ANALYSIS OF BABEL AND PUMA PROTOCOL FOR MOBILE ADHOC NETWORKS (MANETS)

Anand Nayyar*

Abstract:

An adhoc network is a collection of wireless mobile nodes that forms a temporary network without any centralized administration. A short lived network just for the communication needs of the moment. It is an infrastructureless network. While early research efforts assumed a friendly and cooperative environment and focused only on problems such as wireless channel access and multihop routing. The main aim of this Research Paper is to present an overview of BABEL and PUMA protocols along with their respective advantages and disadvantages. Design of a suitable routing protocol is difficult for mobile ad hoc networks due to its inherent dynamism and frequent topology change. Multicasting is even more complex because it requires transmission of an information to various destinations at approximately same time, if possible. Active research work in this field has resulted in a variety of proposals based on tree or mesh structures. This Research Paper presents a state-of-the-art overview of multicast routing protocols Babel and puma for ad-hoc networks.

Keyword: Adhoc Networks, BABEL Protocol, PUMA Protocol, DIDD Backoff, Multicast.

* Assistant Professor, Department of Computer Applications & IT, KCL Institute of Management and Technology, Jalandhar.

I. Introduction

Mobile Adhoc Network (MANET) refers to a form of infrastructure less network connecting mobile devices with wireless communication capability [11] [12]. Each node behaves as a router as well as an end host, so that the connection between any two nodes is a multi-hop path supported by other nodes. MANET represents a system of wireless mobile nodes that can freely and dynamically self-organize in to arbitrary and temporary network topologies, allowing people and devices to allowing people and devices to communicate without any preexisting communication architecture [8]. Each node in the network also acts as a router, forwarding data packets for other nodes. They communicate directly with devices inside their radio range in a peer-to-peer nature. If they wish to communicate with a device outside their range, they can use an intermediate device or devices within their radio range to relay or forward communications to the device outside their range. An ad hoc network is self-organizing and adaptive .Generally, the main function of routing in a network is to detect and maintain the optimal route to send data packets between source and destination via intermediate nodes. A MANET is a peer-to-peer network that allows direct communication between any two nodes, when adequate radio propagation conditions exist between these two nodes [10]. If there is no direct link between the source and the destination nodes, multi-hop routing is used. In multi-hop routing, a packet is forwarded from one node to another, until it reaches the destination.

A. Characteristics of Mobile Ad-Hoc Networks

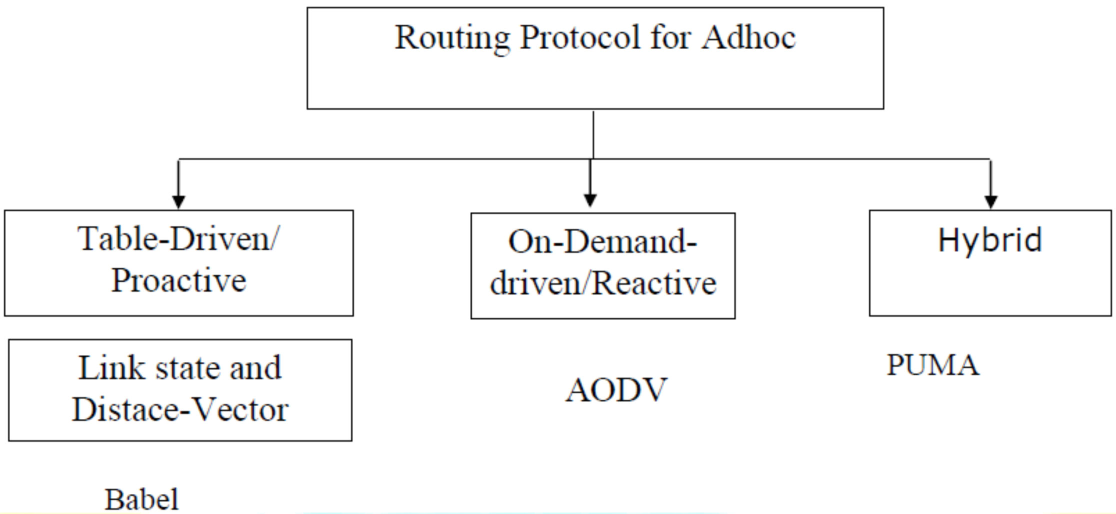
- **Autonomous and infrastructure less**-MANET is self-organized and independent of any established infrastructure and centralized network administration.
- **Multihop routing-** As there is no dedicated router, every node functions as a router and aids in forwarding each others' packets to intended destination.
- **Dynamic network topology-** Since MANET nodes move randomly in the network, the topology of MANET changes frequently, leading to regular route changes, network partitions, and possibly packet losses.

