Study on Wireless Heterogeneous Mesh Network Construction and Load Balanced Routing

CAO Peng[1], *NING Qian*[1*], *HUANG Linyu*[1], *CHEN Bingcai*[2], *TANG Chengwen*[1]

1. College of Electronics and Information Engineering, Sichuan University, Chengdu 610065, P. R. China;
2. School of Computer Science and Technology, Dalian University of Technology, Dalian 116024, P. R. China

(Received 5 May 2023; revised 10 September 2023; accepted 13 October 2023)

Abstract: In application scenarios such as smart cities, a single wireless communication technology cannot cover complex network environments, while different communication technologies cannot communicate with each other, making it impossible to take full advantage of different wireless communication technologies. Therefore, we design a device that can communicate with multiple wireless communication nodes, which is called wireless heterogeneous communication modules (WHCM). WHCM is used as a hub to build a communication heterogeneous mesh network (CHMN). CHMN consists of nodes with multiple wireless communication protocols. We propose an AD HOC on-demand multi-path distance vector (AOMDV) routing protocol based on CHMN networks, named CH-AOMDV. CH-AOMDV is able to identify different types of communication protocols during the route initiation and path establishment process. It can also identify the communication distance, data transmission rate, energy and the load capacity of each node in CHMN. Simulations with NS-2 show that the performance of the CHMN is better than that of the traditional network when the distribution density between nodes is higher in terms of packet delivery rate, average end-to-end delay, throughput rate and routing overhead. This work proposes a new method of implementing CHMN and the corresponding routing protocol CH-AOMDV. The proposed protocol is superior to the other three protocols, improves network lifecycle, throughput and packet delivery rate, and reduces node overhead and average end-to-end delay. The protocol is useful for CHMN.

Key words: heterogeneous network; STM32; AD HOC on-demand multi-path distance vector(AOMDV); path load; NS-2

CLC number: TN927.2 **Document code:** A **Article ID:** 1005-1120(2023)06-0738-15

0 Introduction

With the development of Internet of things (IoT) and wireless communication technology, a variety of wireless networks coexist in smart cities, smart homes, smart agriculture and other fields. Each wireless communication technology has different communication protocols. They construct communication heterogeneous mesh network (CHMN). WiFi, Bluetooth, Lora, ZIGBEE and other wireless communication technologies have different advantages and disadvantages. Different communication technologies have strong complementarity. Due to different communication protocols, different types of wireless communication technologies cannot achieve communication, while a single wireless communication technology cannot adapt to the complex and changing environment.

Among the existing studies on heterogeneous networks, Kim et al.[1] proposed a load balancing algorithm for heterogeneous networks based on motion state estimation. The user's motion state was first estimated, and then some of the load was transferred from the cellular network to the WLAN based on the estimated result. Wang et al.[2] pro-

*Corresponding author, E-mail address: ningq@scu.edu.cn.

How to cite this article: CAO Peng, NING Qian, HUANG Linyu, et al. Study on wireless heterogeneous mesh network construction and load balanced routing[J]. Transactions of Nanjing University of Aeronautics and Astronautics, 2023, 40(6): 738-752.

http://dx.doi.org/10.16356/j.1005-1120.2023.06.010

posed a heterogeneous networking scheme for Profinet and Modbus networks based on software-defined networking (SDN) due to the advantage of Profinet and Modbus in the industrial field to improve the efficiency of industrial production. Jiang et al.[3] proposed a four-layer HetIoT architecture consisting of sensing, networking, cloud computing, and applications. Cheng et al.[4] used the underlying heterogeneous small cell network for IoT and proposed an energy-efficient framework based on it. Wang et al.[5] integrated cellular network with dedicated short range communications (DSRC) for heterogeneous vehicular network (HetVNET) as a potential solution to meet its communication requirements as a potential solution. Most of the existing research on heterogeneous networks[2-13] are reflected in the heterogeneity of the network architecture and the different states of the nodes. This neither fully utilizes the advantages of different wireless communication technologies, nor allows for flexible networking. Plus, most of them adopt network simulation and lack of concrete realization methods. Therefore, this paper proposes a heterogeneous network using multiple wireless communication technologies for self-organization and implements this network using STM32.

In this paper, in order to realize the data communication between different communication technologies and form a CHMN, as shown in Fig.1, we design the wireless heterogeneous communication module (WHCM) that can communicate with multiple wireless communication nodes.

The WHCM is capable of interacting with a variety of communication nodes for data. After writing

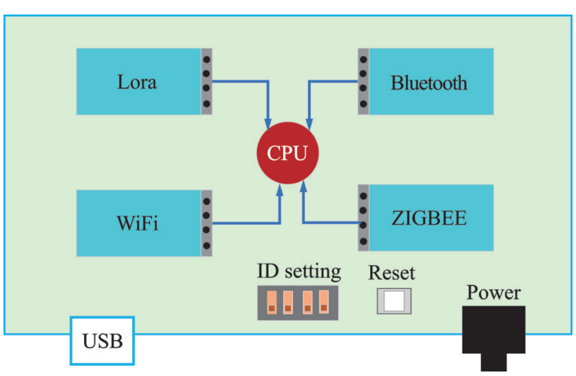

Fig.1 WHCM module

the module initialization code to the CPU and turning on the power, the WHCM completes the initialization of the modules and establishes connections with the nearby nodes in the environment. Several WHCMs are put into a complex network with multiple communication technologies. With the WHCM acting as a bridge, nodes that cannot communicate directly establish connections, thus forming a CHMN, where the CHMN is a wireless self-organizing mesh network, as shown in Fig.2.

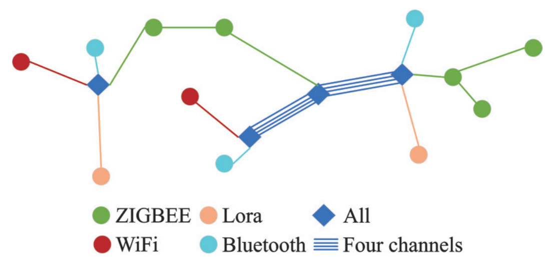

Fig.2 WHCM

The CHMN is developed on the basis of the traditional wireless self-organized mesh network and has its characteristics. The CHMN has the characteristics of WiFi, Bluetooth, Lora and ZIGBEE at the same time. Compared with the traditional wireless mesh network, the CHMN appears to be more complex, and these features bring serious challenges to the design of routing algorithms.

