Distributed flow controller for mobile ad-hoc networks

Gheorghe Mulec*, Radu Vasiu*, Flaviu Frigura-Iliasa**

* POLITEHNICA University of Timisoara, Faculty of Electronics and Telecommunications Engineering
** POLITEHNICA University of Timisoara, Faculty of Electrical and Power Engineering, Romania

george.mulec@rdslink.ro, radu.vasiu@cm.upt.ro, flaviu.frigura@et.upt.ro

Abstract—A mobile ad-hoc network (MANET) is a self-configuring, self-organized, and infrastructureless network of mobile stations, connected by radio links. The challenge to ensure network’s operation means to continuously maintain updated information about the network’s topology, the characteristics of the radio links and security at all levels. Due to the decentralized nature of the network, this is quite difficult. Distributed coordination and control of MANET has always been a challenge for network designers. This paper presents a proposal for a distributed flow controller, a new approach regarding the network architecture for MANET. From the perspective of the current approaches, the entire network has to be treated as a whole, from the point of view of the management, control and data plane. Our proposed approach, based on the paradigm of SDN (Software Design Network), implies a clear separation between the data plane and the control plane. This separation leads to greater flexibility on the control network, to the possibility of dynamic reconfiguration, and to the improvement of its security.

I. INTRODUCTION

Mobile ad hoc network (MANET) are wireless networks built "on-the-fly" by mobile stations without any support of fixed infrastructure.

Communication is directly between nodes or through intermediate nodes. The main characteristic of mobile ad hoc network is the dynamic change of network topology [1]. The mobile stations can join and disjoin the network in an unpredictable manner, due to different types of reasons, that leads to impossibility in achieving a central administration.

The advantages of such type of network are fast deployment, flexibility, cost (no need to achieve an infrastructure) and obviously the mobility. Literature review present different fields where mobile ad hoc networks can have an outstanding contribution [2].

We consider that mobile ad hoc networks are vital for achieving communications on areas where fixed communication infrastructure (wireless or wired) is no longer available or on isolated areas where fixed infrastructure has not been developed. As well, mobile ad hoc network are vital on military missions, where communication has to be made on a small area that has no connectivity with a fix infrastructure, for protecting the members that are on military mission.

In this paper, we propose a new approach of mobile ad-hoc network architecture from the point of view of the network coordination [3][4][5]. Our approach is based on the paradigm of SDN (Software Design Network) that implies a clear separation between the control plane and the data plane (see Fig. 1).

The main advantage of this approach is that it defines a central point for network’s coordination and configuration, that allows easy dynamic reconfiguration of the entire network, as well as fast response against different type of security threats. The entire network can be dealt in a unitary manner, from all aspects: traffic engineering, quality of service, security issues, etc. All this aspects, including security services have a major impact on network’s performances [6].

With traditional software design for network architectures, the control plane is formed by the software controller, while the data plane is formed by the flow switches. The connection between them will take place through an encryption channel.

From the phase of initial configuration, the switches know the controller’s address and establish a secure connection with them following the booting process.

In other words, the stations know, prior to join the network, who are the network controllers (or controller), even in distributed control platforms [7][8][9].

We implemented, for mobile ad-hoc networks, a distributed controller as a virtual coordination center [10]. Due to the dynamic characteristics of a mobile ad-hoc network, in such environment doesn’t exist a predefined controller and their addresses are known by all stations. All stations may act as main or secondary controller, that depend on the current network topology. Our distributed controller will be able to communicate with equipment that supports the open-flow specification [11][12].

It is only needed a little patch for the data plane to dynamically finding and configuring the remote controllers’ addresses.

In mobile ad hoc networks, the stations play multiple roles at the same time: provide radio connectivity, act as a...
router, firewall, authenticator, etc. As shown in fig. 2, the control and data plane coexist in the same hardware device, but are treated in the SDN approach. The data plane will function independently, but the control plane will act in a unitary manner.

The local controller doesn’t have local attributions (coordinating the local data plane), but it takes part to a virtual coordination center, or virtual control plane, that has responsibility for the network’s dynamic administration.

II. DISTRIBUTED CONTROLLER

The main objectives (functions) of the controller are:
- AAA (authentication, authorization and accounting) services for the network nodes,
- centralized network traffic control,
- achievement of network security policy.

For data consistency in the control plane, only one of the stations will act as primary controller at a time. Virtual control plane is formed by all active controllers from the network – fig. 3.

The selection mechanism establishes the order of controller priority and how many controllers must be active in a given network. It is very important for the active controllers and especially for the main controller to be “in the network center”, in other words the controller that has the minimum length of path to all active network nodes will be selected as the main controller. In order to calculate this, in the first step we need to have the shortest length path between all nodes of the network. For this, we used the Floyd-Warshall algorithm, that calculates the all-pairs shortest paths for graph elements as a matrix of n x n elements, where n is the number of the mobile station [13][14]. Then, we transformed the resulted matrix in a bidirectional array that contains an upward ordered list with controllers.

After that, we established the number of controllers from the network. Always, there will be at least two controllers in the network. For calculating the controller’s number, we have to know the degree of dispersion of the network. This information will be tacked from network information base, available at all the active controllers.
Depending on the degree of dispersion and the number of nodes in the network, it sets a number of active controllers that will operate the network.

IV. MATHEMATICAL MODEL OF THE SELECTION MECHANISM

In order to implement the mechanism for selecting the controllers, we built the mathematical model shown below in fig. 4, as a graph with \( n \) elements. Starting from the Floyd-Warshall algorithm, we have defined \( G = (V, E) \), a directed graph with real edge weights and no negative weights.

Let’s consider the weight function \( w : E \rightarrow \mathbb{R} \), and \( V = \{1, 2, 3, ..., n\} \), an array with \( n \) elements.

