad Hoc Networks," *Consumer Communications and Networking Conference* (CCNC), 7th IEEE, vol., no., pp.1-5, 9-12 Jan. 2010.
[10] Vidanagama, T.N.; Nakazato, H., "Name-based message forwarding for home ad hoc networks", submitted for publication.
[11] A. Ephremides, J.E. Wieselthier, D.J. Baker,; A design concept for reliable mobile radio networks with frequency hopping signaling, in Proc. of IEEE, 75(1) (1987), pp.56~73.
[12] A. K. Prakah, "Selecting routers in ad-hoc wireless networks, Proc. of SBT/IEEE ITS (1994).
[13] C. -C. Chiang, H. K. Wu, W. Liu, and M. Gerla, "Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel," Proc. of IEEE SICON'97, pp. 197-211, April 1997.
[14] Vidanagama, T.N.; Nakazato, H., "Mobility in a description based clustered ad hoc network," *GLOBECOM Workshops* (GC Wkshps), 2010 IEEE, vol., no., pp.148-152, 6-10 Dec. 2010.
[15] Mznah A. Al-Rodhaan, L. Mackenzie, M. Ould-Khaoua, "Improvement to Blocking Expanding Ring Search for MANETs*", DCS Technical Report Series*, Dept. of Computing Science, University of Glasgow, 2008.
[16] Su, W., Lee, S.-J. and Gerla, M. (2001), "Mobility prediction and routing in ad hoc wireless networks". *Int. J. Network Mgmt*., 11: 3–30. doi: 10.1002/nem.386.
[17] William Su; Gerla, M., "IPv6 flow handoff in ad hoc wireless networks using mobility prediction," *Global Telecommunications Conference* '99, vol.1A, no., pp.271-275 vol.1a, 1999.

V.G. Tharinda Nishantha Vidanagama received his B.Sc. with first-class honors from Peradeniya University, Sri Lanka in 2003. He received M.S. in Computer Systems and Network Engineering from Waseda University in 2010 where he is currently pursuing his PhD. His research interests include name-based ad hoc networks, multicast, mobility and content centric networks.

Hidenori Nakazato received his B. Engineering degree in electronics and telecommunications from Waseda University in 1982 and his MS and Ph.D. degrees in computer science from University of Illinois in 1989 and 1993, respectively. He was with Oki Electric from 1982 to 2000. Since 2000, he has been a faculty member of Graduate School of Global Information and Telecommunications Studies, Waseda University. His research interests include performance issues in distributed systems and networks.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)