Most of current routing technologies based on mesh networks are developed on the basis of on-demand routing protocols. The common on-demand routing protocols are AD HOC on-demand distance vector routing (AODV) and AD HOC on-demand multi-path distance vector (AOMDV). Due to the complexity of CHMN performance and topology variability, single-path routing cannot reflect the advantages of each node in the CHMN, while multi-path algorithms can reduce network congestion by shifting traffic to other paths that are less loaded. In the study of existing multipath routing protocol AOMDV, Zhang et al.[12] proposed a multipath routing protocol for mobile edge computing based on link lifetime and energy consumption prediction with an energy hierarchy strategy. When the node energy is below a threshold value, it no longer participates in routing discovery. In the route selection phase, the path is selected based on the survival of the routing link and the minimum energy consumption of the route. Chen et al.[14] proposed a topology change adaptive AOMDV (TA-AOMDV) routing protocol that deployed a stable path selection algorithm that not only used node resources (remaining energy, available bandwidth and queue length) as path selection parameters, but also considered the probability of link stability between nodes. Yang et al.[15] proposed a multipath routing protocol adaptive energy and queue AOMDV (AEQAOMDV) based on adaptively sensing node residual energy and buffer queue length. Combined with the existing research on AOMDV routing protocols[16-28], the CH-AOMDV routing protocol based on CHMN proposed in this paper has the ability to identify different communication protocol types and is able to operate on CHMN. At the same time, CH-AOMDV is able to establish paths for different types of nodes in terms of data transmission rate, communication distance, energy condition, and node load, and the established paths have smaller link load.

The rest of the paper is organized as follow: Section 1 discusses the principles of communication heterogeneous network implementation; section 2 describes the CHMN-based multipath routing protocol; section 3 discusses the experimental and comparative results for communication heterogeneous network; section 4 summarizes this study and presents future work.

1 Heterogeneous Network Construction

In this paper, we use the WHCM as a hub to form the CHMN. The NS-2 network simulation platform is modified according to the implementation idea of WHCM.

1.1 Design and implementation of WHCM

In order to realize CHMN, four types of wireless communication modules are selected, which are widely used and easily available in the market such as WiFi, Bluetooth, Lora and ZIGBEE. The core of realizing CHMN lies in realizing the data transmission between different types of wireless communication modules. As shown in Fig.3, by designing the WHCM, the initialization code is written into the STM32 chip, and then several wireless communication modules are inserted into the corresponding positions of the printed circuit board (PCB). After the device is powered on, the modules are initialized and automatically establish contact with multiple wireless communication nodes in the vicinity. The method is tested and found to be feasible. WHCM is operated by using the method described in Algorithm 1.

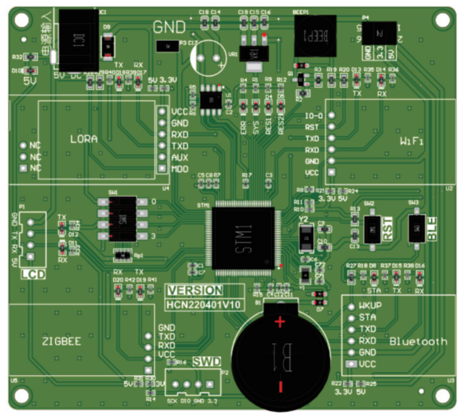

Fig.3 Implementing WHCM using STM32

Algorithm 1 WHCM operation

```
while (1)
  if (WiFi received ) then
    run WiFi processing code
  else if (Lora received) then
    run Lora processing code
  else if (ZIGBEE received) then
    run ZIGBEE processing code
  else if (Bluetooth received) then
    run Bluetooth processing code
end if
```

1.2 Split object model

Due to many factors such as cost and environment, we implement the simulation of the CHMN through the NS-2 network simulation platform. As shown in Fig.4, NS-2 is an object-oriented simulator written in OTcl and C++. OTcl acts as the front-end and C++ acts as the back-end for running the actual simulation.

The simulation and testing of the CHMN could not be accomplished because the wireless communication node structure of the NS-2 platform does not match the WHCM. The key to implement the CHMN on NS-2 is to design the wireless heterogeneous nodes like WHCM, which requires modification of the wireless communication node hierarchy. The wireless communication node involves the underlying files from the physical layer to the application layer. Since the split object model of NS-2 uses both C++ and OTcl, it is necessary to modify both files. As can be seen in Fig.4, the class hierarchy of the two languages can be independent or the OTcl and C++ interfaces can be linked together using Tclcl. There are two types of classes in each domain, and the first type includes classes linked between the C++ and OTcl domains. These OTcl and C++ class hierarchies are called the interpretation hierarchy and the compilation hierarchy, respectively. The second type includes OTcl and C++ classes that are not linked together. These classes are not part of either the interpretation hierarchy or the compilation hierarchy. Roughly speaking, OTcl builds the network (e.g., creating and connecting nodes), while C++ runs the actual simulation (e.g., passing packets from one node to another). The implementation of WHCM requires full modification of both C++ and OTcl files.

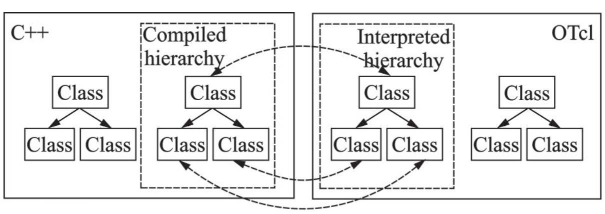

Fig.4 Split object model of NS-2

1.3 Wireless heterogeneous node model

When creating a wireless communication node, it is necessary to set up a structure including physical layer, link layer, mac layer, application layer, antenna, etc. Different layers are encapsulated into different modules through C++ code. Different modules retain interfaces to other modules. Through the logical relationship that exists between OTcl code and C++ variables, discrete hierarchical modules can be connected to establish a hierarchical structure of wireless communication nodes.

The specific files include ns-mobilenode.tcl, ns-lib.tcl, ns-trace.tcl, and so on. The hierarchical structure of wireless communication nodes in NS-2 is given in Fig.5, where ARP means the address resolution protocol. It is not difficult to find that the structure is obviously mismatched with the structure of the WHCM as shown in Fig.1. The key to the implementation of the CHMN lies in the implementation of the functions and structure of the WHCM, so the hierarchical structure of wireless communication nodes needs to be redesigned.

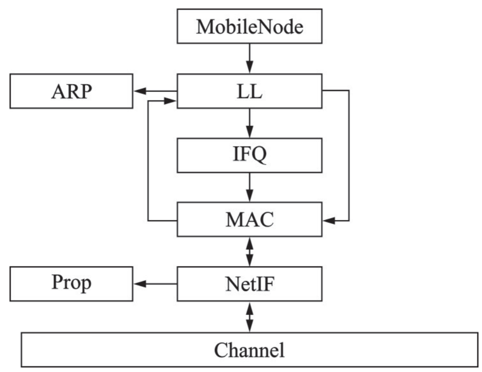

Fig.5 Single channel wireless communication node

Fig.6 presents the structure of a multi-channel wireless communication nodes like WHCM. LL, IFQ, MAC, NetIF and Prop are the date link layer, node buffer queue, mac layer and propagation model of the node, respectively. This constitutes the four channels of WHCM, which represent the four communication protocols of WiFi, Bluetooth, Lora and ZIGBEE. When different channels are adopted, the data transmission rate, communication distance, energy consumption and node load capacity are also different.

