A Fuzzy-Based Power-Aware Management for

Mobile Ad Hoc Networks

Kuo-Qin Yan¹, Shu-Ching Wang¹*, Mao-Lun Chiang²

¹ Chaoyang University of Technology, Taiwan, R.O.C.,
² National Chung Hsing University, Taiwan, R.O.C.,
*Corresponing E-Mail: scwang@cyut.edu.tw

ABSTRACT

In recent years, people have become more dependent on wireless network services to obtain the latest information at any time anywhere. Wireless networks must effectively allow several types of mobile devices send data to one another. The Mobile Ad Hoc Network (MANET) is one important type of non-infrastructure mobile network that consists of many mobile hosts, usually cellular phones. The power consumption rate and bandwidth of each mobile host device becomes an important issue and needs to be addressed. For increasing the reliability of the manager in Hierarchical Cellular Based Management (HCBM), this paper proposed a power-aware protocol to selecte a stable manager from mobile hosts by fuzzy based inference systems based on the factors of speed, battery power and location. Further, our protocol can trigger a mobile agent to distribute the managerial workload.

Keywords: Mobile Ad Hoc Network, Non-infrastructure mobile network, Hierarchical Cellular Based Management, Mobile Agent

1. The background and motivation

In general, the mobile host (MH or node) of a traditional fixed wireless network is built by specific intermediaries (ex: mobile station) to communicate each other. However, the infrastructure is easy to be destroyed by external environment, such as natural disasters and wars. Therefore, the Mobile Ad Hoc Network (MANET) is more flexible due to absence of a fixed infrastructure. A MANET consists of mobile hosts that can flexibly and quickly obtain the latest location information for automated battlefield, disaster relief, and rescue situations.
There exist several challenges to the MANET due to its dynamic nature, such as a limited bandwidth, battery power, and communication routing [13,16,18-21]. As mentioned, an efficient routing among a set of MHs is one of the most critical issues in MANET. Therefore, the traditional routing protocols [2,4] focusing on the aspect includes the shortest path and cluster methods. The concept of the shortest path method is using the least MHs to forward messages. However, it is difficult to establish co-coordination due to the MHs is usually working independently in MANET. In cluster method, the cluster head (manager) can be elected to manage and forward message for MHs within a specific range. Only manager needs to keep the routing table and other MHs can save the batter power in MANET. Therefore, this paper uses popular cluster method, the Hierarchical Cellular-Based Management (HCBM) is proposed by Chang et al. [2] to assign the manager in MANET.

Unfortunately, the topology of MANET may be destroyed due to the manager exhausts its limited energy and bandwidth when overloaded with packets. For prolong the lifetime of manager, the power-aware manager is invoked by introducing a fuzzy theory inference system [6,25] into the HCBM method. A multi-Mobile Agent (multi-MA) is proposed in this protocol to assist the manager to manage the specific MHs when the workload is overloaded.

The rest of this paper is organized as follows. Section 2 illustrates the relevance of the previous work. The details of methodology are shown as Section 3. Section 4 illustrates the results of simulation. The conclusion is presented in Section 5.

2. Previous work

The traditional routing protocols can be divided into two kinds, table and demand driven [4]. In table driven, each MH has a routing table to route the message. In contrast, the MH in demand driven must find the routing path every time.

The advantage of the table driven routing protocol is that communications can be set up quickly when the source MH wants to send the message to destination MH. This is because each MH must periodically update the latest routing information into the routing table. All MHs must update their routing tables when the MH roams around in MANET. Therefore, this method has additional overhead in network, such as the network congestion and occupies much memory. The related literatures include “Destination-Sequenced Distance-Vector routing (DSDV)”, “Clusterhead Gateway Switch Routing (CGSR)”, and “Wireless Routing Protocol (WRP)” [17]. Besides, the
“Active Route-Maintenance Protocol (ARMP)” [22] is proposed by Tu et al. to periodically detect the connection node status.

On the other hand, the advantage of the demand driven is each MH does not need to note down the routing information in daily work. However, the source MH using the demand driven routing protocol needs to waste more time to set up the routing path. The related researches include “Dynamic Source Routing (DSR)”, “Ad Hoc On demand Distance Vector routing (AODV)” [4,15], “Temporally Ordered Routing Algorithm (TORA)” [4,14], and “Associative Based Routing (ABR)” [4].

The routing protocols mentioned above have respective advantages and drawbacks, thus some researchers proposed a hybrid method. The hybrid methods include the “Zone Routing Protocol (ZRP), “GRID” [4,10], and “Hierarchical Cellular Based Management (HCBM) [4]. These hybrid methods can improve the drawbacks of the table driven and demand driven methods to reduce the broadcast storm [12]. A comparison of the routing protocols is shown in Table 1.