- **Variation on link and node capabilities-** Each participating node may be equipped with different type of radio devices that have varying transmission and receiving capabilities, and possibly operate on multiple frequency bands Asymmetric links might be resulted due to this heterogeneity in the radio capabilities.
- **Energy-constrained operation-** The processing power of node is restricted because the batteries carried by portable mobile devices have limited power supply. As a result, the services and applications that can be supported by each node are limited.
- **Network scalability-** Many MANET applications may involve large networks with tens of thousands of nodes especially that can be found in tactical networks. Scalability is very crucial for the successful deployment of MANET [5].

B. Applications of Mobile Ad-Hoc Networks

- **Military battlefield-**The modern digital battlefield demands robust and reliable communication in many forms. Most communication devices are installed in mobile vehicles, tank, trucks etc. Also soldiers could carry telecomm devices that could talk to a wireless base stationer directly to other telecom devices if they are within the radio range.
- **Sensor Networks-** This technology is a network composed of a very large number of small sensors. These can be used to detect any number of properties of an area. Examples include temperature, pressure, toxins, pollutions, etc.
- **Automotive Applications-**Automotive networks are widely discussed currently. Cars should be enabled to talk to the road, to traffic lights, and to each other, forming ad-hoc networks of various sizes.
- **Commercial sector-**Ad hoc can be used in emergency/rescue operations for disaster relief efforts, e.g. in fire, flood, or earthquake.

C. Classification of Routing Protocols



II. Babel — A Loop-Free Distance-Vector Routing Protocol

BABEL is a proactive routing protocol based on the distance-vector algorithm. It is a distance-vector routing protocol for IPv6 and IPv4 with fast convergence properties. It is based on the ideas of DSDV, AODV but uses a variant of link cost estimation rather than a simple hop-count metric. Babel was designed to be robust .It is a proactive protocol, but with adaptative (reactive) features. It senses link quality for computing route metrics using a variant of the link cost estimation algorithm. It uses a feasibility condition that guarantees the absence of loops .It uses sequence numbers to make old routes feasible again .It is efficient on both wireless mesh networks and classical wired networks. It speeds up convergence by reactively requesting a new sequence number. Babel was designed to be robust and efficient on both wireless mesh networks and classical wired networks [12].

BABEL has two distinctive characteristics that optimize relay performance. First, it uses history-sensitive route selection to minimize the impact of route flaps - the situation where a node continuously changes its preferred route between source and destination pair and leads to route instability. Thus, when there is more than one route of similar link quality, the route selection favors the previously established path rather than alternating between two routes. Second, BABEL executes a reactive update and forces a request for routing information when it detects a link failure from one of its preferred neighbors. Given the link quality measurements were previously completed at initialization stage, BABEL claims to have almost immediate route convergence time when triggering an explicit update.

Babel uses history-sensitive route selection to avoid route flapping, the situation in which routers repeatedly switch between two routes of similar quality. Babel enjoys fairly fast convergence. Since Babel uses triggered updates and explicit requests for routing information, it usually converges almost immediately after the link quality measure has completed. The initial solution is not optimal — after converging to a satisfactory set of routes, Babel will take its sweet time before optimizing the routing tables. In the presence of heavy packet loss, converging on an optimal set of routes may take up to a minute or so (with the default update interval of 30 seconds).

Babel has provisions for link quality estimation and for fairly arbitrary metrics. When configured suitably, Babel can implement shortest-path routing, or it may use a metric based e.g. on packet loss statistics. Babel nodes will successfully establish an association even when they are configured with different parameters. For example, a mobile node that is low on battery may choose to use larger time constants (hello and update intervals, etc.) than a node that has access to wall power. Conversely, a node that detects high levels of mobility may choose to use smaller time constants. The ability to build such heterogeneous networks makes Babel particularly adapted to the wireless environment.