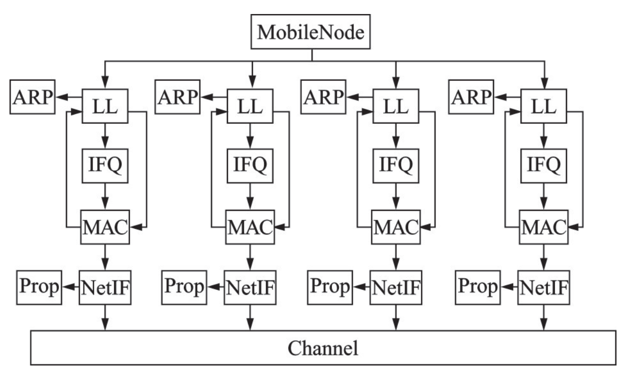

Fig.6 Multi-channel WHCM

1.4 Implementation principle of heterogeneous node structure

NS-2 utilizes a split-object model, where each module structure has a logical relationship in C++ and OTcl. The MobileNode class and its parent class Node are defined in the C++ file, while the add-interface class and its objects are defined at the OTcl file through Node/MobileNode in the OTcl language, including channel, mactype, lltype, ifqtype, anttype and the member variables mac_, ll_, ifq_, netif_, etc. The NS-2 modules are shown in Figs.5,6.

Each block diagram represents an NS-2 module with independent functions, such as mac layer, link layer, interface buffer queue, etc. which is modularized through C++ and retain the logical relationship with OTcl and the interface for interaction with other layers. The OTcl provides parameters and metrics for network emulation construction and operation, which is bundled with C++ files through the OTcl language.

As shown in Fig.7, the node obtains their own node ID value according to the OTcl to confirm whether they are WHCM. If not, they maintain the node structure, otherwise they change the node structure to the WHCM structure with four channels.

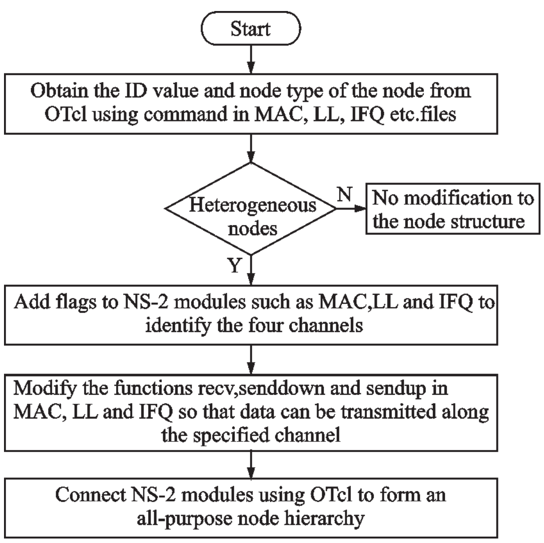

Fig.7 Modification of wireless node structure

The four channels are also differentiated and given different heterogeneous characteristics. The functions for data transmission at this level are modified accordingly so that packets can be transmitted within the same channels of the protocol. Finally, the OTcl language is used to connect the modules together as described by Algorithm 2.

Algorithm 2 Wireless node structure setting

Set ns [Simulator instance]

If (node_type==WHCM)

i←4 /*set four channels */

While {i}

ifq_i_($t) ← [new $qtype] /*set interface queue */

mac_i_($t)←[new $mactype] /*set mac layer */

ll_i_($t)←[new $lltype]

netif_($t)←[new $iftype]

$mac up-target $ll /*connect mac layer and link layer */

$ll down-target $ifq

$mac netif $netif

$ifq target $mac

$channel addif $netif

i←i-1

In OTcl language, we use the add-interface defined by Node/MobileNode to complete the module connection. Firstly, an ns entity is defined, and that the node is the WHCM based on its ID is confirmed. Then, the mac layer, link layer, and IFQ layer interfaces and connect the modules are initialized, as described in Algorithm 2, which provides a simple idea for connecting discrete NS-2 modules into a whole structure. In addition, NS-2 module definition and other settings also need to be modified separately in OTcl.

1.5 Heterogeneous features setting

CHMN consists of several wireless communication technologies which include different wireless communication modules. According to the research, the performance differences between different communication modules are mainly reflected in four aspects: Energy consumption, data transmission rate, data transmission distance and load capacity. When data transmission is performed, except for WHCM which can transmit data with all nodes, other nodes cannot communicate directly with different types of nodes.

In the CHMN, a unique ID is generated when the node is created in the OTcl file and stored in the node class as a chain table in the corresponding C++ file. The node ID values are obtained by C++ and OTcl, and used to distinguish the different node types and modify their energy consumption characteristics accordingly.

The node and its ID are created in OTcl. Then the OTcl language is used to read the pointer to this node in the C++ file. According to the relationship between OTcl and C++, we call their common parent class TclObject and use it to point the pointer to the mobilenode class of this node. In this way we get the ID value of this node. Based on the ID value, the node's energy consumption is modified.

During data packet transmission, the node's location information (X, Y, Z) and node type (node_type) are uploaded. Each time a packet is received, the node first calculates the distance between two nodes and determines whether it can communicate directly according to the protocol between the two nodes. In this way the nodes complete the interaction of data when the communication distance and communication protocol meet the communication conditions.

The timer class function LinkLoadTimer in NS-2 is used to get the load of data. By periodically obtaining the load value of the node, the function calculates the weighted average of the node load and uses this value to represent the load of node.

Finally, the transmission rate of different nodes is modified accordingly in the OTcl file according to the node ID value. The agent mode is user datagram protocol (UDP), and constant bitrate (CBR) data stream is used for data transmission. Each node can act as a source node or a destination node in the network. The transmission rate is modified as described in Algorithm 3.

Algorithm 3 Transmission rate setting

```
Set udp($i) [new Agent/UDP]
$ns attach-agent $node($i) $udp($i)
while {$dest == $i}
 set dest [expr round([$randomNode value])]
 set cbr($i) [new Application/Traffic/CBR]
 $cbr($i) attach-agent $udp($i)
 $cbr($i) set rate_ [expr $val(throughput)
 *rate_type]
```

