An improved routing protocol Ad-AODV Based on AODV

Zuhong Feng [1, a], Long Wang [1,b] and Xiujuan Gao [1,c]

[1,]Beifang University of Nationalities Yinchuan 750021,China

[a]fengzh_1@163.com, [b]wangwanglong1987@163.com, [c]449086157@qq.com

Keywords: Ad Hoc network; AODV routing protocol; energy model; load balancing

Abstract. In Ad Hoc network, AODV routing protocol is an on-demand routing protocol of relative maturity and extensive application. Due to the fact that AODV routing protocol doesn't consider the residual energy and the load situation of the nodes when choosing routes, its efficiency declines sharply in the case of the high load and fast moving velocity. To solve the above problems, we propose an improved protocol Advanced-AODV (Ad-AODV) of AODV routing algorithm based on a strategy of energy model and load balancing. When Ad-AODV routing protocol performs the routing request, it will consider the residual energy and the load situation of the nodes. According to the simulation results, the Ad-AODV routing protocol improves the efficiency of Ad Hoc network, and the packet delivery ratio, lowers the average end-to-end delay and reduces the routing load.

Introduction

Ad Hoc network is a mobile, multi-hop, Self-discipline mobile system, each mobile node is both a host and a router[1]. Ad Hoc network without a fixed network infrastructure can provide communication links for the nodes. Its range of applications is extremely broad.

AODV routing protocol[2] was presented based on DSDV[3] and the improvement for on-demand routing mechanism in DSR[4]. It hasn't only the advantages of DSDV and DSR[5], but also its own characteristics, which makes it become a widely used routing protocol. In fact, if the network has light load, AODV routing protocol runs effectively. However, its performance becomes worse sharply in the case of high load. This is because when choosing a route, AODV routing protocol only pays attention to the path that is the shortest without considering the energy and load of the nodes. So when AODV routing protocol chooses the routes, it is very necessary to consider the residual energy and the load situation of the nodes[6].

In this paper, we use the residual energy and the load level of the nodes as metrics to propose an improved protocol "Ad-AODV routing protocol" which is based on the energy model and load balancing of AODV routing protocol.

AODV routing protocol

AODV routing protocol evolves from DSDV. It not only adopts the routing discovery and routing maintenance mechanism of DSR, but also makes use of the hop-by-hop routing and sequence number of DSDV. The detailed steps of AODV routing protocol in the routing discovery stage are as follows[7]:

1) When the source node has data packets to send, it searches for its routing cache table. If there is an entry that indicates a route to the destination node in the routing cache table, the source node directly sends the data packets. Otherwise, the source node directly broadcasts a RREQ. Then, jump to 2).

2) When an intermediate node receives a RREQ, it does the following operations.

① If the node isn't the destination node, then jump to ③. Otherwise, jump to ②.

② If the destination node is the first time to receive the RREQ, the destination node will insert the value of RREQ's request source address field and the value of RREQ's request broadcast ID field into the broadcast ID cache table, then establish the reverse route with its last hop node. Then, jump to ⑦. Otherwise, the destination node will discard the RREQ. Then, jump to ⑧.

③ It searches its broadcast ID cache table according to the values in the source address field and broadcast ID field of the RREQ. If there is a broadcast ID cache entry which has the same values of request source address field and request broadcast ID field as those of the RREQ, jump to ⑥. Otherwise, jump to ④.

④ Since the node is the first time to receive the RREQ, it will insert the value of RREQ's request source address field and the value of RREQ's request broadcast ID field into the broadcast ID cache table, then establish the reverse route with its last hop node.

⑤ If the node has a route to reach the destination node, then jump to ⑦. Otherwise, the node will randomly generate a delay and after the delay has arrived, the node will broadcast a RREQ. Then, jump to 2).

⑥ Since the node received a RREQ before. The node will discard the received RREQ which comes from its last hop node, rather than establish the reverse route with its last hop node. Then, jump to ⑧.

⑦ The node sends a RREP to the source node. The establishment of the route is completed.

⑧ The node does nothing.