Table 1. The comparison of different routing protocols

<table>
<thead>
<tr>
<th>Items</th>
<th>Routing Protocol</th>
<th>Table driven</th>
<th>Demand driven</th>
<th>Hybrid</th>
</tr>
</thead>
<tbody>
<tr>
<td>The speed of setting up the connection</td>
<td>Fast</td>
<td>Slow</td>
<td>Medium</td>
<td></td>
</tr>
<tr>
<td>Save the power</td>
<td>Worst</td>
<td>Better</td>
<td>Medium</td>
<td></td>
</tr>
<tr>
<td>The loading of routing maintenance</td>
<td>Heavy</td>
<td>No</td>
<td>Medium</td>
<td></td>
</tr>
<tr>
<td>The performance in the large topology</td>
<td>Worst</td>
<td>Worst</td>
<td>Better</td>
<td></td>
</tr>
<tr>
<td>The performance in the small topology</td>
<td>Better</td>
<td>Better</td>
<td>Worst</td>
<td></td>
</tr>
<tr>
<td>Related topologies</td>
<td>DSDV, WRP, CGST, etc.</td>
<td>ABR, TORA, AODV, etc.</td>
<td>ZRP, GRID, HCBM, etc.</td>
<td></td>
</tr>
</tbody>
</table>

Other than table and demand driven method, other choices include: (1) the flat versus hierarchical and (2)GPS-based [27] versus non-GPS-based. In a flat routing method [4], all MHs are treated equally, thus any MH can be used to forward packets between arbitrary source and destination MHs. For permitting scaling, the hierarchical methods [4,10,22] are usually to be used in MANET. The major advantage of hierarchical routing method is that the routing process is simplified to managers.

In GPS-based routing scheme, it can provide the latest position information such as the label, time of receipt, status, direction of speed, date of receipt, and longitude and
latitude per second. Namely, each MH can know its physical location by GPS device. In contrast, each MH cannot know its location and position when the GPS device absence.

For scalability and flexibility, the HCBM and GPS device are used as our backbone to adapt to the dynamic of MANET. HCBM has three critical advantages as follows:

- HCBM can reduce the routing cost similar to that in GRID [1,10]. This is because we can search for MH within a specific range instead of a global search. Therefore, we can more easily manage our MH than other methods.
- The distances among the managers in Fig.1(a) are equivalent in the HCBM structure. In contrast, the distances among the GRID structure managers are different as shown in Fig.1(b). Each MH has the same transmission power strength in the HCBM structure.

![Fig. 1. The strength of transmission power between HCBM and GRID structures](image)

- The source MH can use fewer numbers of managers to forward messages to destination MH. This is because the signals in the HCBM structure have greater coverage than those in the GRID structure even though the radii are the same. HCBM can also increase the QoS via hierarchical management.

In this paper, an inference system can be used to elect and prolong the lifetime of manager which located in the center of the intra-cell by filtering the factors, such as distance, the average roaming speed and battery power. Further, a multi-Mobile Agent (multi-MA) fuzzy engine is proposed to consider the battery power and bandwidth factors in this paper. This fuzzy technology of our proposed protocol can reduce the Ping Pong Effect [24] among the threshold by triggering the multi-MA appropriately. The manager assignment and fuzzy engine methods are described as next section.

### 3. Methodology

There are two phases in the proposed method, the Power-aware Manager (PM)


election phase and Power-aware multi-MA (PMA) assignation phase. The main job of the PM election phase is electing the appropriate manager for MANET when a management overload exists. The power-aware multi-MA is elected by the PMA assignation phase to divide the management workload. The assumptions and details of these phases will be described as follows:

- **H1**: Each MH has a unique id.
- **H2**: Each MH has the same transmission radius.
- **H3**: Each MH has the same signal power.
- **H4**: The communication area can be divided into the cellular (cell) by GPS.
- **H5**: MHs can periodical receive the current longitude, latitude, and speed from GPS.
- **H6**: Each MH was distributed equally.
- **H7**: Any MH can ensure the data transmission and receipt successfully.

### 3.1 The Power-aware Manager (PM) election phase

This phase is divided into three parts. First, the GPS receiver information is used to select a manager for each cell. Subsequently, a new method is given in which the distance, average roaming speed, and battery power are considered. Finally, the Power-aware Assignation Inference System (PAIS) is induced to improve the election phase to select an appropriate manager.

The National Marine Electronics Association 0183 (NMEA-0183) standard was used to set up the GPS device. The GPS data transmission device uses the Global Position Recommended Minimum Specific (GPRMC), which is the minimum format to output the ASCII code. This format provides the latest position information such as the label, time of receipt, status, direction of speed, date of receipt, and longitude and latitude per second [27].

The HCBM is derived from the cluster concept. Hsieh et al. [1,2,10] deemed that a MH in cell center should be selected as cell manager according to the GPS information. This is because a MH in the cell center can roam around a larger space than the other MHs. However, a manager in the cell center may have lower battery power, making it easy for the MH to exhaust its battery power. A MH with lower battery power usually needs to execute the handoff procedure when the manager leaves its section or exhausts its battery power. Therefore, the MH should avoid frequent handoff procedures and maintain good transmission. To ensure manager capability, the selected manager must have the appropriate location, roaming speed, and battery power.
in the election phase. This paper proposes two steps to elect the power-aware manager as follows.

- **Step1**: The cell is divided into several ranks and the MH is selected in center of the cell according to the GPS information.
- **Step2**: The average roaming speed and battery power are computed in all MHs to elect the appropriate manager.

The location of each MH is compared to determine which MH can become the manager. However, this action of comparing all MHs will increase the load on the entire network. To reduce the network load the intra-cell is divided into three filter levels in place of the traditional HCBM method, as shown in Fig. 2.

![Fig. 2. The level of the divided filter](image)

First, we divide the intra-cell into six equal triangles based on the intra-cell center. Subsequently, we connect the center in each triangle to form a hexagon, which is 2/3 multiples of the intra-cell as level II. Level II is used to select the multi-MA as shown in Section 3.2.