2 Analysis of Routing Algorithms

2.1 Path load calculation

The CH-AOMDV multipath routing algorithm can operate in CHMN and take full advantage of each type of node to achieve load balancing of the network. The principle of implementation is that the communication transmission rate, communication distance, the node load and energy status of each node is used to calculate the path load value while establishing the path. The path is selected based on this value so that the selected path fully takes into account the load of the nodes on the path. The calculation method is given as

$$\text{path\_load} = \text{Energy}\cdot\text{energy} + \text{Speed}\cdot\text{speed} + \text{Len}\cdot\text{length} + \text{Distance}\cdot\text{distance} \tag{1}$$

where "energy" represents the path energy suitability; "speed" the transmission rate suitability, "length" the link load suitability; and "distance" the communication distance suitability; "Energy" "Speed" "Len", and "Distance" are their weight values, respectively. For example, if we want to focus on the impact of node residual "energy" on the routing protocol performance, we should increase "Energy" appropriately. The calculation method is given as

$$\text{Energy} = 1 - \frac{\sum_{i=1}^{S_{um}} E_i}{E_{\text{initial}} \times S_{um}} \tag{2}$$

$$\text{Distance} = \frac{\sum_{i=1}^{S_{um}} D_i}{D_{\max} \times \sum_{i=1}^{S_{um}} H_i} \tag{3}$$

$$\text{Speed} = 1 - \frac{\sum_{i=1}^{S_{um}} V_i H_i}{V_{\max} \times \sum_{i=1}^{S_{um}} H_i} \tag{4}$$

$$\text{Len} = 1 - \frac{\sum_{i=1}^{S_{um}} L_i H_i}{L_{\max} \times \sum_{i=1}^{S_{um}} H_i} \tag{5}$$

$$L_i = \frac{\sum_{j=1}^{j=5} j \times R}{\sum_{j=1}^{5} j} \tag{6}$$

where S_{um} represents the number of nodes between the current node and the destination node; E_i the remaining energy; D_i the transmission distance; V_i the communication rate; H_i the hop count and L_i the buffer value. After normalising the heterogeneous characteristics of CHMN, the smaller the value obtained, the smaller the load value of that metric on the corresponding path, the range of values is shown below

$$\text{Eenrgy} + \text{Speed} + \text{Len} + \text{Distance} = 1 \tag{7}$$

$$\text{Eenrgy} \in [0,1] \tag{8}$$

$$\text{Len} \in [0,1] \tag{9}$$

$$\text{Distance} \in [0,1] \tag{10}$$

$$\text{Speed} \in [0,1] \tag{11}$$

2.2 Routing discovery process

The CH-AOMDV algorithm is modified so that it can run on the CHMN constructed in this paper and achieve the function of load balancing. When a node in the network is about to send data, the steps are as follows.

Step 1 When a node wants to send data to another node, the source node will query the routing table to find out if there is an active route to the destination node. If there is "yes", it will choose a path with the minimum path_load value from all paths to send data. If there is "no", the node will write its node_type and location into the "Route Request" packet and send it out.

Step 2 When the intermediate node receives the "Route Request" packet, it first checks whether the node_type and communication distance of the message meet the reception conditions. If it is "satisfied", it goes to Step 4, otherwise it discards the packet. When node_type=0, it means the node is WHCM.

Step 3 If the reverse path does not exist, a reverse path is established and then whether the "Route Request" packet is from itself or it is received repeatedly is determined. If it is "no", it goes to Step 7; otherwise it discards the packet.

Step 4 If the intermediate node is the destination node, the node will create a "Route Reply" packet. The packet is updated with node_type, node location information "X, Y, Z" and heterogeneous parameter values "energy" "len" "speed" "distance", and then the "Route Reply" packet follows the reverse path of the "Route Request" packet.

Step 5 If the intermediate node is not the destination node, the node looks up the routing table and determines if a route to the destination node exists. If "exists", the node updates the packet with node_type, node location information "X, Y, Z" and heterogeneous parameters "energy" "len" "speed" "distance" to the "Route Reply" pachet. Then the "Route Reply" packet is sent to the source node following the reverse path of the "Route Request" packet information packet. Otherwise, it goes to Step 6.

Step 6 The node determines whether the final broadcast is set to real. If it is "set to real", the packet is discarded; otherwise, the node updates node_type and node location information "X, Y, Z" to the "Route request" packet and forward it. The route discovery process is described in Algorithm 4.

Algorithm 4 Route discovery process

```
if(s_type==r_type||s_type==0||r_type==0)
 if (distance<reference distance)
  if (s_destination== r) then discard RREQ
   if (reverse route table=NULL) then
    add reverse route table
   end if
if (r_seqnum <s_ seqnum) then
  pathlist = NULL
  Reverse_Path_insert(nexthop, lasthop, hop-
  count+1)
else if (s_seqnum==r_seqnumdP&&r_adver-
      tised_hops>s_hopcpunt)
    then
    if(r_firsthop==s_firsthop&&r_lasthop
    ==r_lasthop)
      update reverse_path
    else if (new_disjoint_path)
      Reverse_path_insert (nexthop, lasthop,
      hopcount+1)
end if
```

```
if (s_dst == r)
   sendReply (node_type, seat, heterogeneous
   parameters)
else
        update RREQ
        forward RREQ
      end if
    end if
   else discard RREQ
   end if
else discard RREQ
end if
```

In the CH-AOMDV protocol, when the destination node receives a "Route Request" packet, or when an intermediate node with a valid route to the destination node receives a "Route Request" packet, it sends a "Route Request" packet to the source node. The processing steps of a node receiving a "Route Request" packet in the CH-AOMDV protocol are as follows. It is worth mentioning that when node_type=0, it indicates that the node is WHCM.

Step 1 After a "Route Reply" packet is received, the node first determines whether the protocol and communication distance meet the reception conditions. If it "meets", it first determines whether the "Route Reply"packet is sent by itself. If it is "yes", it discards the packet. Otherwise it goes to Step 2.

Step 2 The heterogeneous parameters is updated by the parameters in the "Route Reply" packet. It is checked whether there is a forward routing table for the destination address of the "Route Reply"packet. If it "exists", it goes to Step 3. If it "does not exist", the forward routing table corresponding to the "Route Reply"packet is added, and then it goes to Step 3.

Step 3 It is checked whether the sequence number of the "Route Reply" packet is larger than the sequence number of the corresponding forward routing table entry. If it is "no", it goes to Step 4. If it is "yes", the original forward routing table information is cleared, and the new forward routing table information is added. The updated heterogeneous parameters are used to calculate the link load value and inserted into the route table, and then Step 7 is executed.

Step 4 It is checked if the sequence number and the corresponding forward routing table sequence number are equal and the "Route Reply" packet suitability is smaller than the minimum suitability of the forward routing table entry. If they are "no", the "Route Reply" packet is discarded. If they are "yes", it goes to Step 5.

Step 5 It is determined whether the forward path corresponding to the "Route Reply"packet exists. If it "exists", the forward path information is updated, and then it goes to Step 7. If it "does not exist", it goes to Step 6.

Step 6 If the number of paths is less than the maximum number of paths and there is no significant difference in length, the updated heterogeneous parameters are used to calculate the link load value, and the link load value is inserted into the path table. And then it goes to Step 7.

Step 7 The minimum fitness of the forward routing table entry is updated.

Step 8 If there are cached packets that need to be transmitted along the forward route, they are transmitted.

Step 9 If a node the intended recipient of the "Route Reply"packet, the packet is discarded. Otherwise, the reply source address is updated and the "Route Reply" packet is modified with node_type, node location information "X, Y, Z" and the updated heterogeneous parameter values "energy" "len" "speed" "distance", and then forwarded. The "Route Reply"packet is described in Algorithm 5.

