2017 International Conference on Computer Network, Electronic and Automation (ICCNEA 2017)

WLAN Based Wireless Self-organization Link: Research and Realization

Yunhai Guo
Engineering Research Center of Digital Audio &Video
Ministry of Education
Communication University of China
Beijing, China
E-mail: guoyunhai@cuc.edu.cn

Zhengxiang Li
Engineering Research Center of Digital Audio &Video
Ministry of Education
Communication University of China
Beijing, China
E-mail: lizhengxiang@cuc.edu.cn

Abstract—**Communication is one of essential elements of our life. In some cases, we need wireless links to support data transmission such as television relay. But the existing GPRS and other network often need the support of power transmission line. And in the remote areas, the transmission rate of traditional network is very slow. In this paper, the development and realization of wireless self-organization link based on the wireless router and WLAN, managed by OpenWrt open source system is presented. It implements a strategy for Wireless Self-organization Link system, which can effectively solve the problem of automatic relay between wireless routers, prevent the establish of loopback in two routers, make multiple routers connect automatically, and set up a high-speed wireless data link in AP+Client mode.**

Keywords-WLAN; Wireless Self-organization Link; OpenWrt; Ad Hoc; WSOL

I. INTRODUCTION

SINCE 1971, Norman Abramson, a professor at the University of Hawaii, developed the world's first wireless computer communication network, ALOHA net. The Wireless communication gradually spread and the improvement of the data quantity is a higher requirement for the signal transmission. However, the laying of the signal cable is often a very complicated process, and in some cases, such as mountainous area or the disaster area, the cable is not allowed to lay. At this time, the signal transmission requires the use of wireless links. The existing Internet, GPRS network and so on often need the support of power transmission line, and in the remote areas, the transmission rate of traditional network is very slow, not enough to support the high-definition video signal transmission.

At present, the Wireless Mesh Network (WMN) which based on the IEEE802.11s and Wireless Sensor Network (WSN) are gradually becoming mature [1], [2]. Last year, the RSSI-based swarm robots passing narrow corridor based on Mesh network, and the adaptive video protection in large scale peer-to-peer video streaming which based on WMN has been developed [3], [4]. As the main problem of solving the mesh network interconnection, the data rate requirements are often slow [5]. Moreover, the design of the special wireless link transmission based on the router has not appeared.

In this paper, based on the OpenWrt system, a Wireless Self-organization Link (WSOL) strategy is implemented. This strategy is based on IEEE802.11 protocol which can be used to establish the wireless connection between the nodes effectively. And this method can implement the self-assembling and self-constructing of the link node, and effectively prevent the generation of self-loop.

The design can effectively improve the coverage of the wireless link, and provide support for the high speed data transmission.

II. WSOL SYSTEM OVERVIEW

The structure of WSOL used in this paper is based on WLAN and IEEE802.11 protocol [6], which is “one-by-one” link structure. As shown in Figure 1.

Source Terminal and Relays are routers use OpenWrt system and user program. This architecture is divided into three parts: Source Terminal (S), Relay (C), and User Terminal (U). The Source Terminal is directly connected to the Central Treatment System and is only used as an Access Point (AP); Relay uses AP+Client mode on the one hand as the client access to AP, on the other hand to build its own AP for the next Relay or User Terminal access. The User Terminal can be any device with the function of wireless access, such as mobile phones, cameras, etc.

In this structure, under the control of the program algorithm, the user deploys several Relays and a Source Terminal between User Terminal and Central Treatment System. S and Central Treatment System connected by cable. A wireless link between U and Central Treatment System can be automatically generated when we turn on the power supply of each router. And when a relay node failure, just replace the fault router, open a new router can re-establish links.