Fig. 2 indicates that the cell closest to the center is level I. The level I area is set up according to the density of MH in the intra-cell. In general, the manager is closer to the center in the intra-cell than Multi-MA, thus the level I is smaller than level II. We assume the $count_{max}$ as maximum number of MH and $s$ is multiple of $count_{max}$. The area of level I is $1/s$ times to level II when the number of MHs are $s$ times to preset threshold of MH. Namely, the level I area is $2/3s$ times to intra-cell. For example, the level II area is $20m^2$ and level I is $10m^2$ while intra-cell area is $30m^2$ and the amount of MHs are two times $MH_{count_{max}}$. However, the diagonal line of the hexagon formed by level I cannot be less than $(r-0.86603)/2$. This is because the level I area may not exist when we have a small amount of MH in the MANET (the 0.8663 is the distance between the center of a circle and edge of a hexagon). The limited condition is shown in Fig. 3.
Fig. 3. The minimum area of level I

In general, the divided filter area varies with the density of the MHs. However, we do not know the amount of MHs in the management election phase, thus the level I area is equal to level II initially. Therefore, we use the above method to divide the filter into levels I, II, and III. Subsequently, we can select the manager from level I in the PM election phase.

In general, the cluster head is a critical factor in the MANET topology lifetime. Chatterjee et al. proposed a Weighted Clustering Algorithm (WCA) [5] to elect the cluster header based on the following (1).

\[ I_v = c_1 D_v + c_2 P_v + c_3 M_v + c_4 T_v \]  

We assume that each MH in intra-cell as node \( v \). A description of the parameters is shown as follows:

- \( D_v \): first, we find the \( d_v \) that is the connectivity of each node \( v \). Subsequently, we compute the \( D_v = |d_v - M| \) to indicate the difference between \( d_v \) and \( M \), \( M \) represents the threshold of connectivity,
- \( P_v \): the summary of distances between the node \( v \) and its connected neighbors,
- \( M_v \): the average roaming speed of node \( v \),
- \( T_v \): the misery power index of node \( v \),
- \( c_1, c_2, c_3, c_4 \): these parameters set the weight according to the network situation,
- \( I_v \): the cost of the node \( v \).

In this paper, we use the divided filter to select appropriate MH as cluster header, thus only the specific MHs need to participate in the election phase. The cluster head is elected by \( \min\{I_v, 1 \leq v \leq n, v \in n, n \) represents the amount of nodes\}. For convenient controlling, we assume all of the MHs can connect to the manager. Furthermore, this paper improves the WCA is proposed by Chatterjee et al. [5] as follows:

\[ I_u = c_1 M_u + c_2 T_u + c_3 D_u \]  

The equation proposed by Chatterjee [5] needs each node \( v \) to compute the difference \( (D_v) \) between its connectivity and threshold \( M \). Subsequently, we choose the
minimum difference in all of nodes as the manager. However, this paper based on theory of Chang [2] to elect the manager closest to the center. We use the $D_u$ to substitute for $D_v$ in the (1). The $D_u$ represents the distance from the center and can be computed by $|L_0-L_u|$ ($L_0$ represents the center and $L_u$ represents the location of node $u$). Namely, the MH has smaller $D_u$ and closer to the center. This paper considers the average roaming speed and battery power, thus the $I_u$ represents the costs for the average roaming speed, battery power, and distance from the center. However, the battery power is the most important factor, thus this paper uses the power misery index computation based on Shingh and Raghavendra [19] to compute the battery power. According to (3), we can know the lower $c'_i$ represents higher power misery index on packet transmission/receipt, and similar the $f_i(c'_i)$ is high.

$$f_i(c'_i) = 1/c'_i \quad (3)$$

$i$: the id of MH / $t$: time / $c'_i$: the battery power of MH$_i$ in time $t$ (the value of $c'_i$ is between [0…100])

After the PM election phase, a temporary power-aware manager in the intra-cell can be elected. However, the above equations cannot fully represent the relationship among the distance, roaming speed, and power misery index. Therefore, the Power-aware Assignation Inference System (PAIS) needs to invoked into the PM election phase to elect more appropriate power-aware manager. The PAIS can be used to elect a manager. This PAIS can divide into two steps, setting up the fuzzy members and fuzzy rule design.

**Step 1: Setting up fuzzy members**

The fuzzy members include into three parts, the distance from the center, roaming speed, and power misery index. To reduce the complexity, we divided the distance into far and near. The roaming speed is divided into slow, medium and fast. The power misery index is divided into low, medium and high. Table 2 shows these relationships.

<table>
<thead>
<tr>
<th>Roaming speed</th>
<th>Slow</th>
<th>Medium</th>
<th>Fast</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>The misery index of power</th>
<th>Low</th>
<th>Medium</th>
<th>High</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Distance</th>
<th>Near</th>
<th>Far</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>
In general, a MH closer to the center has a lower distance item value, thus the signal strength is consistent between the selected manager and MHs. Namely, we can gain lower cost \( I_u \) from the (2), however the high \( I_u \) represents the MH is far from the center. Similarly, the MH, which has slow speed, is not easy to roam out the intra-cell, thus it has lower cost in roaming speed item than other MH with high roaming speed. In the power misery index can be used to gain the same conclusion that a MH with enough battery power has a lower cost.