Algorithm 5 Routing reply process

```
if(s_type==r_type||s_type==0||r_type==0)
then
  if (distance< reference distance) then
   if (s_destination ==r) then
   discard RREP/* energy, length, distance,
   speed*/
   update heterogeneous parameters count
   path_load
   if(route table==NULL)
     then add route table
end if
```

```
if (r_seqno < s_dst_seqno)
  path=NULL
  forward_path_insert ( nexthop, lasthop,hop-
  count+1, path_load)
else if ( r_seqno == s_dst_seqno &&r_adver-
      tised_hops > s_hop_count&&path_load
      <minFit )
 then
  if (r_firsthop==s_firsthop&&r_lasthop =
  =r_lasthop)update path
  else if (new_disjoint_path&& r_num_paths_
   < aomdv_max_paths_) forward_path_insert
  (nexthop, lasthop,hopcount+1, path_load)
  end if
   update minFit
if (s== r) then discard RREP
else
  update RREP
  foward RREP
  end if
 else discard RREP
 end if
else discard RREP
end if
```

2.3 Path selection

The calculation of the path load value is introduced in section 2.1. Heterogeneous characteristics such as data transfer rate and energy status are reflected in path_load. The path load on different paths can be represented by adding the extended field path_load to the path table.

The process of initiating a route is described in section 2.2, which creates multiple paths from the source node to the destination node, and the following pseudo-code describes how to select the path with the lowest path load from the multiple paths. The path selection is described in Algorithm 6.

Algorithm 6 Path selection

```
*p←rt_ path_ list. lh_ first
*path←NULL
min_ link_ load←0xffff
for( ; p;p=p->link. le_next)
 for( ;p;p=p->path_link. le_next)
  if(p->path_load<min_path_load)
   path←p
   min_ path_ load←p->linkload
  end if
 return path
```

During the routing process, if the path_load value in the path table is small, it means that the average remaining energy of the nodes on the path is small, or the load is small, or the data transmission rate is small, or the path communication distance is close. This means that the path is suitable for data transmission and the packet is chosen to be sent via this path. By using the data rate, energy and load of the communicating nodes as the routing criteria, the source node eventually establishes a communication path to the destination node, which usually has relatively sufficient energy and light load, as well as a relatively fast data rate and a near communication distance.

3 Performance Evaluation

In this section, we conduct simulation experiments using the NS-2 platform to compare and test the performance of CH-AOMDV in communication heterogeneous networks with traditional networks, compare and analyze the performance differences between the CH-AOMDV protocols and the AOMDV, AODV, and AEQAOMDV protocols. The packet delivery rate, average end-to-end delay, routing overhead, throughput and network life cycle are simulated and analyzed for different networks with different node distribution densities.

3.1 Environment and parameter setting test

Based on the CHMN, as shown in Table 1, various parameters are used to evaluate the performance of the proposed protocol and the built CHMN, such as the number of nodes, energy model, network type and so on.

Table 1 Simulation environment and protocol parameters

Parameter	Value
Protocol	CH-AOMDV, AOMDV, AODV, AEQAOMDV
Simulation time/s	1 000
Propagation model	TwoRay
Antenna	OmniAntenna
Channel	WirelessChannel
Energy model	Energy-Model

3.1.1 Packet delivery rate

The packet delivery rate is the ratio of the number of packets sent by the application layer to the number of packets received by the destination. This criterion indicates the quality of the routing protocol from the source to the destination. The greater the packet delivery rate, the better the performance of the routing protocol. and $P_{delivery}$ denotes the packet delivery rate, and is calculated as

$$P_{delivery} = \frac{P_R}{P_S} \times 100\% \tag{12}$$

where P_R represents the number of data packets that successfully reached the destination; and P_S the total number of packets sent by the source node.

3.1.2 Average end-to-end delay

The average end-to-end delay is the delay generated by the successful delivery of a data packet from the source node to the destination node. A_{delay} represents the average end-to-end delay and is calculated as

$$A_{delay} = \frac{1}{N}\sum_{i=0}^{N}(R_{time}(i) - S_{time}(i)) \tag{13}$$

where N represents the number of successfully transmitted data packets; $R_{time}(i)$ the time of arrival of the ith packet at the destination node; and $S_{time}(i)$ the time when the ith packet is sent.

3.1.3 Routing overhead

Routing overhead is the ratio of the number of routing control packets sent by all nodes in the network to the number of data packets successfully received by all destination nodes. This metric is used to indicate the utilization of effective network resources. The smaller the metric, the lower the additional network overhead and the higher the effective network resource utilization. N_{load} indicates the routing overhead, and is calculated as

$$N_{load} = \frac{P_C}{P_D} \tag{14}$$

where P_C represents the total number of the control packets sent by the node; and P_D the total number of the data packets received by the destination node.

3.1.4 Throughput rate

The throughput rate is the total number of bits successfully delivered over the network to the destination. It is a quality and performance indicator. High throughput means that fewer packets are dropped, or more data are transferred, during the transfer of data from the source to the destination. T_h express the throughput rate, and is calculated as

$$T_h = \frac{1}{T_{Rend} - T_{Rstart}}\sum_{i=0}^{M} R_{bytes}(i) \times 8 \tag{15}$$

where $R_{bytes}(i)$ represents the number of bytes of the ith packet that successfully reached the destination; M the total number of the packets received at the destination; T_{Rend} the time when the reception of data packets ends in the network; and T_{Rstart} the time when a data packet starts to be sent in the network.

3.2 Simulation results

According to Ref.[15], the initial value of the simulation environment area is set to 800 m × 800 m, and the simulate environment $x = y = k \times \text{area}^{1/2}$. Nodes are distributed in the simulation environment in a random manner. We modify the value of k to control the range of the simulation environment so that the node distribution density can be varied.

3.2.1 Validation of difference between CH-AOMDV under traditional and heterogeneous networks

Fig.8 displays the effect of different node distribution factor k on the packet delivery rate in different networks.

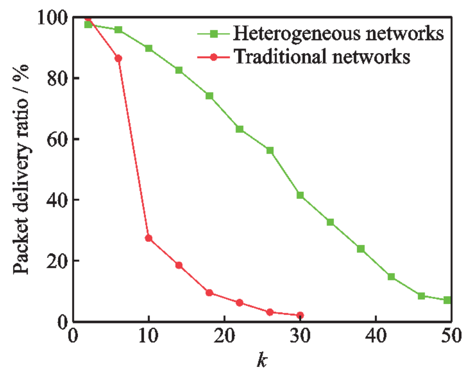

Fig.8 Relationship between k and data packet delivery rate

As the distance of node distribution increases, the packet delivery rate decreases from 98.56% to 2.11% for traditional networks and from 97.5% to 41.5% for heterogeneous networks when k=30, The reason is that as nodes become more and more distant from each other, route origination and path establishment become more and more difficult.

However, the presence of long range communication nodes such as Lora and WHCM in CHMN allows the network to work despite the long distance distribution of nodes. Whereas, conventional networks (WiFi is chosen here) have shorter communication distances and hence the performance decreases drastically when the distribution distance is far. When $k > 30$, the communication capability is completely lost and the limit of CHMN is much better than the conventional network.