We have the weights between every node of the graph as a matrix with \( n \times n \) elements, where \( n \) is the number of nodes of the network and \( c_{i,j} \) are the weights between the node \( i \) and the node \( j \) defined as follow:

\[
 w_{i,j} = \begin{cases} 
 c_{i,j}, & \text{if } i \neq j \text{ and } (i, j) \in E \\
 \infty, & \text{if } i \neq j \text{ and } (i, j) \notin E 
\end{cases}
\] (1)

To calculate the minimum weights between any two points of the graph, we defined \( D^{(k)} \), a matrix (2) like \( w \) (\( n \times n \) elements), that gives the shortest weight between \( v_i \) and \( v_j \), from the \( V \) array, as:

\[
 D^{(k)}_{i,j} = \min(D^{(k-1)}_{i,j}, (D^{(k-1)}_{i,k} + D^{(k-1)}_{k,j}))
\] (2)

This recurrence function can be easily implemented by means of a computer programming language. The resulted matrix contains the shortest weight between any two points of the graph and is expressed by the formula (3).

\[
 D_{i,j} = \begin{cases} 
 0, & \text{if } i \neq j \\
 d_{i,j}, & \text{if } i \neq j 
\end{cases}
\] (3)

where \( d_{i,j} \) represents the shortest weight between \( v_i \) and \( v_j \) calculated with the Floyd-Warshall algorithm.

Further, we present in some details, the algorithm for establishing the list of active controllers in the network.

Now we have to calculate an upward ordered list for all edge of the graph, considering the shortest weights matrix. We define \( z_k \), a function that calculates the amount of weights for one node to others (4).

\[
 z_k = \sum_{k=1}^{n} \left( \sum_{i=1}^{n} w(i,k) + \sum_{j=1}^{n} w(k,j) \right)
\] (4)

and a \( Z_{(k,l)} \) matrix, defined as follow:

\[
 Z_{(k,l)} = \begin{cases} 
 z_k, & \text{if } l = 0; \\
 k, & \text{if } l = 1;
\end{cases}
\] (5)

The upward list is given by the following formula:

\[
 S_{(i,j)} = \text{sort}(Z_{k,0})_{k=1}^{n}
\] (6)

Now, we have an upward list with all the nodes of the graph (6). The \( S_{0,1} \) is the node with the shortest weights towards all nodes of the graph. It is considered to be in a “center” of the graph (main controller in our implementation). To calculate the number of secondary controllers of the network, it is necessary to calculate the scattering of the network with the following formula:

\[
 L = \sum_{i=1}^{n} \sum_{j=1}^{n} k, \text{ when } k = \begin{cases} 
 1, & \text{if } w(i,j) = \infty \\
 0, & \text{otherwise}
\end{cases}
\] (7)

The number \( L \) represents the amount of missing links between the nodes in graph (7). For a “linear” network, this value is maximum \( \frac{n(n+1)}{2} \), where \( n \) is the number of nodes. For full mesh network, this value is 0;

After multiple tests, we established a number of secondary controllers as:

\[
 m = \frac{L}{n} + 1
\] (8)

And now we have the final array with the list of active controllers:

\[
 C_j = \begin{cases} 
 S_{(i,j)}_{i=1}^{m}, & j = 1; 
\end{cases}
\] (9)

where \( m \) is calculated from (8), \( c_1 \) is the main controller, and \( c_2 ... c_m \) are representing the secondary controllers.

V. IMPLEMENTATION

For testing the proposed model, we built a platform comprising seven wireless devices. (Linksys WRT54GL) with Openwrt Backfire 10.03 [15]. We wanted to do testing on a real platform with all the constraints this generates, as represented in fig. 5 (limited processing
power, limited memory, limited storage, variable level of radio signal, etc.) [16]. Taking into account the above remarks, our distributed controller has been written in LUA language, a fast, lightweight and an embeddable scripting language [17].

We tested the mechanism for controllers’ election, in different types of scenarios, including processes generated by joined and disjoined a standby controller, a secondary controller or a main controller to the network. Also we tested the active controller auto configuration, due to the changes in the level of radio links between nodes. Fig. 6 presents the amount of time necessary for the controllers’ selection process, on the basis of the nodes number. We can see that the time increases with the growth of the number of nodes and that the growth is logarithmic.

**Figure 5. Test bed platform**

**Figure 6. Convergence election time**

An important aspect for fast convergence of network information base is to maintain at minimum the time of the selection process. In this sense, the hello packets between active controllers and hello packets between standby controllers and active controllers must be small enough for fast convergence and large enough to maintain at reasonable value the management traffic in the network. Therefore the Hello time interval is not a fixed value and it is a dynamic configurable parameter of the network. For nodes from data network path with high value of traffic, the value of the Hello time interval will be lower. Also, the time interval Hello message between the main controller and the “first secondary controller” will be lower. During the selection process, network’s control is provided by the active controllers. In case that the main controller leaves the network, the "first secondary controller" takes responsibilities of the main controller and triggers the selection process.

**VI. Conclusions**

We proposed a new approach to mobile ad hoc network management, based on the SDN paradigm.

The mechanism for the selection of the network controllers is very important for network’s functionality. SDN allows a unified approach to the management of the entire network and treats it as a monolith. To ensure network’s operation it is very important to maintain an adequate number of active controllers on the network. All controllers can serve as main controller or as secondary controllers, due to the dynamic nature of network’s topology.

Creating a distributed controller by grouping all controllers in a virtual control center, in order to implement a mechanism for selecting and maintain active network’s controllers, is the first step to ensure a robust and flexible network, able to promptly respond to any topology change.

**REFERENCES**