Ad-AODV routing protocol

According to the working steps of AODV, in order to reducing the time of route establishment, AODV doesn't consider the residual energy and load level of the nodes in the routing discovery stage, which makes the selected routes instable. Compared with AODV, Ad-AODV routing protocol considers the residual energy and load level of the nodes in the routing discovery stage. It has the following improvements: in the routing discovery stage, the generation of the broadcasted RREQ delay of a node is based on the residual energy and load situation of the node. The delay's calculation is based on energy and load as parameters, using the certain mathematical model. The work steps of Ad-AODV routing protocol is basically the same as AODV 's, in addition to the fifth small step of the second step of AODV 's work steps is changed to " ⑤ If the node has a route to reach the destination node, then jump to ⑦. Otherwise, the node will take its energy and load as parameters to calculate the delay using the corresponding delay formula. After the delay has arrived, the node will broadcast the RREQ. Then, jump to 2).", the rest are the same.

Ⅰ. Next we will introduce the mathematical model which is used in the routing discovery process of Ad-AODV routing protocol in details.

(a) The Residual Energy Of The Node

We hope that the residual energy of the nodes in the selected route is sufficient, so that when the source node transmits data packets, the residual energy would decrease too much to rapidly depleted. Therefore, when the source node makes a route request, any intermediate node who has received the RREQ will make a delay according to the amount of the current residual energy of the node in order to make the node that has enough residual energy be preferentially selected. The formula[8] to calculate the selected energy consumption model is:

$$Energy=Power\times Time. \quad (1)$$

this is, when a node sends or receives a data packet, the energy which is consumed by the node is determined by the transmission power or the received power of the node and the time needed for handling the data packet. Among the time needed for handling a data packet is:

$$Time=8\times Packetsize / Bandwidth. \quad (2)$$

Therefore,

$$E_{tx}=P_{tx}\times 8\times Packetsize / Bandwidth. \quad (3)$$

$$E_{rx}=P_{rx}\times 8\times Packetsize / Bandwidth. \quad (4)$$

here, P_{tx} is the transmission power, P_{rx} is the receiving power. E_{tx} and E_{rx} indicate the amount of energy consumed by a node.

The total energy consumed by a node forwarding a data packet E_{full} is:

$$E_{full}=E_{tx}+E_{rx}. \quad (5)$$

We assume that the total energy at node "a" is Ea_{full}, and the current residual energy is $Ea_{current}$ that obeys uniform distribution on [0, Ea_{full}], the energy delay weights Wa_e at the node "a" is

$$Wa_e=1-Ea_{current} / Ea_{full}. \quad (6)$$

We know $0\leq Wa_e\leq 1$;

(b) The Load Of The Node

The current load level of a node has the certain influence to localized routing in the network, so it is very necessary to put the current load level of a node as the basis of routing and forwarding packets.

Assume it can hold up to Qa_{max} packets in the buffer queue of the node "a" and the current packet number is $Qa_{current}$ in the buffer queue of the node "a", the load delay weights Wa_q of the node "a" is

$$Wa_q=Qa_{current} / Qa_{max}. \quad (7)$$

We know, $0\leq Wa_q\leq 1$, which also shows that the bigger Wa_q is, the heaver the load of the node "a" is.

(c) The Delay Model

According to the above considered two metrics: the current residual energy of the nodes and the load of the nodes. The formula to calculate the delay time T is

$$T=(\alpha\times Wa_e+\beta\times Wa_q)\times Tc. \quad (8)$$

where, "α, β" are two constants and $0\leq\alpha\leq 1$, $0\leq\beta\leq 1$. Tc is a delay constant. In terms of selecting values of the delay constant, the delay constant cannot be too large nor too small.

Ⅱ. The Design and Implementation of the New Algorithm

According to the current residual energy and the load level of the nodes, the routing algorithm of Ad-AODV routing protocol delays in the routing request process.

(a) The Design Of The New Algorithm

1）Energy attributes

we add the energy attribute E in the routing table entry of a node. We assume that the total energy of the node "a" is Ea_{full}, the current residual energy is $Ea_{current}$ which obeys uniform distribution on [0, Ea_{full}].

A node in the network has two states: living condition and death state. If the current residual energy of a node is greater than 0J, its condition is living. Otherwise its condition is dead.