Fig.9 shows the effect of k on the average end-to-end delay in different networks. As the distance of node distribution increases, the average end-to-end delay of the traditional network first increases to 55.08 ms and then decreases to 0.87 ms, while the average end-to-end delay of the heterogeneous network first increases to 12.84 ms and then decreases to 1.23 ms. The increase of k leads to more difficult communication and increasing average end-to-end delay. When the distance among the nodes exceeds a certain limit, the nodes are unable to establish a stable data transmission path and the average end-to-end delay decreases. In CHMN, Lora and WHCM support long distance communication. In addition, WHCM connects to long range communication nodes while prioritizing nearby short range communication nodes. This not only improves the packet delivery rate but also significantly reduces the average end-to-end delay.

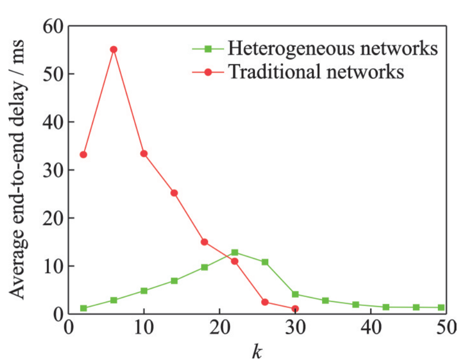

Fig.9 Relationship between k and average end-to-end delay

Fig.10 displays the effect of k on the routing overhead in different networks. The routing overhead of the traditional network increases from 12.84 each to 111.02 each, while that of the heterogeneous network increases from 1.04 each to 25.81 each. The presence of Lora nodes and WHCM in the CHMN allows the network to establish more stable long-range paths. Therefore, there is no need to initiate frequent route establishment requests, thus the routing overhead is redued.

Fig.11 shows k on the throughput rate in different networks.

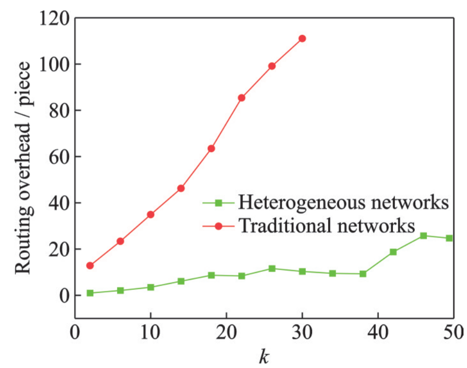

Fig.10 Relationship between k and routing overhead

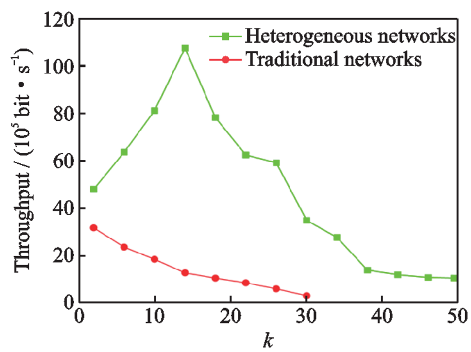

Fig.11 Relationship between k and throughput rate

As the distance of node distribution increases, the throughput rate of traditional network decreases from 44.7×10^4 bit/s to 3.14×10^4 bit/s, while that of the heterogeneous network increases to 107.42×10^4 bit/s and then decreases to 10.56×10^4 bit/s. Due to factors such as the decrease in packet delivery rate and the increase in node distribution distance, the throughput of traditional networks has significantly decreased. The WHCM has obvious advantages in this environment and maintains good performance.

3.2.2 Verification of impact of node distribution density on protocol performance

The effect of k on routing packet delivery rate is shown in Fig.12. As the node distribution distance increases, the packet delivery rate of CH-

AOMDV protocol decreases from 97.56% to 7.52%, AEQAOMDV protocol decreases from 90.84% to 8.78%, AOMDV protocol decreases from 93.16% to 3.22%, and AODV protocol decreases from 84.3% to 1.56%. The reason is that as the distance among the nodes increases. The possibility of establishing a path between the nodes decreases and the possibility of path interruption increases, which leads to the decrease in packet delivery rate.

The impact of *k* on average end-to-end delay is shown in Fig.13.

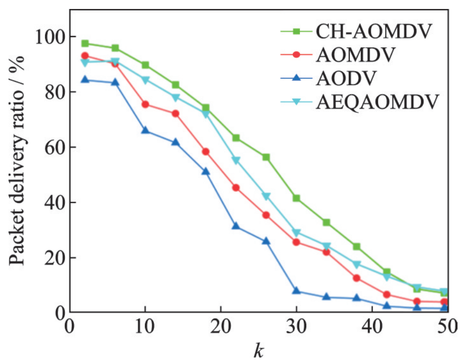

Fig.12 Relationship between *k* and date packet delivery rate

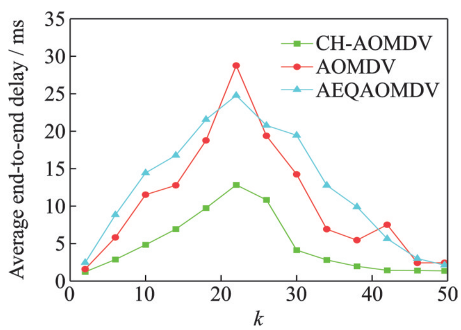

Fig.13 Relationship between *k* and average end-to-end delay

As the distribution distance among nodes increases, the average end-to-end delay of the CH-AOMDV protocol increases to 12.84 ms, then decreases to 1.37 ms, AEQAOMDV protocol increases to 24.77 ms, then decreases to 2.13 ms, and AOMDV protocol increases to 28.77 ms, then decreases to 2.45 ms. Although AOMDV can find multiple paths and reduce the link disconnection times, it cannot control the load of the communicating nodes on the link, especially for CHMN. The presence of highly loaded nodes on the same link negatively affects the network load due to the significant performance differences between different nodes.

The effect of *k* on routing overhead is shown in Fig.14. The routing overhead of the CH-AOMDV protocol increases from 1.12 each to 25.81 each as the node distribution distance increases, AEQAOMDV protocol increases from 1.74 each to 41.7 each, AOMDV protocol increases from 1.51 each to 40.13 each and AODV protocol increases from 0.89 each to 13.37 each. As the nodes are distributed further and further, the short-range communication nodes in the network gradually lose their advantages and the long-range communication nodes gradually gain advantages. Therefore, the path load value decision appears to change. In general, as the nodes are distributed further apart from each other, path building becomes more difficult. The nodes need to initiate route requests more frequently, thus the routing overhead is increased. The AODV algorithm does not require frequent initiation of routes because it is a single-path routing algorithm, so AODV performs better than AOMDV and AEQAOMDV in terms of routing overhead, and slightly better than CH-AOMDV, but this comes at the expense of other performance.