**Step 2: The fuzzy rule of manager**

After the Step 1, each MH transfers the information about the distance, roaming speed and power misery index to the temporary manager. Subsequently, the temporary manager selects an appropriate manager according to that information. However, we set up the distance, roaming speed, and power misery index weights as 0.7, 0.2, and 0.1 as proposed by Huang et al. [7], shown in Table 3. Therefore, we can resilient adjust the weight of the (1) to adapt to network situation. The fuzzy rules contain 18 items and the less sum of value has higher probability to be the manger.

<table>
<thead>
<tr>
<th>Weight</th>
<th>( c_3=0.7 )</th>
<th>( c_1=0.2 )</th>
<th>( c_2=0.1 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rules</td>
<td>Distance ( (D_u) )</td>
<td>Speed ( (M_u) )</td>
<td>Misery index of power ( (T_u) )</td>
</tr>
<tr>
<td>Rule 1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Rule 2</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Rule 3</td>
<td>1</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>Rule 4</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Rule 5</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Rule 6</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Rule 7</td>
<td>1</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Rule 8</td>
<td>1</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Rule 9</td>
<td>1</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Rule 10</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Rule 11</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Rule 12</td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>Rule 13</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Rule 14</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Rule 15</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Rule 16</td>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Rule 17</td>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Rule 18</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>

In MANET, the manager usually takes care the MHs in intra-cell and handles the routing and packet transmission between the inter-cells. The main management tasks are divided into four parts as follows:
Routing maintenance: The manager needs to maintain the location, roaming speed, and bandwidth of all MHs in intra-cell by member table.

Packet relay: The manager takes careful the message exchange and connection between the MHs in intra-cell including the routing phase and packet relay.

Routing discovery: Set up the routing path and packet relay in inter-cell.

Select a Successor: When the manager leaves the intra-cell or retires, it needs to re-compute the area of level I to reduce the filter area and find the appropriate successor.

In general, a single manager needs to manage the MHs, handle the routing path construction, and date relay between inter-cells. Therefore, the manager has two kinds of tables in its memory, member table and routing table. The member table notes the location, level of location, level of identification, speed, and battery power of the MHs in the intra-cell. Otherwise, the routing table has the information of routing path about the source and destination. Therefore, the workload increases fast due to the simultaneous handling cost and bandwidth assignment. Further, the management workload of manager will reduce the performance and exhaust its battery power easily if the misery index rises. The multi-MA inference system is triggered to reduce the management workload in the second phase (PMA assignation phase) when the bandwidth and power misery index achieve the preset maximum threshold. The Power-aware multi-MA (PMA) assignation phase is shown in the next section.

3.2 The Power-aware multi-MA phase (PMA)

The original MA concept comes from the remote procedure call. Its main function is designed for a non-stable network. The MA with the highest mobility can adjust itself immediately to achieve transparency among the platforms. The agent software usually has cross-platform character in that related products such as IBM Aglets, ObjectSpace Voyager, Mitsubishi Concordia, and so on [8,26] can communicate. Therefore, the MA is a useful application for roaming heterogonous networks. Mieso et al. [11,23] introduced a MA application for MH hardware installation in a cluster environment. The selected MA acts as a gateway to take care of inter-cluster communications. Yeh et al. [26] indicated that mobile agents could generate clone agents to divide the tasks. Accordingly, the MA, which has the compatibility, extension, and clones, can adapt to the group communications in an ad-hoc wireless network. The MA has multiple process capability to assign tasks to different kinds of MHs. The different kinds of MHs are also called multi-MA.
The main task of multi-MA is to assist in managing the MHs in intra-cell and divide the management workload. The manager can assign the multi-MA via PMA phase where the congestion and misery index rises. The manager must elect the appropriate multi-MA by computing the capability of all MHs.

We propose two kinds of MAs in the PMA phase, the Power-aware IntrA-MA (PIA-MA) and Power-aware IntEr-MA (PIE-MA).

**The Power-aware Intra-MA (PIA-MA) in PMA election phase**

In general, the manager needs to detect where bandwidth is overloaded and select the appropriate multi-MA during the PMA phase. If the overload is coming from the MHs in the intra-cell, the manager will use the PAIS to elect the PIA-MA, which is located in level II to divide the management in the PMA phase. The manager needs only to take careful the routing construction and packet relay between the inter-cells. The PIA-MA is responsible for the other jobs such as identity management of MHs, routing construction, and packet relay in intra-cell. The Fig. 4 shows the framework between the manager and PIA-MA as follows.

![Fig. 4. The framework of the manager and PIA-MA](image)

The member table of PIA-MA needs to record the intra-cell MH and management information. However, the manager member table needs only to record the PIA-MA information. The main change is that the manager needs to release the bandwidth and relay the packets forwarded to the PIA-MA. The PIA-MA is replaced by the intra-cell manager. The PIA-MA has six characteristics as follows.

1. Auto-detection: the PIA-MA must periodically detect the status of MHs in the intra-cell to get the latest information on the members and routing path.
2. Superintendence: the PIA-MA needs to know whether the manager exists or not.
3. Transmission: the same as the manager, the PIA-MA needs to take careful the path construction and packet relay among the MHs in intra-cell.
4. Clone: Because of the nature of a mobile network, the PIA-MA needs to detect the degree of loading by the PAM inference system. The PIA-MA can directly choose the appropriate MH as new PIA-MAs to divide the jobs when the load is heavy.
5. Mobility: the original PIA-MA needs to choose the new PIA-MAs when it needs to roam away from its intra-cell.