Finally, Babel is a hybrid routing protocol, in the sense that it can carry routes for multiple network-layer protocols (IPv4 and IPv6) whichever protocol the Babel packets are themselves being carried over [3].

Babel on wired networks

Babel will also work efficiently on wired networks. When the Babel daemon detects a wired network, it will use a larger interval between hellos, disable link quality estimation, and perform split-horizon processing. In the absence of mobility (on a stable network with no link failures), Babel over a wired network will generate roughly between 3/4 and 1.5 times the amount of traffic that RIP would generate, depending on the exact network topology. However, since Babel uses explicit Hello messages and never counts to infinity, its update interval can be set to much larger values.

Babel on embedded systems

Being a distance vector protocol, Babel has extremely modest memory and CPU requirements. I have never seen the Babel daemon appear on either a CPU or a memory monitor.

Babel on dual-stack networks

Unlike most routing protocols, which route either IPv4 or IPv6 but not both at the same time, Babel is a *hybrid* IPv6 and IPv4 protocol: a single update packet can carry both IPv6 and IPv4 routes (this is similar to how multi-protocol BGP works). This makes Babel particularly efficient on dual (IPv6 and IPv4) networks.

Advantages:

- Babel was designed to be robust and efficient on both wireless mesh networks and classical wired networks.
- It is a distance-vector protocol.
- It is a proactive protocol, but with adaptative (reactive) features;
- It senses link quality for computing route metrics using a variant of the ETX algorithm.
- It uses a feasibility condition that guarantees the absence of loops (the feasibility condition is taken from EIGRP and is somewhat less strict than the one in AODV);

- It uses sequence numbers to make old routes feasible again (like DSDV and AODV, but unlike EIGRP);
- It speeds up convergence by reactively requesting a new sequence number (like AODV, and to a certain extent EIGRP, but unlike DSDV);
- It allows redistributed external routes to be injected into the routing domain at multiple points (like EIGRP, but unlike DSDV and AODV).

Limitations:

Babel has two limitations that make it unsuitable for use in some environments.

- First, Babel relies on periodic routing table updates rather than using a reliable transport; hence, in large, stable networks it generates more traffic than protocols that only send updates when the network topology changes. In such networks, protocols such as OSPF or EIGRP might be more suitable.
- Second, Babel does impose a hold time when a prefix is retracted While this hold time does not apply to the exact prefix being retracted, and hence does not prevent fast re-convergence should it become available again, it does apply to any shorter prefix that covers it. Hence, if a previously disaggregated prefix becomes aggregated, it will be unreachable for a few minutes. This makes Babel unsuitable for use in mobile networks that implement automatic prefix aggregation.

III. PUMA –Priority unavoidable multiple access

PUMA establishes and maintains a shared mesh for each multicast group without depending upon a unicast routing protocol. The new DIDD backoff mechanism is especially designed for a network consisting of a large number of stations and working under heavy loads. It reduces the number of collisions, so brings the growth of the network efficiency. The idea of CTS over RTS packet domination in radio channel completely solves the hidden station problems.

In PUMA, any source can send multicast data to a multicast group without having to knowing the constituent members of the group that is shown in figure 1.1.

Moreover source does not require joining the group to dispatch the data. PUMA is a receiver initiative approach where receivers join the multicast group using the address of a special core node without the need for flooding of control packets from the source of the group. When a receiver wishes to join a multicast group, it first determines whether it has received a multicast announcement for that group before. If the node knows the core, it starts transmitting multicast announcements and specifies the same core for the group. Node propagates multicast announcements based on the best multicast announcements it receives from its neighbors.

Hence, multicast data packets move hop by hop, until they reach mesh members. The packets are flooded within the mesh, and group members use a packet ID cache to detect and discard packet.