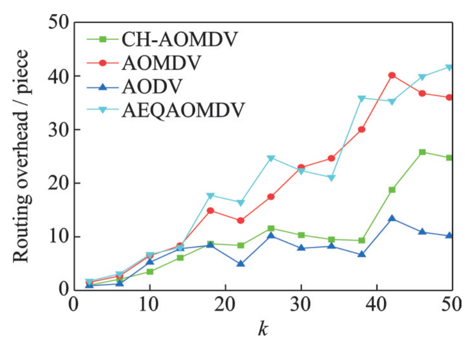

Fig.14 Relationship between *k* and routing overhead

The effect of *k* on the throughput rate is shown in Fig.15. As the node distribution distance increases, the throughput rate of CH-AOMDV protocol increases to 107.42 × 10^4 bit/s and then decreases to 10.55 × 10^4 bit/s, the AEQAOMDV protocol increases to 107.41 × 10^4 bit/s and then decreases to 10.55 × 10^4 bit/s, the CH-AOMDV protocol increases to 74.62 × 10^4 bit/s and then decreases to 5.01 × 10^4 bit/s, the AOMDV protocol increases to 70.75×10^4 bit/s and then decreases to 6.59×10^4 bit/s, and the AODV protocol increases to 48.36×10^4 bit/s and then decreases to 2.45×10^4 bit/s. Bluetooth and ZIGBEE, which have slower data transmission rate and shorter transmission distance, gradually lose their advantages in path selection. WiFi with faster data transmission and slightly longer transmission distance becomes the first choice for path selection, so its throughput rate shows an increasing trend at the beginning. As the communication distance increases, WiFi nodes also lose their advantage. Lora and WHCM with long communication distance, slow data transmission and small load gradually become the mainstream of data transmission in the network, so the data throughput rate gradually decrease.

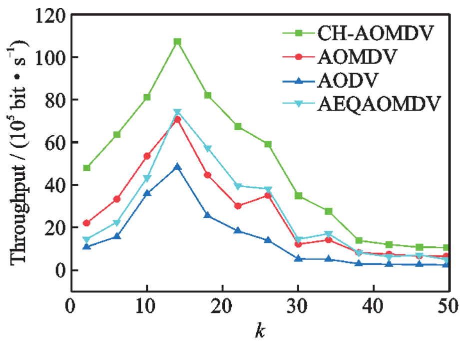

Fig.15 Relationship between k and throughput rate

In Fig.16, the network life cycle refers to the time of death of the first node in the network. As k increases, the network life cycle increases from 31.01 s to 285.1 s for CH-AOMDV protocol, from 37.09 s to 200.12 s for AEQAOMDV protocol, from 14.11 s to 117.98 s for AOMDV protocol, and from 11.87 s to 107.16 s for AODV protocol. As the nodes are distributed further away, path establishment and initiation become difficult, data interactions become less frequent, and therefore less energy is consumed, thus the network life cycle is increased. In comparison, both CH-AOMDV and AEQAOMDV significantly improve the network lifecycle because energy conditions are also used as the path selection criteria. When the energy is low, nodes are less likely to be selected as routes for establishing paths, thus the data flow is distributed to some nodes with better conditions. However, CH-AOMDV outperforms AEQAOMDV because it takes into account the heterogeneous characteristics.

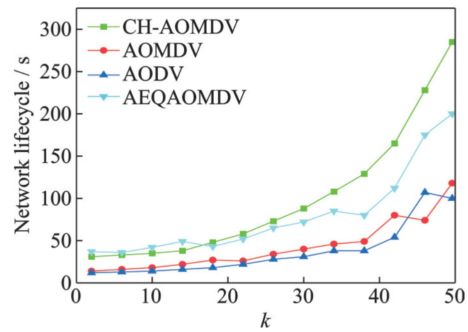

Fig.16 Relationship between k and network life cycle

4 Conclusions

We design a device based on STM32 that can communicate with multiple wireless communication nodes and build a CHMN with multiple communication protocols. In the CHMN, the path selection takes into account the heterogeneous characteristics of the nodes and selects a path with lower path load to achieve the complementary advantages of multiple wireless communication technologies and reduce the load balance of the network. Simulation and test experiments show that CH-AOMDV performs better on the CHMN compared with AOMDV, AEQAOMDV and AODV, and can effectively reduce the load balancing of the network. It improves packet delivery rate, reduces average end-to-end delay, increases throughput rate, and reduces routing overhead. In addition, CHMN performance is better than traditional networks.

In the future, the protocol proposed in this paper will be implemented on an established network tested to validate the test results. In addition, the protocol will be applied to IoT-based wireless sensor networks to validate its usefulness in application environments such as smart cities and smart agriculture.

References

[1] KIM H S, KIM H, PAEK J, et al. Load balancing under heavy traffic in RPL routing protocol for low power and lossy networks[J]. IEEE Transactions on Mobile Computing, 2017, 16(4): 964-979.

[2] WANG H, DENG A, HU C, et al. A SDN-based heterogeneous networking scheme for profinet and

modbus networks[C]//Proceedings of the 12th International Conference on ICT Convergence (ICTC)—Beyond the Pandemic Era with ICT Convergence Innovation. South Korea: ICT, 2021: 915-920.

[3] JIANG H, XIAO Z, LI Z, et al. An energy-efficient framework for internet of things underlaying heterogeneous small cell networks[J]. IEEE Transactions on Mobile Computing, 2022, 21(1): 31-43.

[4] CHENG J, CHENG J, ZHOU M, et al. Routing in internet of vehicles: A review[J]. IEEE Transactions on Intelligent Transportation Systems, 2015, 16(5): 2339-2352.

[5] WANG T, LUO H, ZENG X, et al. Mobility based trust evaluation for heterogeneous electric vehicles network in smart cities[J]. IEEE Transactions on Intelligent Transportation Systems, 2021, 22(3): 1797-1806.

[6] MA G, YANG Y, QIU X, et al. Fault-tolerant topology control for heterogeneous wireless sensor networks using multi-routing tree[C]//Proceedings of 2017 IFIP/IEEE Symposium on Integrated Network and Service Management(IM). [S.l.]: IEEE, 2017: 620-623.

[7] ZHANG K, MAO Y, LENG S, et al. Energy-efficient offloading for mobile edge computing in 5G heterogeneous networks[J]. IEEE Access, 2016, 4: 5896-5907.

[8] SHI C, LI Y, ZHANG J, et al. A survey of heterogeneous information network analysis[J]. IEEE Transactions on Knowledge and Data Engineering, 2017, 29(1): 17-37.

[9] QIU T, CHEN N, LI K, et al. How can heterogeneous internet of things build our future: A survey[J]. IEEE Communications Surveys and Tutorials, 2018, 20(3): 2011-2027.

[10] YAO Y, CAO Q, VASILAKOS A V. EDAL: An energy-efficient, delay-aware, and lifetime-balancing data collection protocol for heterogeneous wireless sensor networks[J]. IEEE-ACM Transactions on Networking, 2015, 23(3): 810-823.

[11] ZHENG K, ZHENG Q, CHATZIMISIOS P, et al. Heterogeneous vehicular networking: A survey on architecture, challenges, and solutions[J]. IEEE Communications Surveys and Tutorials, 2015, 17(4): 2377-2396.