6. Auto-release: the PIA-MA can change its mode to general MH when the number of MHs is getting to disappear and the PIA-MA is idle for a period of time.

**The election of Power-aware IntEr-MA (PIE-MA) in PMA phase**

According to mention above, the manager needs to elect a PIA-MA to divide the jobs due to the overloading is coming from the MHs in the intra-cell. In contrast to PIA-MA, the manager needs to elect a Power-aware IntEr-MA (PIE-MA) when the overload comes from the Inter-cells. Therefore, the PIE-MA must be elected in level I to find the appropriate MHs.

The PIE-MA is divided into two kinds of multi-MAs according to their job characteristics. The first kind is transient PIE-MA, the main job is providing the service of transmission when the manager is overloaded. The transient PIE-MA is different from the PIA-MA, the transient PIE-MA has the character of volatility due to it can change its mode to general MA after finishing its jobs. Namely, the transient PIE-MA can be temporal elected to divide the management load. Another kind of PIE-MA is the permanent PIE-MA. It is the same as the PIA-MA. The permanent PIE-MA needs to elect the new PIE-MA to replace it when original permanent PIE-MA roams away its intra-cell. In general, the manager can select many PIE-MAs to transfer the packets.

Due to the MANET is dynamic in nature, thus the manager needs to handle the routing construction and packet relay in the intra and inter-cell. The next subsection will show the power-aware multi-MA inference system (PMAIS) and how the PIE-MA is elected in the PMA phase.

**Power-aware multi-MA Inference System (PMAIS)**

In this section, we use the multi-MA to divide the management workload and reduce the misery index. To avoid the Ping Pong Effect, we use the fuzzy theorem to replace with the preset fixed threshold as the evaluated index. This is because the manager may continuously generate many multi-MAs to divide the jobs when the manager moves around the fixed threshold. Therefore, the idea behind our PMAIS phase introduces the fuzzy theorem to decide whether the manager needs to elect a multi-MA or not. The PMAIS phase includes there steps: set up the fuzzy member, construct the fuzzy rule to trigger the multi-MA and fuzzy engine, and de-fuzzy.

**Step 1: Set up the fuzzy member**

We have two factors in fuzzy members to consider the status of manager, the
throughput load (assume as \( t \)) and power misery index (assume as \( p \)). To reduce the complexity, we divided the throughput into the low (Lt), medium (Mt), and high (Ht). However, the throughput load needs to be between \([0, t_{\max}]\). We used a trapezoidal-shaped fuzzy number to represent the continuous throughput state shown in Fig. 5.

![Fig. 5. The state of loading of throughput](image)

The power misery index is divided into low (Lp), medium (Mp), and high (Hp), with the values between \([0, p_{\max}]\). We also used the trapezoidal-shaped fuzzy number to represent the power misery index as shown in Fig. 6.

![Fig. 6. The state of power](image)

The workload is divided into four states: low (Lw), medium (Mw), high (Hw), and very high (VHw). Similarly, the values need to be between \([w_{\min}, w_{\max}]\). We set the membership function for the preset four states between \([0,1]\) as shown in Fig. 7. Equations (4) to (7) are inferred from Fig. 7.

![Fig. 7. The state of workload of manager](image)
Step 2: Constructing the fuzzy rule to trigger the multi-MA and fuzzy engine

Equations (4) to (7) the if-then fuzzy rule can be used to represent the workload state. We infer nine rules and discuss these rules one by one in Table 4. After steps 1 and 2, the management workload can be generalized as shown in Table 5 that the vertical axis is the throughput load ($t$) and the horizontal axis is the power misery index ($p$).

Table 4. Fuzzy rule of multi-MA

<table>
<thead>
<tr>
<th>Rule</th>
<th>If $t=L_t$ and $p=L_p$ then $w=L_w$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule 2</td>
<td>If $t=L_t$ and $p=M_p$ then $w=L_w$</td>
</tr>
<tr>
<td>Rule 3</td>
<td>If $t=L_t$ and $p=H_p$ then $w=M_w$</td>
</tr>
<tr>
<td>Rule 4</td>
<td>If $t=M_t$ and $p=L_p$ then $w=L_w$</td>
</tr>
<tr>
<td>Rule 5</td>
<td>If $t=M_t$ and $p=M_p$ then $w=M_w$</td>
</tr>
<tr>
<td>Rule 6</td>
<td>If $t=M_t$ and $p=H_p$ then $w=H_w$</td>
</tr>
<tr>
<td>Rule 7</td>
<td>If $t=H_t$ and $p=L_p$ then $w=M_w$</td>
</tr>
<tr>
<td>Rule 8</td>
<td>If $t=H_t$ and $p=M_p$ then $w=H_w$</td>
</tr>
<tr>
<td>Rule 9</td>
<td>If $t=H_t$ and $p=H_p$ then $w=VH_w$</td>
</tr>
</tbody>
</table>
Table 5. The inference engine of manager