We use the OpenWrt system in the router. OpenWrt is a highly extensible GNU/Linux distribution for embedded devices (typically wireless routers). Its core is a Linux system with only the most basic functions. OpenWrt provides a fully writable file system and software package management, which is ideal for writing programs and even modifying systems according to the project requirements [7]. Therefore, we use OpenWrt as the system of routers and make use of its mature 802.11 protocol stack, write a series of program control routers to achieve the function of WSOL.

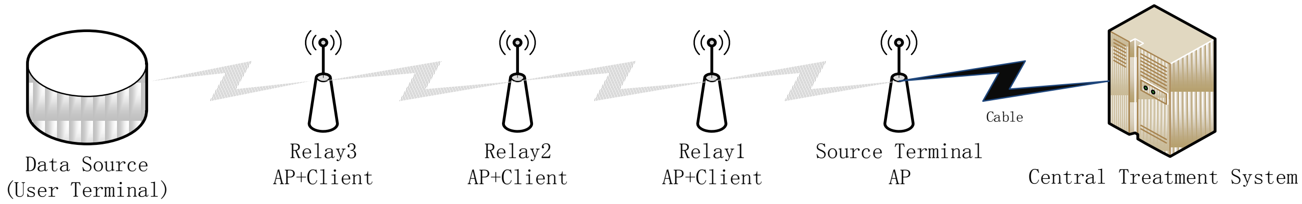

Figure 1. WSOL structure chart

UCI (Unified Configuration Interface) is a small utility written in C and is intended to centralize the entire device's configuration running OpenWrt. Examples are the main network interface configuration, firewall settings, wireless settings and remote access configuration. OpenWrt central configurations are split into several files located in the /etc/config/ directory [8]. We can edit the configuration files with a text editor or modify them with the command line by program “uci”. UCI configuration files are also modifiable through various programming APIs like Shell or C.

OpenWrt systems are widely used in various router brands, so this scheme can work on many routers. And this system can improve the speed and stability by updating the hardware or the built-in protocol.

III. System Design

A. *Summary of nodes*

The WSOL system includes the following key nodes: Source Terminal (only one), User Terminal (only one), and Relay (N nodes). The workflow of each node is as follows:

1) Source Terminal: The Source Terminal is directly connected to the Central Treatment System. After boot, the terminal launch its AP firstly, and then turnoff DHCP.

2) User Terminal: The User Terminal can be any device with the function of wireless access, such as mobile phones, cameras, etc. After boot, connect to the strongest AP around terminal.

3) Relay: Because the Relay uses AP+Client mode on the one hand as the client access to AP, on the other hand to build its own AP for the next Relay or User Terminal access, the “Discovery, Decision and Connect” algorithm is mainly applied to Relay. There are 3 kinds of connection of Relay (base on Fig.1) :

a) Relay1 connected to Source Terminal

b) Relay3 connected to Relay1

c) Insert Relay2 between Relay3 and Relay1 when Relay3 is connected to the Relay1 or Relay3 is disconnected from Relay1 because of the distance between Relay3 and Relay1 is too far.

B. *The “Discovery, Decision and Connect” algorithm*

- Step1: Relay1 startup firstly, then, scan APs around the Relay1, connect to the strongest AP. (Relay1 connected to Source Terminal).
- Step2: Relay3 startup firstly, then, scan APs around the Relay3, connect to the strongest AP. (Relay3 connected to Relay1).
- Step3: (At this time, insert Relay2 between Relay3 and Relay1) After Relay3 connected to the Relay1, when the signal strength of the newly inserted Relay2 is greater than that of the original connection (the signal strength of Relay1 detected by Relay3) 20dBm, the connection is going to be changed. When Relay3 making changes, Relay3 disconnect original connection firstly, then, wait 10s (prevent Relay2 gear into Relay3) and connect to Relay2 (the strongest AP).
- Step4: After Relay2 connected to an AP, check if it can communicate with Source Terminal. If not, break off the connection and select another AP to join. As shown in Fig. 2.

It should be noted that, a certain distance between Relays is necessary. The signal strength difference needs to reach 20dBm. The routers use static IP address.

