ACBRAAM: A Content Based Routing Algorithm using Ant Agents for MANETs

Ramkumar K. R.[1], **Sakthivel K.**[2] and **Ravichandran C. S.**[3]

Submitted in July 2010; Accepted in November 2010

Abstract - ***A mobile ad hoc network (MANET) is a temporary network which is formed by a group of wireless mobile devices without the aid of any centralized infrastructure. In such environments, finding the identity of a mobile device and maintaining the paths between any two nodes are challenging tasks, in real time the limited propagation range of mobile devices restrict its identity only to its neighbors and a new host enters in to a MANET does not know the complete details of that instantaneous MANET. This paper analyses the possibility of content based route discovery and proposes a framework for request based route discovery and path maintenance using ant agents. The ant agents fetch routing information along with content relevancy which will have a major influence on pheromone value. The pheromone value is used to find the probability of goodness. The proposed framework consists of ant structures and algorithms for route discovery and path maintenance.***

Index Terms - MANETS, ANT, Request Based Path Setup

1.0 INTRODUCTION

A Lot of research work is going on the development of routing algorithms for MANETs. The swarm intelligence based routing algorithms are Antnet [5], ARA [3] and AntHocNet [4] The categorization will be in general either as *proactive* or *reactive routing.* Proactive routing algorithm updates routing tables constantly but reactive routing algorithms update routing information when required. In path maintenance phase the ants' exploratory behavior is limited around the current optimal path. The basic design behind ACO algorithms for routing is the consciousness of routing information through path sampling using ant agents. These ant agents are generated concurrently and independently by the source nodes, with the task to try out a path to an assigned destination. Assigned destination is an assumption in all existing algorithms ie) the user has to mention source and destination addresses manually. In the range limited networks there is no standard approach to identify the destination node. The general algorithms are working as forward ants always attempts to discover newer routes and the backward ants update path quality and maintain pheromone values. The pheromone value is a measure of probability of goodness going over that neighbor on the way to the destination.

In this paper the content based route discovery is proposed, the precise number of forward ant generations and implementation of heuristic routing methodology are given as algorithms. The rest of the paper is organized as follows. In Section II, the different types of forward ants and backward ants which ensure the content based route discovery and guaranteed data transaction are discussed. The timer introduced here is to reduce the number of backward ants in heavily loaded or congested path .The XML privileges are taken in to account to maintain the consistent forward ant and back ward ant functionalities. Next In Section III the proposed algorithms for forward ants, in section IV path updating algorithms are discussed and in section V the simulation and results are explained.

[1,2]*Sri Venkateswara College of Engineering, Sriperumbudur, Chennai, INDIA*
[3]*SSK College of Engineering and Technolgy, Coimbatore, INDIA*
E-Mail: [1]*ram@svce.ac.in,* [2]*sakthivel87@gmail.com and* [3]*eniyanravi@gmail.com*

2.0 ANT AND HOST PROFILE

A FORWARD ANT WITH CONTENT TAG

When a source node needs some information or content from an existing MANET, it first checks the cache for existing routes, when no routes are known, it broadcasts forward request ants with content tag and it is propagated through the network till it reaches maximum hop count. The forward ant carries the content to be searched, when a relevant content is found then forward ant is converted in to backward ant, at the same time the forward ant continues its travel for more relevant contents till it reaches maximum hop count. A forward ant at each intermediate node selects next hop using the information stored in the routing table of that node or by rebroadcast. The timer attribute is used to find out congested path for load balancing. The forward ant initializes the timer value to zero and increments the value by milliseconds till it reaches destination. When a forward ant finds the relevant content from an authenticated node, then backward ant is generated as in AntHocNet[4] which takes same path but in opposite direction. The backward ant updates pheromone value as it moves on its way to source node. The content relevancy and availability ratio decides *pheromone* value, more relevant content increases pheromone value. The definition of forward ant is given in XML format to acquire all benefits of XML in data delivery. XML could be easily combined with DTD [Document Type Definition], XML Schema for integrity checking and SOAP, XML RPC for accessing remote methods and devices, and also it could be benefited from the XML Security.