2）Load attributes

When we configure a mobile node, we add the load queue which is a queue between LL layer and MAC layer. We can consider it as a MAC queue. However, the defined queue and the parameter "limit_" in aodv_rqueue.cc of AODV routing protocol refer to the queue of AODV protocol layer, which are limited to use in AODV routing protocol. Therefore, the two queues have two different concepts and have no relationship each other.

(b) The Implementation Of The New Algorithm

1）By default, NS2 does not automatically enable energy model, so we need to manually enable it.

In the .tcl script, we add the following code:

```
$ns_ node-config
    –energyModel   EnergyModel  \    –initialEnergy   $val(initp)        \
    –rxPower          $val(rxpower)   \    –txPower         $val(txpower)
```

where, "-energyModel" is energy model, "-initialEnergy" is initial energy, "-rxPower" is the amount of energy consumed by a node when it receives a data packet, "-txPower" is the amount of energy consumed by a node when it sends a data packet.

2）In the header file "aodv.h", add the following code:

```
#include <common/mobilenode.h>    #define   α    0.5
#define   β    0.5    #define   Tc    0.01
```

where mobilenode.h is the header file of mobile nodes. According to the formula (8), α，β are two constants and $0 \le \alpha \le 1$, $0 \le \beta \le 1$. Tc is a delay constant.

Then, add the object pointer of mobile nodes and the object pointer of energy model of mobile nodes in the class AODV: public Agent:

```
MobileNode *node;   EnergyModel *em;
```

3）In the source file "aodv.cc", add the following code in the function void forward(aodv_rt_entry *rt, Packet *p, double delay):

Firstly, define the variables:

```
double   init;    int   qlen, qlim;    double   remain;    double   T;
```

where, "init" stores the initial energy of mobile nodes, qlen stores the current queue length of mobile nodes, qlim stores the maximum buffer queue length of mobile nodes, remain stores the current residual energy of mobile nodes and T stores the calculated delay.

Secondly, in the broadcasting conditional statement "if (ih->daddr() = = (nsaddr_t) IP_BROADCAST)", we add the following delay code:

```
em=node->energy_model();     remain=em->energy();
init=em->initialenergy();       qlen=ifqueue->length();
qlimit=ifqueue->limit();
```

where, the function "energy_model()" calculates the object pointer of energy model, the function "energy()" calculates the current residual energy of mobile nodes, the function "initialenergy()" calculates the initial energy of mobile nodes, the function "length()" calculates the current queue length of mobile nodes, and the function "limit()" calculates the maximum buffer queue length of mobile nodes.

Thirdly, we calculate the delay T using the formula (8).

```
T=(α*(1-remain/init)+β*(qlen/qlim))*Tc;
```

Replace the random delay delivery time "0.01*Random::uniform()" in the original send function "Scheduler::instance().schedule(target_ ,p ,0.01*Random::uniform())" with T, i.e., Scheduler::instance().schedule(target_ , p ,T).

Finally, we need to recompile NS2, to run the modified code.

Simulation Results And Analysis

Under the Linux system in a virtual machine, we used NS2 to simulate the network environment and carried on a contrast analysis on the performance of Ad-AODV protocol and AODV protocol. We used the three most commonly used quantitative indicators to judge the performance of the routing protocol: Packet Delivery Ratio, Average End-to-End Delay, and Routing Load.

The experimental simulation results are show in Fig. 1, Fig. 2, Fig. 3. The following analysis on the simulation results are based on three indicators of network performance evaluation:

Table 1. The Performance of the main parameters of the simulation experiment

parameter	set value
The number of mobile nodes	20、40、60、80
Simulation scenario	500m*500m
Communication radius	250m
Maximum buffer queue	50 packets
Residence time of nodes	0、5、10、15、20、25、30
Maximum mobile speed of nodes	10 m/s
The rate of sending data packets	15 packets/s
The number of maximum communication meetings	10
Data flow	cbr
Parameters used in the paper	α=β= 0.5，Tc = 0.01s
Simulation time	200s