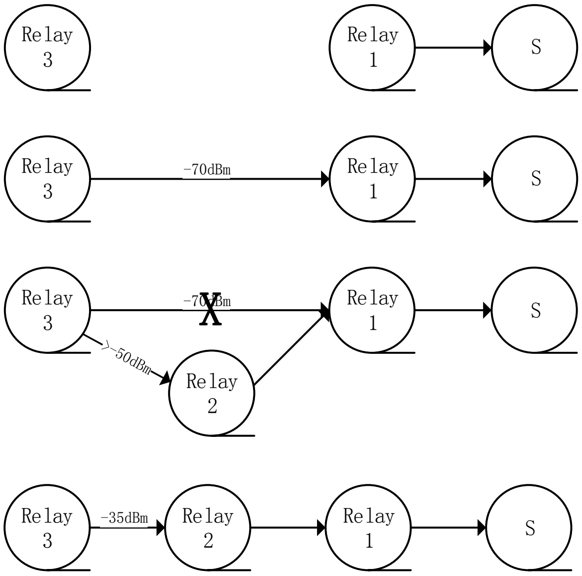

Figure 2. Algorithm flow of Relay

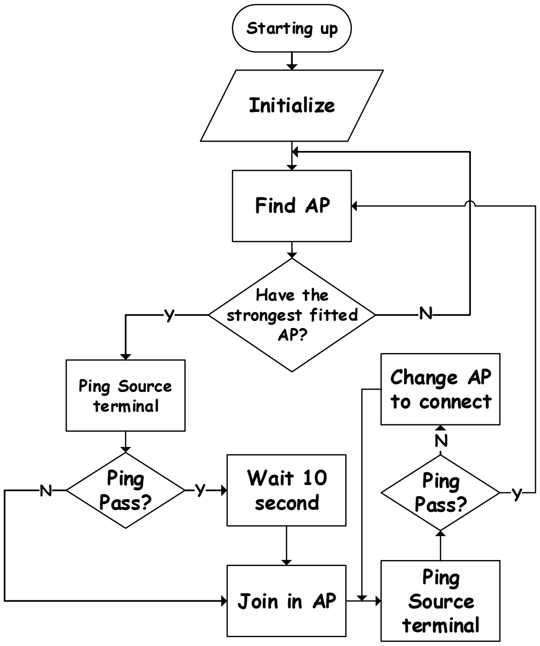

Figure 3. System flow chart of Relay

IV. Implementation

In this section, the implementation details will be explained in five parts: Setup AP and the initial work, find routers, choose routers, join in AP, decide and change router. The flow chart of relay terminal's behavior is as shown in Fig. 3.

A. Setup AP and the initial work

For the Relay, the first work after boot is turned on its AP and get ready to connect another device. Specifically, we must turn on the AP function of the device, and configure the channel, power, interface, SSID, encryption, LAN IP, then establish new interface "WWAN", configure firewall, shut down DHCP of LAN interface, and restart network service in the end.

Because the router uses the Linux kernel, there are many network configuration tools in Linux we can take advantage. The network interface, wireless functions and firewall functions are easily to configure by "uci" command. If these command lines are written into a Shell file, they can be run automatically.

B. Find Router

After the AP and the initial configurations are completed, the Relay needs to know other routers around it and find the fitted AP to connect. In this paper, we use a uniform method to name the SSID of AP. For example, we use a string "openwrt" connect up to the last segment of IP address. Such as [1] "openwrt12" means the LAN IP of this AP is "192.168.12.12". The advantage of this method is we can select the appropriate AP in complexity wireless circumstance.

C. Choose Router

After getting information of surrounding APs, this information needs to be selected again by router. The best AP will be selected which signal strength is the strongest one. Then, the information of the best AP will be written into file "bestrouter.txt" and this file will be supplied to AP access program.