2.1 Schema definition

```
<?xml version="1.0"?>
 <xsi:schema
xmlns:xsi="http://www.w3.org/2001/XMLSchema">
<xsi:element name="fwdant">
<xsi:complexType>
 <xsi:sequence>
```

```
            <xsi:elementname="fwd" type="xsi:integer"/>
             <xsi:element name="req" type="xsi:integer"/>
             <xsi:element name="payload"type="xsi:integer"/>
            <xsi:element name="hopcount" type="xsi:integer"/>
            <xsi:element name="maxhopcount"
type="xsi:integer"/>
            <xsi:element name="timer" type="xsi:integer"/>
            <xsi:element name="srcaddr" type="xsi:integer"/>
             <xsi:element name="destaddr" type="xsi:integer"/>
             <xsi:element name="content" type="xsi:string"/>
             <xsi:element name="path">
            <xsi:complexType>
            <xsi:sequence>
            <xsi:element name="n1"
              type="xsi:integer"maxOccurs="unbounded"
            minOccurs="0"/>
 </xsi:sequence>
</xsi:path>
 </xsi:complexType>
 </xsi:element>
 </xsi:sequence>
 </xsi:attribute>

 </xsi:complexType>
 </xsi:element> </xsi:schema>
```

2.2 Forward ant structure

The forward ant structure could be combined with XML Schema and explored to all its neighbors to discover content and new routes.

```
<fwdant id = no>
<fwd>1</fwd>
 <req>1</req> <payload>0</payload>

<hopcount>CurrentHopCount</hopcount>
<maxhopcount>Theory Standards  </maxhopcount>
<timer>00:00:00:00</timer>
 <srcaddr>Address </srcaddr>
 <nexthop>Neighbor Address<nexthop>
    <destaddr>Empty</destaddr>
     <content>Content To be searched</content>
     <path>
            <n1>Neighbor1</n1>
            <n2>Neighbor2</n2>
                    ………

            <nN>NeighborN</nN>
     </path>
</fwdant>
```

B NODE PROFILE

It is been assumed that user sets profile for his device which participates in MANET. Profile states the nature of content it has and defines access rights. A forum could be constituted to define profile format. If a node wishes to contribute or distribute its data then it can tag the content's availability in various categories. First one is public content which could be accessed by any node in that network, data movement and ant movement will not be sensed by the user of that device, in other terms the public contents will be delivered without human intervention. Second content type is categorized as protected data, protected data also will be shown for public view, content tag could be identified by any node but content delivery has to be authenticated by the user of that target host. Proper certification method and security algorithms will provide a secured way of protected data transaction. Third is categorized as private content, which cannot be accessed by other hosts.

2.3. Node profile

```
<node>
<address>Nodeip</address>
<public>

<contenttag>Tagname[JAVA]
            <filename>Name of the file<filename>
 </contenttag>
 <contenttag>Tagname[C++]

             <filename>Name of the file</filename>
</contenttag>

</public>
<protected>

    <contenttag>Tagname[Photos]
    <filename>Name of the file</filename>

    </contenttag>
    </protected>
</node>
```

3.0 ALGORITHMS FOR ROUTE DISCOVERY

The basic structure is taken from ARA [3], the attributes like timer value; content tag and content relevancy are updated. The standard stack structure to hold the path information is changed as path variable since XML format is used for forward ants. The XML schema is used to check integrity. So the corrupted forward ants could be discarded.

Route discovery is the process of finding possible paths between source and destination which seizes required content. The result of route discovery process is the generation n of multiple backward ants with updated paths to relevant content holder targets.

F=1	R=1	P=0	FU	AntID
HopCount				MaxHopCount
Timer [00:00:00:00]				
Source Address				
Target Address [empty]				
Next Hop [neighbor]				
Content Tag				Content Relevancy
Path [empty]				

Figure 3.1: Forward ant Structure

Algorithm 1: Generation of forward ant with content

Input:f_{ant}: forward ant attributes, a_f:forward/backward, a_r:request/reply, a_p:payload/empty, a_{id}: request id,

a_{nhop}:neighbour host
a_{hc}:current hopcount,

a_{mhc}:maximumhopcount limit,
a_{timer}: timervalue,a_{src}:sourceaddress,
a_{dst}:destination address,
a_{ctag}:Contenttag, a_{crel}:contentrelevancy,
a_{path}:ant path n_{list}: neighbor list

Output: f_{ant}:forward ant

//Construct a forward ant with all initial parameters.
a_{id}=unique id ,
a_{ctag}= Requestedcontent
a_{crel}=0.0, a_{mhc}=maximum hopcount

a_{dst}=null,a_f=1,a_r=1,a_p=0,a_{hc}=0
a_{timer}=00:00:00:00,a_{nhop}=null a_{path}=null
f_{ant}=construct_Fi(a_f,a_r,a_p,a_{id},a_{hc},a_{mhc},

atimer,asrc,anhop,adst,actag,acrel,apath)
forward(f_{ant},n_{list});

end

Algorithm 2: Routediscovery (f_{ant},n_{list})

Input :f_{ant} :forward Ant, n_{list} :neighbor list
Output: Route discovery and table updating
//Constitute a route discovery process from Source to destination
if a_{hc}=0 || a_{id} is new then
if a_{hc}<=a_{mhc} then
for a_{nhop} Є n_{list} do
if *actag* Є *nctaglist* **&&** n_{ctag}=*public* || *protected* then
f_{ant}=updateandgenerate(a_{hc},a_{nhop},a_{path})

Converttobackwardant(f_{ant})
end
else

Routediscovery(f_{ant},n_{list})

end
else
discard(f_{ant})
end

Generation of backward ant is a simple process of changing the addresses and setting proper timer values.Backward ant structure given below changes some flag bits and content relevancy is filled with the help of ranking algorithms.