[12] ZHANG D G, CHEN L, ZHANG J, et al. A multipath routing protocol based on link lifetime and energy consumption prediction for mobile edge computing[J]. IEEE Access, 2020, 8: 69058-69071.

[13] BHARDWAJ A, EL-OCLA H. Multipath routing protocol using genetic algorithm in mobile AD HOC networks[J]. IEEE Access, 2020, 8: 177534-177548.

[14] CHEN Z, ZHOU W, WU S, et al. An adaptive on-demand multipath routing protocol with QoS support for high-speed MANET[J]. IEEE Access, 2020, 8: 44760-44773.

[15] YANG X P, LI M J, WANG P, et al. An adaptive-aware energy and queue improvement of AOMDV[J]. International Journal of Communication Systems, 2019. DOI: 10.1002/dac.4054.

[16] AL-ZAIDI R, WOODS J, AL-KHALIDI M, et al. Next generation marine data networks in an IoT environment[C]//Proceedings of the 2nd International Conference on Fog and Mobile Edge Computing (FMEC). Valencia, Spain: [s.n.], 2017: 50-55.

[17] KAUR S, SHARMA T. Efficient load balancing using improved central load balancing technique[C]//Proceedings of the 2nd International Conference on Inventive Systems and Control (ICISC). Coimbatore, India: [s.n.], 2018: 1-5.

[18] SURBHI, OSHIN, BHATT M C. Performance evaluation of load balancing algorithms in Hadoop[C]//Proceedings of the 2nd International Conference on Computing Methodologies and Communication (ICCMC). Erode, India: [s.n.], 2018: 491-496.

[19] AL-ALI A R, ZUALKERNAN I A, RASHID M, et al. A smart home energy management system using iot and big data analytics approach[J]. IEEE Transactions on Consumer Electronics, 2017, 63(4): 426-434.

[20] QIU T, CHEN N, LI K, et al. How can heterogeneous internet of things build our future: A survey[J]. IEEE Communications Surveys and Tutorials, 2018, 20(3): 2011-2027.

[21] LEE H C, KE K H. Monitoring of large-area IoT sensors using a LoRa wireless mesh network system: Design and evaluation[J]. IEEE Transactions on Instrumentation and Measurement, 2018, 67(9): 2177-2187.

[22] KIM H S, KO J, CULLER D E, et al. Challenging the IPv6 routing protocol for low-power and lossy networks (RPL): A survey[J]. IEEE Communications Surveys and Tutorials, 2017, 19(4): 2502-2525.

[23] GUO S. Load balancing algorithm for heterogeneous wireless networks based on motion state estimation[C]//Proceedings of IEEE 9th International Conference on Information, Communication and Networks (ICICN). Xi'an, China: IEEE, 2021: 175-178.

[24] SAHNI Y, CAO J, ZHANG S, et al. Edge mesh: A new paradigm to enable distributed intelligence in internet of things[J]. IEEE Access, 2017, 5: 16441-

16458.

[25] LIAO C H, ZHU G, KUWABARA D, et al. Multi-Hop LoRa networks enabled by concurrent transmission[J]. IEEE Access, 2017, 5: 21430-21446.

[26] SAMUEL S S I. A review of connectivity challenges in IoT-smart home[C]//Proceedings of the 3rd MEC International Conference on Big Data and Smart City (ICBDSC). Muscat, Oman: IEEE, 2016: 364-367.

[27] GARRIDO-HIDALGO C, HORTELANO D, RODA-SANCHEZ L, et al. IoT heterogeneous mesh network deployment for human-in-the-loop challenges towards a social and sustainable industry 4.0[J]. IEEE Access, 2018, 6: 28417-28437.

[28] LEONARDI L, PATTI G, LO BELLO L. Multi-hop real-time communications over bluetooth low energy industrial wireless mesh networks[J]. IEEE Access, 2018, 6: 26505-26519.

Acknowledgements This work was supported in part by the Sichuan Provincial Key Research and Development Program (No.2021YFQ0011), and the National Natural Science Foundation of China (Nos.61961040, 61771089).

Authors Mr. **CAO Peng** received his B.E. degree in electronic information engineering from the School of Electronics and Information, Sichuan University, China in 2020. He is currently a graduate, majoring in information and communication engineering at the School of Electronics and Information Technology, Sichuan University, Chengdu, China. His current research includes wireless sensing networks and routing algorithms.

Dr. **NING Qian** received her B.E. degree in mechanical engineering from Xidian University, in 1990, her M.E. degree in mechanical engineering from University of Electronic Science and Technology, in 1997, and her Ph.D. degree from Sichuan University, in 2006. She is currently an associate professor in College of Electronics and Information Engineering, Sichuan University, Chengdu, China. Her current research interests include intelligent system, wireless AD HOC network, and complex problem programming.

Author contributions Mr. **CAO Peng** designed the study, complied the models, conducted the analysis, interpreted the results and wrote the manuscript. Dr. **NING Qian** optimized and improved the overall idea, feasibility, and method of the experiment. Prof. **HUANG Linyu** provided research ideas and improved the research direction. Prof. **CHEN Bingcai** provided guidance on experimental development and thesis writing. Mr. **TANG Chengwen** collected experimental data and assisted in designing experimental details. All authors commented on the manuscript draft and approved the submission.

Competing interests The authors declare no competing interests.

(Production Editor: ZHANG Bei)

无线异构Mesh网络搭建与负载均衡路由研究

曹 鹏[1]，宁 芊[1]，黄霖宇[1]，陈炳才[2]，唐成文[1]

(1.四川大学电子信息学院，成都 610065，中国；2.大连理工大学计算机科学与技术学院，大连 116024，中国)

摘要：在智慧城市等应用场景中，单一的无线通信技术无法覆盖复杂的网络环境，且不同的通信技术间无法相互通信，无法充分利用不同的无线通信技术。因此，本文设计了一种可以与多种无线通信节点进行通信的设备，称为无线异构通信模块(Wireless communication heterogeneous modules, WHCM)。WHCM被用作枢纽来构建通信异构网状网络(Communication heterogeneous mesh network, CHMN)。CHMN由具有多种无线通信协议的节点组成。本文提出了一种基于CHMN的按需多路径距离矢量(AD HOC on-demand multi-path distance vector, AOMDV)路由协议(名为CH-AOMDV)。CH-AOMDV能够在路由发起和路径建立过程中识别不同类型的通信协议，并比较CHMN中每个节点的通信距离、数据传输速率、能量和负载情况。NS-2平台仿真表明，当节点之间的分布距离变远时，在数据包传输速率、平均端到端延迟、吞吐率和路由开销方面，CHMN的性能优于传统网络。本文提出了一种实现CHMN的方法和相应的负载均衡路由协议CH-AOMDV，该协议优于其他3种协议，它提高了网络的生命周期、吞吐量和数据包分组投递率，降低了节点开销和平均端到端延迟，该协议对CHMN非常有用。

关键词：异构网络；STM32；需多路径距离矢量；路径负载；NS-2