D. Join in AP

After determining the best AP, the connection job is about to begin. The information of this AP (SSID, MAC address) will be read into a program, which can use "uci" command to configure the IP address, gateway address, interface mode and MAC address of interface "WWAN". Then, restart network, the connection job is done.

In this paper, the AP+Client mode is used in Relay. We utilize the interface LAN as the interface which can be accessed by another router, and the interface "WWAN" as the interface to connect another router. The LAN mode is "ap", the WWAN mode is "sta".

E. Change Router

Aiming at the third connection cases of Relay, we need to detect the current communication between the Relay and Source Terminal. To be specific, after Relay connecting to the best AP, check the communication with Source Terminal, break off the connection and select another AP to join if can't communicate with Source.

After all these steps, a reliable link will be established between Source Terminal and the last Relay. We tested this link on a board based on *AR9341*[2], and *iperf*[3] program on OpenWrt system to send data between Relays.

After link relay, no significant fluctuations in average bandwidth, multi-hop link runs well.

V. Conclusion

In this work, a WSOL system using OpenWrt build-in system has been presented and tested. By compiling C and Shell programs and running in wireless routers, it is able to establish data links between two points automatically. The average bandwidth between two points is no significant fluctuations. However, this design also has its shortcomings. Because the link update is based on the communication with the Source Terminal, the update speed is slow, encounter a collision of the test signal will bring wait time. So this work is not applicable for mobile wireless network. In the future work, we plan to improve the algorithm to enhance the mobility and optimize the link detection scheme to reduce the re-build connection time.

ACKNOWLEDGMENT

This work was supported by national sci-tech support plan, digital media technology development. (2015BAK05B00, 2015BAK05B01).

REFERENCES

[1] IEEE Standard for Information Technology--Telecommunications and information exchange between systems--Local and metropolitan area networks--Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 10: Mesh Networking," in IEEE Std 802.11s-2011 (Amendment to IEEE Std 802.11-2007 as amended by IEEE 802.11k-2008, IEEE 802.11r-2008, IEEE 802.11y-2008, IEEE 802.11w-2009, IEEE 802.11n-2009, IEEE 802.11p-2010, IEEE 802.11z-2010, IEEE 802.11v-2011, and IEEE 802.11u-2011) , vol., no., pp.1-372, Sept. 10 2011.

[2] Yick, J., Mukherjee, B., & Ghosal, D. (2008). Wireless sensor network survey. Computer Networks, 52(12), 2292–2330.

[3] Hattori, K., N. Tatebe, T. Kagawa, Y. Owada, L. Shan, K. Temma, K. Hamaguchi and K. Takadama. "Deployment of Wireless Mesh Network Using Rssi-Based Swarm Robots Passing Narrow Corridor by Movement Function Along Walls." Artificial Life And Robotics 21, no. 4 (2016): 434-442.

[4] Ghaeini, Hamid Reza, Behzad Akbari, Behrang Barekatain and Alicia Trivino-Cabrera. "Adaptive Video Protection in Large Scale Peer-to-Peer Video Streaming over Mobile Wireless Mesh Networks." International Journal Of Communication Systems 29, no. 18 (2016): 2580-2603.

[5] Chakraborty, Sandip and Sukumar Nandi. "Data Rate, Path Length and Network Contention Trade-Off in Ieee 802.11s Mesh Networks: A Dynamic Data Rate Selection Approach." Computer Networks 91, (2015): 225-243.

[6] Serrano, P., P. Salvador, V. Mancuso and Y. Grunenberger. "Experimenting with Commodity 802.11 Hardware: Overview and Future Directions." Ieee Communications Surveys And Tutorials 17, no. 2 (2015): 671-699.

[7] Anonymous, "OpenWrt", https://openwrt.org/, accessed by 27 April 2017.

[8] Anonymous, "UCI", https://wiki.openwrt.org/doc/uci, accessed by 27 April 2017.