F=0	R=0	P=0	FU	AntID
HopCount				MaxHopCount
Timer [00:00:XX:XX]				
Source Address				
Target Address				
Next Hop				
Content Tag				Content Relevancy
Path				

Figure 4.2: Backward ant structure with content relevancy

Algorithm 3: Generation of backward ant

Input : $f_{an\ t}$: forward ant
Output: b_{ant}:backward ant
copy b_{ant}=f_{ant}

swap b_{ant}.a_{src} with b_{ant}.a_{dst}
bant.atimer=fant.atimer * α [α = 2]
bant.amhc=fant.ahc
unicast(b_{ant})

4.0 PATH UPDATION

4.1 Backward Ant from Destination

The backward ants are used to update the discovered path. Existing algorithms states that backward ants calculates pheromone values using queuing delay and MAC delay as in AntHocNet[4] in turn pheromone values are used to calculate the probability of goodness. The content relevancy is also included to calculate pheromone value. New pheromone value will be calculated as

$\Gamma nd = \alpha * \Gamma nd + (1-\alpha) * \Gamma nd * Cr$ ------------------ (1).

Γ_{nd} – New Pheromone value for the neighbor.
α – 0.7 taken from standards[good probability]
Cr- Content Relevancy.

Content Relevancy ranges from [0,1] which will have direct impact to pheromone value. This pheromone value is updated to the node table of every node by backward ants.
The backward ant travels back to source by following path information and updates pheromone value which also includes content relevancy ratio, content relevancy will have a major impact to retrieve relevant content. The timer value will be trailed automatically to discard longer waiting backward ants. If a backward ant is unable to reach the source on time then it clearly indicates that the path is more congested and data delivery will not be fruitful. So better the backward ant could be discarded in the intermediate node itself so that path could be avoided for data delivery.

Algorithm 5: unicast(b_{ant})

Input : b_{ant}:backward ant

Output: *m*:updated path
// find content relevancy and assign it to Cr.
$b_{ant}.a_f$=0,$b_{ant}.a_r$=0,$b._{ant}.a_p$=0 *bant.acrel=Cr*
if $b_{ant}.a_{dst}$==*currentNodeIP* then

$\Gamma_{nd} = \alpha*\Gamma_{nd} + (1-\alpha)* \Gamma_{nd} *C_r$
start new unicast request from $b_{ant}.a_{src}$ to *bant.adst*
end
else if $b_{ant}.a_{dst}$!=*currentNodeIP* then
if $b_{ant}.a_{hc}$<$b_{ant}.a_{mhc}$ && $b_{ant}.A_{timer}$>*00:00:00:00* then

pickup next node from $b_{ant}.a_{path}$ and

unicast(b_{ant})
$b_{ant}.a_{hc}$=$b_{ant}.a_{hc}$-*1*
autodecrement $b_{ant}.a_{timer}$
// update pheromone value based on content relevancy
else
discard(b_{ant})
end
end

The free function will free up memory space which is allocated to a particular ant. A node can delete a forward ant which crossed maximum hop count and time exhausted backward ants. The discard algorithm is used to control flooded forward and backward ants where ever is possible to reduce the congestion and collision.

Algorithm 4: discard (f_{ant} // b_{ant})

Input : f_{ant}: forward ant or b_{ant}: backward ant

Output : null
free(f_{ant});

4.2 Working Ants

Once the content discovery and path establishment is over then the data transaction thread has to be started. The data transaction thread follows a sequence, first sending a unicast request from source to destination to confirm data delivery, the destination node starts data delivery after receiving confirmation, the payload is accompanied with a backward ant to update path while transferring payload. In case of route failure during payload delivery the backward ant will be detached and converted as forward ant to discover new routes as in HPRAAM [1].