<table>
<thead>
<tr>
<th>Workload (w)</th>
<th>Loading of throughput (t)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Low (L_t)</td>
</tr>
<tr>
<td>Low (L_p)</td>
<td>Low (L_t)</td>
</tr>
<tr>
<td>Medium (M_p)</td>
<td>Low (L_t)</td>
</tr>
<tr>
<td>High (H_p)</td>
<td>Medium (M_p)</td>
</tr>
</tbody>
</table>

Step 3: De-fuzzy

To obtain fast computation results the CoA method [5] is used to convert the fuzzy set representing the overall conclusion obtained in step 2 into a real number that, in some sense, best represents the fuzzy set by using equation (8).

\[ y = \frac{\sum_{i=1}^{n} W_i B_i / \sum_{i=1}^{n} W_i}{\sum_{i=1}^{n} W_i} \]  \hfill (8)

The parameter W is computed from the fuzzy rule and the parameter B is represented the workload state. However, we use the values -5, 0, 5, and 10 to represent the degree of B into the input (show as Fig. 7). The parameter y is output from the de-fuzzification process. The management workload state is shown as follows.

- **Case 1:** The manager is stable when the y value is negative and less than -5.
- **Case 2:** The management workload is “Low” when the y value is between -5 and 0 and the “Low” degree is higher than “Medium”. Thus, the manager does not trigger the multi-MA.
- **Case 3:** The management workload is “Medium” when the y value is between -5 and 0 and “Medium” is higher than “Low”. Thus, the manager does not trigger the multi-MA.
- **Case 4:** The management workload is the same as Case 3 when the y value is between 0 and 5 and “High” is lower than “Medium”.
- **Case 5:** The management workload is “High” when the y value is between 0 and 5 and “High” is higher than “Medium”. Thus, the manager enters the cordon.
- **Case 6:** The management workload is the same as Case 5 when the y value is between 5 and 10 and “High” is higher than “Very High”.
Case 7: The management workload is “Very High” when the y value is between 5 and 10 and “High” is lower than “Very High”. Thus, the manager triggers the multi-MA in PMA phase.

Case 8: The manager triggers the multi-MA in PMA phase when the y value is higher than 5.

However, the output value is between the two stats, such as y=7.5 (the degree of “High” is equal to “Very High”). Based on the conservative rule, the manager must trigger the multi-MA in PMA phase. We illustrate an example as follows:

Example: we assume the degree with loading of throughput \( t \) is \{0.8, 0.2, 0\} and the power misery index \( p \) is \{0.7, 0.3, 0\} to input to fuzzy rule. Subsequently, the results are gained from the \( \min\{t, p\} \) and shown in Table 6.

Table 6 shows the rules 1, 2, 4, and 5 influences the system output value, thus we input (8) \((y=(0.7*(-5) + 0.2*(-5) + 0.3*(-5) + 0.2*0)/0.7+0.2+0.3+0.2 =-4.2)\) to compute the membership degree. The result obtained from (8) is between \([-5, 0]\], subsequently we gain the values \( \alpha=0.84 \) and \( \alpha=0.16 \) by inputting the result from (8) to (4) and (5) again. These results lead to the conclusion that the manager is stable and belongs to case 2 due with the degree of “Low” higher than the degree of “Medium”.

Table 6. The computation result of fuzzy rule

<table>
<thead>
<tr>
<th>Workload ((w))</th>
<th>Loading of throughput ((t))</th>
</tr>
</thead>
<tbody>
<tr>
<td>L(_t) 0.8</td>
<td>L(_p) 0.7</td>
</tr>
<tr>
<td>M(_t) 0.2</td>
<td>M(_p) 0.3</td>
</tr>
<tr>
<td>H(_t) 0</td>
<td>H(_p) 0</td>
</tr>
</tbody>
</table>

Due to the multi-MAs have clone characteristics, inheritance, packet relay, and auto-release, the selected multi-MAs need to detect and decide whether is clone itself by inference engine. The multi-MA needs to select a new multi-MA to divide its job to reduce the loading when it detects the state to become the “High”. However, the state becomes to “Very High”, and then the multi-MA needs to release the status of multi-MA and select another multi-MA to replace it. Finally, the multi-MA, which the state is “Very High”, changes its status to a general MH.
4. The results of experiment

In this section, we illustrate the proposed protocol to prolong the lifetime of the manager. A temporary manager that is closer to the center is elected first, and sequentially it can use the PAIS in PM phase to choose the appropriate manager. However, the PAIS needs to consider three parameters, such as the distance from the center in the intra-cell, average of roaming speed, and battery power.

Besides, the PAM phase is invoked to assist the manager to divide up the loading due to the manager needs to take care all MHs in the intra-cell and neighbor cells. Therefore, we input the battery power and bandwidth of manager to (4) to (7), subsequent the workload can be gained from the de-fuzzification process. We can decide whether the manager needs to select the multi-MA by the workload. In general, the PMA phase has two kinds of multi-MAs, the PIA-MA and PIE-MA. Therefore, the manager needs to detect where the bandwidth is overloading and select the appropriate multi-MA in PMA phase. We assumed the following hypotheses in our experiment to prove the power-aware HCBM can prolong the manager lifetime [3] as follows.

1. All MHs consume their battery power vary with time and they do not charge their battery power even they are in an idle state.

2. There are seven cells in our simulated network topology and we choose the manager, which is locating in the center of cells to observe. This is because a manager at the center can clearly analyze the HCBM and power-aware HCBM.