(1) Packet Delivery Ratio

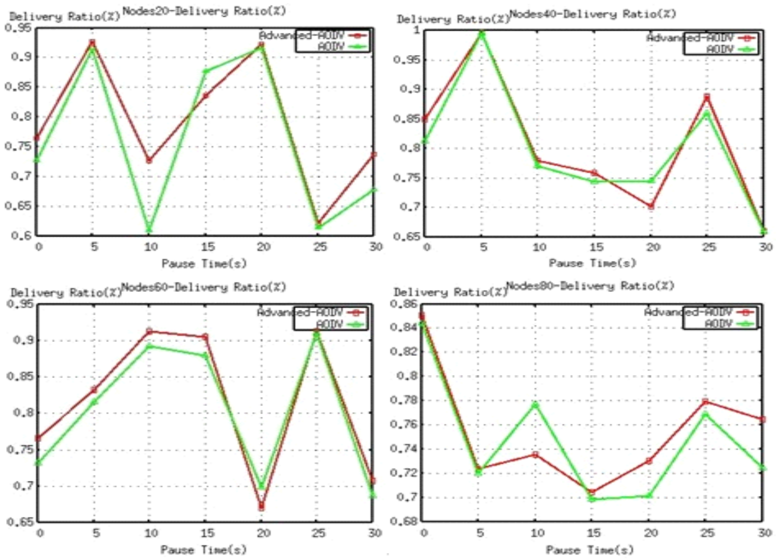

Fig. 1 The comparison of Packet Delivery Ratio

Packet Delivery Ratio reflects the stability of the selected route, the higher it is, the better the stability of the routing protocol is. Fig. 1 shows the simulation results of the packet delivery ratio of Ad-AODV and AODV in the different residence time of the different number of nodes. According to the calculation formula of the average growth rate, in the case of the simulation of twenty nodes, the delivery rate of Ad-AODV is an average of 2.4% higher than AODV 's. In the case of the simulation of forty nodes, the delivery rate of Ad-AODV is an average of 1.5% higher than AODV 's. In the case of the simulation of sixty nodes, the delivery rate of Ad-AODV is an average of 2.1% higher than AODV 's. In the case of the simulation of eighty nodes, the delivery rate of Ad-AODV is an average of 2.0% higher than AODV 's. Ad-AODV optimizes the network performance.

(2) Average End-to-End Delay

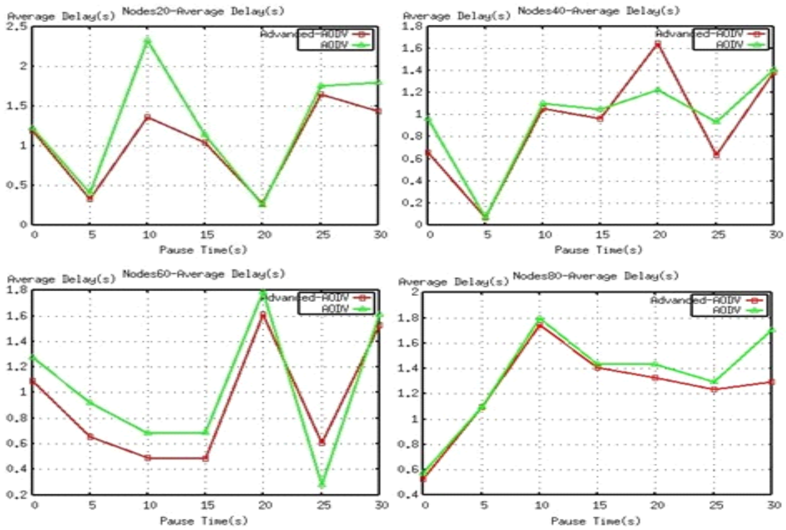

Fig. 2 The comparison of Average End-to-End Delay

Fig. 2 shows the simulation results of the average end-to-end delay of Ad-AODV and AODV in the different residence time of the different number of nodes. According to the calculation formula of the average reduction rate, in the case of the simulation of twenty nodes, the average delay of Ad-AODV is an average of 4.2% lower than AODV 's. In the case of the simulation of forty nodes, the average delay of Ad-AODV is an average of 3.2% lower than AODV 's. In the case of the simulation of sixty nodes, the average delay of Ad-AODV is an average of 4.3% lower than AODV 's. In the case of the simulation of eighty nodes, the average delay of Ad-AODV is an average of 3.7% lower than AODV 's. Ad-AODV increases the network throughput.