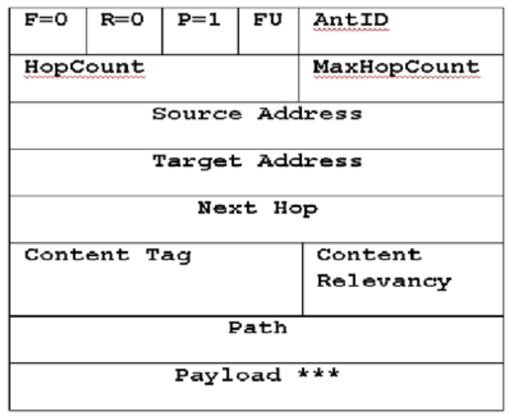

Figure 4.3: Backward ant with payload

5.0 SIMULATION AND RESULTS

The simulation is tried with 60 nodes moving at random way point model with different speed and different pause time. Contents and content relevancies are distributed randomly among all nodes. Simulation is executed for random times ranges from 20 to 200 seconds and requests are made for different contents from different nodes. The proposed algorithm is compared with ARA [3] and it outperforms in searching relevant content in a short period.

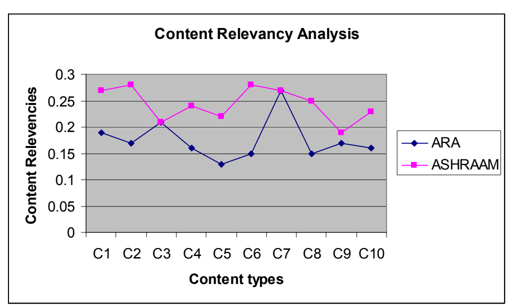

Figure 5.1: Content relevancy analysis

ASHRAAM gathers more relevant results while searching content in MANET. ARA only chooses the minimum hop count destination to retrieve contents which may be irrelevant. So it has to perform route discovery process again and again which will create more congestion and flooded ants in MANET. ASHRAAM initiates only one route discovery process; to get all destinations and content relevancies for the content.

6.0 CONCLUSION

In this study a new proposal for content based routing using ant agents has been made as a framework. The proposed framework can reduce the congestion in MANET, and also it can diminish the number of retransmissions of forward and backward ants. The timer concept decides the life time of a backward ant, a backward ant which is trapped by a heavily congested network will be dropped automatically after a period of time. The long waiting backward ants brings out unreliable paths to the source node, with the help of timer

concept the problem is completely avoided. This framework will create a new path towards the content based route discovery and XML based ant generations to leverage the benefits of both.

REFERENCES

[1]. Ramkumar. K. R, GaneshKumar. M, Hemachandar. N, ManojPrasadh. D"HPRAAM: Hybrid Parallel Routing Algorithm Using Ant Agents for MANETS" IJET ,ISSN:1793-8244,1793-8236,March 2009.

[2]. Ramkumar. K. R, GaneshKumar. M, Hemachandar. N, ManojPrasadh. D,Nisha. M"ARRAAM:A Reliable Routing Algorithm Using Ant Agents For MANET", ISSN:1865-0929,1865-0937,ISBN: 978-3-642-02341-5 ,March 2009

[3]. M. Guine,, U. Sorges, and I. Bouazzi, "ARA-the ant-colony based routing algorithm for MANETs," in Proc. of IWAHN 2002, pp. 79-85, August 2002.

[4]. G. Di Caro, F. Ducatelle, and L. M. Gambardella, "AntHocNet: An Adaptive Nature-Inspired Algorithm for Routing in Mobile Ad Hoc Networks," Tech. Rep. No. IDSIA-27-04-2004, IDSIA/USI-SUPSI, Sep.2004.

[5]. Gianni Di Caro, Marco Dorigo, "AntNet: Distributed Stigmergetic Control for Communications Networks" Journal of Artificial Intelligence Research 9 317-365. Aug'1998

Continued from page no. 275

[17]. D. B. Johnson et al., "The dynamic source routing protocol for mobile ad hoc networks (DSR)", Internet Draft, MANET working group, Feb 2002.

[18]. L. Lamport, "Password Authentication with Insecure Communication", Comm. of ACM, 24 (11), pp. 770-772, Nov. 1981

[19]. A.Kush, C.Hwang, "Proposed Protocol For Hash-Secured Routing in Ad hoc Networks", MASAUM JOURNAL OF COMPUTING (MJC) Volume: 1 Issue: 2 Month: September 2009 , pp 221-226

[20]. A.Kush, C.Hwang, P.Gupta, "Secured Routing Scheme for Adhoc Networks" International Journal of Computer Theory and Engineering (IJCTE). May 2009, Volume 3. pp 1793-1799.

[21]. NS Notes and Documentation, available at www.isi.edu/vint

[22]. Hongmei Deng, Dharma P. Agrawal TIDS: threshold and identity-based security scheme for wireless ad hoc networks Elsevier journal of adhoc networks, Vol 2 (2004), pp 291-307