Based on the above assumptions, our experiments are divided into two steps. At first, we use the Network Simulation 2 (NS2) [28] to generate the MHs randomly. Subsequently, we use the JAVA to generate the two inference engines, the PAIS and PMAIS. The experimental environment is shown in Table 7.

Table 7. The environment of experiments

<table>
<thead>
<tr>
<th>Item</th>
<th>Environment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topology</td>
<td>100*100 m</td>
</tr>
<tr>
<td>Node</td>
<td>20</td>
</tr>
<tr>
<td>Packet Size</td>
<td>512 Bytes</td>
</tr>
<tr>
<td>Initial Battery Power</td>
<td>Random</td>
</tr>
<tr>
<td>Current Transfer</td>
<td>0.6 Ampere</td>
</tr>
<tr>
<td>Current Receive</td>
<td>0.3 Ampere</td>
</tr>
<tr>
<td>Current Idle</td>
<td>0.009 Ampere</td>
</tr>
</tbody>
</table>
We set the topology as 100 * 100 m by NS2 and generate 20 MHs randomly in step 1. In general, the MH enter the topology may not have full capacity of battery power, thus we set the initial battery power randomly to adapt to the real environment. The packet sets were 512 bytes, using a Constant Bit Rate (CBR) to transfer. The costs of transmission and receipt which are computed by energy consume equation of NS2 are 0.6 A and 0.3. The energy formula shows as formula (9).

\[ E_{\text{total}} = E_{\text{total}} - (P_{\text{power}} \ast T_{t}) \]  

(9)

The parameter \( E_{\text{total}} \) represents the remainder battery power, the \( P_{\text{power}} \) represents the consumed current of transmission, and the \( T_{t} \) represents the transmission time, thus the \( P_{\text{power}} \ast T_{t} \) represents the consumed battery power of transmission action (the unit is Coulomb).

This paper divides our experiments into four scenarios and discusses them respectively. The four scenarios show as follows:

Scenario 1: The node with a slow average roaming speed and sufficient battery power.
Scenario 2: The node with a fast average roaming speed and sufficient battery power.
Scenario 3: The node with a slow average roaming speed and insufficient battery power.
Scenario 4: The node with a fast average roaming speed and insufficient battery power.

**The scenario 1 of network topology**

In type 1, the election procedures of manager and multi-MA are the same as HCBM among nodes with sufficient battery power and slow average roaming speed. We set the roaming speed and battery power weights in (2) between the [0.1, 0.4] due to those factors are not the critical issues. Besides, the weights of distance are set between [0.5, 0.8]. The setting of weights shows as follows:

\[
\begin{align*}
I_{u_1} &= 0.1M_{u_1} + 0.1T_{u_1} + 0.8D_{u_1} \\
I_{u_2} &= 0.2M_{u_2} + 0.1T_{u_2} + 0.7D_{u_2} \\
I_{u_3} &= 0.3M_{u_3} + 0.1T_{u_3} + 0.6D_{u_3} \\
I_{u_4} &= 0.4M_{u_4} + 0.1T_{u_4} + 0.5D_{u_4} \\
I_{u_5} &= 0.1M_{u_5} + 0.2T_{u_5} + 0.7D_{u_5} \\
I_{u_6} &= 0.1M_{u_6} + 0.3T_{u_6} + 0.6D_{u_6} \\
I_{u_7} &= 0.1M_{u_7} + 0.4T_{u_7} + 0.5D_{u_7}
\end{align*}
\]

For keeping the objectivity, we simulate 500 times for each group in Fig. 7. We can see the power-aware HCBM (see the ———) we proposed has longer lifetime than original HCBM (see the ———). At first, the power-aware HCBM result is closer to the HCBM result before 15 seconds. After 15 seconds, the power-aware manger triggers the multi-MA to divide the load when it detects “Very High” load. However, this trigger action may increase the temporary manager load and stops after 20 seconds. This brief workload increase will not cause a serious effect on the manager lifetime, thus the power-aware can still suspend the degree of power to decrease.
The scenario 2 of network topology

In scenario 2, we elect appropriate manager or multi-MA among nodes with sufficient battery power and fast average of roaming speed. We set the distance and battery power weights between [0.1, 0.4] because those factors are not critical factors. The main factor is average roaming speed, thus we do not hope to elect the manager with fast average of roaming speed. We set the weights of distance between [0.5, 0.8] to elect stable manager. The setting of weights shows as follows:

\[
I_{u1} = 0.8M_{u1} + 0.1T_{u1} + 0.1D_{u1} \quad I_{u2} = 0.7M_{u2} + 0.2T_{u2} + 0.1D_{u2} \quad I_{u3} = 0.6M_{u3} + 0.3T_{u3} + 0.1D_{u3}
\]

\[
I_{u4} = 0.5M_{u4} + 0.4T_{u4} + 0.1D_{u4} \quad I_{u5} = 0.7M_{u5} + 0.1T_{u5} + 0.2D_{u5} \quad I_{u6} = 0.6M_{u6} + 0.1T_{u6} + 0.3D_{u6}
\]

\[
I_{u7} = 0.5M_{u7} + 0.1T_{u7} + 0.4D_{u7}
\]

In the Fig. 8, the power-aware HCBM gains had better result in lifetime than HCBM. It is because that the nodes with fast average roaming speed easily roam away its original cell. Thus, packet loss and re-transmission frequently occurs. Namely, the manager is easy to exhaust its battery power and reduce the lifetime of manager of HCBM. However, the power-aware HCBM adjust the weight to adapt to this network, thus it can prolong the manager lifetime.