(3) Routing Load

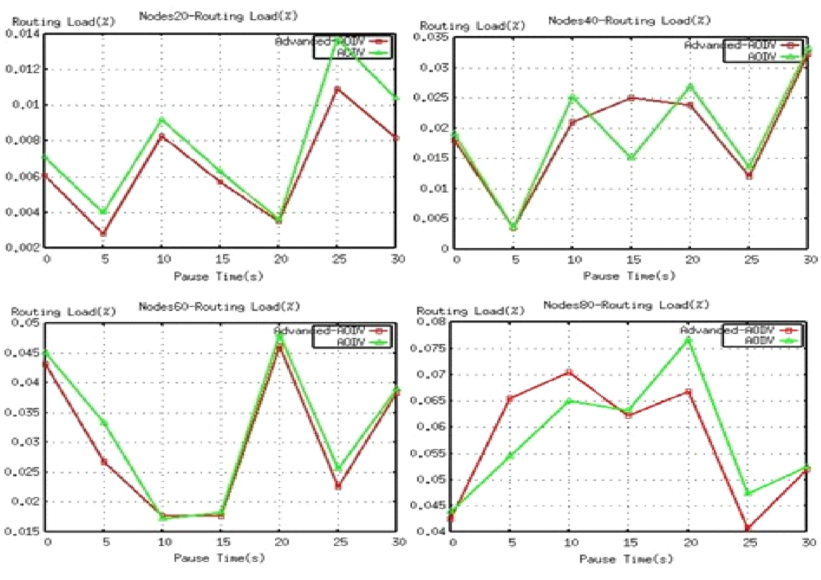

Fig. 3 The comparison of Routing Load

Fig. 3 shows the simulation results of the routing load of Ad-AODV and AODV in the different residence time of the different number of nodes. Due to the nodes of the more residual energy are selected on the route in Ad-AODV, so the route will not quickly come into force, thus it reduces the number of routing discovery. According to the calculation formula of the average reduction rate, in the case of the simulation of twenty nodes, the routing load of Ad-AODV is an average of 3.9% lower than AODV 's. In the case of the simulation of forty nodes, the routing load of Ad-AODV is an average of 2.1% lower than AODV 's. In the case of the simulation of sixty nodes, the routing load of Ad-AODV is an average of 3.3% lower than AODV 's. In the case of the simulation of eighty nodes, the routing load of Ad-AODV is an average of 2.5% lower than AODV 's. Ad-AODV saves the network resources.

Conclusion

In the paper, Ad-AODV is presented based on the original AODV. Ad-AODV considers two metrics which are the current residual energy and the load balancing of the nodes in the routing discovery process of AODV. According to these two metrics, a node may delay the received RREQ and choose an optimal route eventually. Compared with AODV, Ad-AODV not only prolongs the survival time of the network, but also improves the packet delivery ratio, lowers the average end-to-end delay and reduces the routing load.

References

[1] X.Saadawi. Revealing the Problems with 802.11 MAC Protocol in multi-Hop Wireless Networks[J]. Computer Networks. 2002; 38(4): 531-548.

[2] C Perkins, E Belding-Royer, S Das. Ad Hoc On-Demand Distance Vector(AODV) Routing. RFC 3561. 2003.

[3] C Perkins, Bhagwat P. Highly dynamic destination-sequenced distance vector routing(DSDV)for mobile computers[C]. Computer Communication Review. 1999; 24(4): 234-244.

[4] David B, Johnson, David A.Malta, Yih-Chun Hu. The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks(DSR). IEIF Internet Draft, draft_ietf_manet_dsr_09.txt.

[5] Liao Deng. Comparison of Typical Routing Protocols for Ad Hoc Mobile Wireless Networks Based on NS2 [J]. Nat .Sci . 2005.9; 9(17): 23-26.

[6] Lee S J, Gerla M. Dynamic load-aware routing in ad hoc net-works. IEEE International Conference on 2001. 10: 3206-3210.

[7] Fang Lu ping, Liu Shi hua, Chen Pan, etc. NS-2 network simulation Fundamentals and Applications. Bei Jing: National Defence Industry Press. 2008: 147-148.

[8] Floriano De Rango, Mauro Tropea. Energy Saving and Load Balancing in Wireless Ad Hoc Networks through Ant-based Routing[C]. International conference on Symposium on Performance Evaluation of Computer & Telecommunication Systems. Istanbul. 2009: 117-124.