The scenario 3 of network topology

Subsequently, we considered a different environment in which nodes with a slow roaming speed and insufficient battery power existed. In this environment, the battery
power is most important factor. We set the battery power weight between [0.5, 0.8]. The distance and roaming speed weights were set between [0.1, 0.4]. The setting of weights shows as follows:

\[
I_{u1} = 0.1M_{u1} + 0.8T_{u1} + 0.1D_{u1} \quad I_{u2} = 0.2M_{u2} + 0.7T_{u2} + 0.1D_{u2} \quad I_{u3} = 0.3M_{u3} + 0.6T_{u3} + 0.1D_{u3}
\]

\[
I_{u4} = 0.4M_{u4} + 0.5T_{u4} + 0.1D_{u4} \quad I_{u5} = 0.1M_{u5} + 0.7T_{u5} + 0.2D_{u5} \quad I_{u6} = 0.1M_{u6} + 0.6T_{u6} + 0.3D_{u6}
\]

\[
I_{u7} = 0.1M_{u7} + 0.5T_{u7} + 0.4D_{u7}
\]

In this scenario, the manager easily exhausts its battery power in HCBM and power-aware HCBM. The power-aware HCBM has longer lifetime than HCBM due to the power-aware HCBM elects the appropriate manager and multi-MA to divide the workload. However, the results shown in Fig. 9 are worse than that in scenarios 1 and 2. This is because the battery power is a critical issue in MANET for prolonging manager lifetime. Obviously, the lifetime of MANET in HCBM method almost stops at 20 seconds.

![Power Consumption Graph](image)

**Fig. 9. The result of experiment in scenario 3**

### The scenario 4 of network topology

Finally, we introduce last kind of network with nodes that have a fast average roaming speed and insufficient battery power. We set the battery power and roaming speed weights between the [0.35, 0.45] due to those factors are important factors. Besides, we set the weight of distance between [0.1, 0.3] and show the setting as follows:

\[
I_{u1} = 0.45M_{u1} + 0.45T_{u1} + 0.1D_{u1} \quad I_{u2} = 0.40M_{u2} + 0.40T_{u2} + 0.2D_{u2} \quad I_{u3} = 0.35M_{u3} + 0.35T_{u3} + 0.3D_{u3}
\]

Fig. 10 shows the network of scenario 4 is worst case among the all kinds of networks. The manager easily roams out its section and exhausts its battery power. Therefore, the results of the HCBM and power-HCBM are worst among all of the types. However, the power-aware HCBM can elect the multi-MA to divide the workload, thus it can gain better result than HCBM.
We determined that the power-aware HCBM could prolong the manager lifetime by electing the appropriate manager and multi-MA. Regardless of the network type, the power-aware method can gain better results than HCBM. Therefore, we can gain a more stable network topology by power-aware HCBM than previous works [1].

5. Conclusion

This study used an inference engine to elect a power-aware manager and multi-MA by considering the bandwidth and power misery index. In the PM phase, we elect an appropriate power-aware manager to take care of the intra-cell and inter-cell tasks. Our method uses the multi-MAs to divide the manager workload in the PMA phase. Furthermore, we divide the multi-MAs into two parts, the PIA-MA and PIE-MA. The PIA-MA is responsible to support the workload of manager in intra-cell and the PIE-MA is responsible to inter-cell. Therefore, the power-HCBM can prolong the manager lifetime resulting in a stable network topology.

In our experiment, the situation of network can be divided into four scenarios and discuss them respectively. This paper uses different values as roaming speed, battery power, and distance weights for simulation. The MHs with fast roaming speed and insufficient battery power gain worst results than others. We observed four scenarios in our experiments to determine that battery power is a critical factor in MANET. Therefore, the manager in MANET needs sufficient battery power to handle the tasks in its cell. It is for this reason that the manager in power-aware HCBM we proposed can adapt to the changeability of MANET.

Reference


[14]. V.D. Park and M.S. Corson, “Applicability of The Temporally-ordered Routing


Kuo-Qin Yan received the B.S. and M.S. degrees in Electrical Engineering from Chung-Cheng Institute of Technology and the Ph.D. degree in Computer Sciences from National Tsing-Hua University, Taiwan. Currently, he is a Professor of the Department of Business Administration, Chaoyang University of Technology, Taichung County, Taiwan. His current research interests include grid computing, distributed data processing, parallel processing, fault tolerant computing, and mobile commerce.

Shu-Ching Wang received the B.S. degree in Computer Science from Feng-Chia University, the M.S. degree in Electrical Engineering from National Chen-Kung University, and Ph.D. degree in Information Engineering from National Chiao-Tung University, Taiwan. Currently, she is a Professor with the Department of Information Management, Chaoyang University of Technology, Taichung County, Taiwan. Her current research interests include grid computing, distributed data processing, parallel processing, and algorithm analysis & design.

Mao-Lun Chiang received the M.S. degree in Information Management from Chaoyang University of Technology, Taiwan. Currently, he is pursuing his Ph.D. degree in Department of Computer Science, National Chung-Hsing University, Taiwan. His current research interests include distributed data processing, fault tolerant computing, and mobile computing.