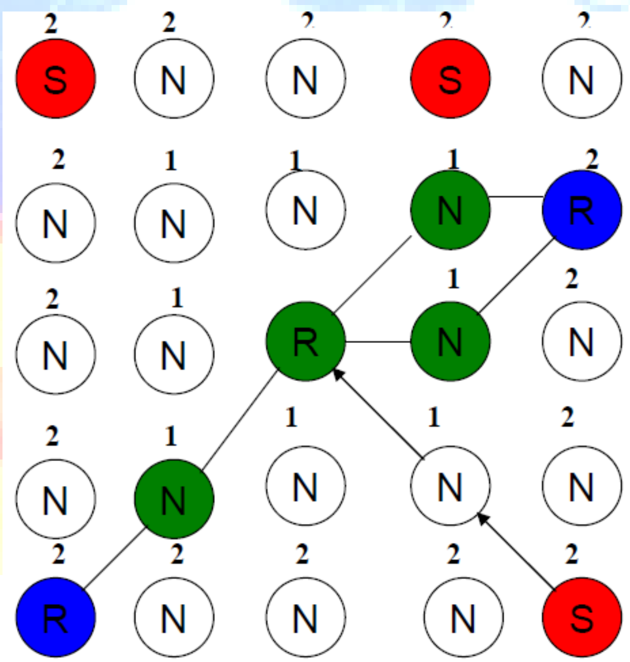


Fig 1.1 PUMA Protocol

- Mesh Based Routing Protocol
- Assume S1, S2, S3 are senders
- Assume R1, R2, R3 are receivers and R3 is elected as core
- Cores in PUMA transmit a ***multicast announcement***
- Multicast announcements have an additional field called ***mesh member flag***
- Receivers set mesh member flag to TRUE by default
- If a node has a neighbor which has member flag set and distance to core less than its own, then it considers itself a mesh member and sets mesh member flag to TRUE in its own multicast announcement.
- PUMA does not need a separate join announcement.
- PUMA includes ***all*** shortest paths between each receiver and the core

Data packets are sent towards the next-hops and flooded within the mesh as soon as they reach the first mesh member.

Isochronous mode of transmission

Time-bounded services can be implemented in PUMA. Since PUMA offers packet transmission with priority, it allows for isochronous service provision. It is possible with the aid of the special JAM signal that is transmitted in specified moments by all stations wishing to realize isochronous services. Three different time intervals: SIFS, PIFS, DIFS (DIFS>PIFS>SIFS) are defined in PUMA protocol. Every station measures these time intervals after the end of data transmission to determine the moment they can start its own transmission. The station proceeds with its isochronous transmission if the medium is determined to be idle for an interval that exceeds the PIFS time. All stations sending isochronous packets should start its transmission in the same time and send the JAM signal. The JAM signal consists of pulses of energy and has the length of one slot. This signal informs all other stations (especially stations sending asynchronous packets) that for a moment in their neighborhood, an isochronous transmission begins. It means that all other stations have to defer its transmission until reception of RTS or CTS packet to update their net allocation vector (NAV). The operation of JAM signal is similar to the busy tone in other multi-access protocols. A random interval (backoff interval) is then selected and is used to initialize the backoff timer. The backoff timer is decremented only when the medium is idle. It is frozen when the medium is busy until the next PIFS period.

A station initiates a RTS packet transmission when the backoff timer reaches zero. On reception of a RTS packet the receiver responds with a CTS packet, which can be transmitted after the channel has been idle for a time interval exceeding SIFS. After the successful exchange of RTS and CTS packets the transmitter sends in collision free manner, the data packet after SIFS. In the case when a CTS packet is not received within the predetermined time interval, the RTS is retransmitted following the backoff rules. The

PUMA protocol uses NAV – a timer that is always decreasing if its value is non-zero. A station is not allowed to initiate a transmission if its NAV is non-zero. The use of NAV to determine the busy/idle status of the channel is referred to as the Virtual Carrier sense mechanism.

It can be noted from the earlier study that transmission of small data packets is unprofitable because of large overhead . Moreover, for each data transmission all contention procedures have to be run. The packet train mechanism has been implemented in PUMA to increase the protocol performance measures. Data packets are transmitted in sequence without collision after the successful medium reservation by a RTS/CTS packets exchange. The number of data packets transmitted in sequence can be set to the specific value.

The idea of isochronous transmission assumes that packets generated by a traffic source should be delivered do its destination in the specified time. To accomplish this, the lifetime of each isochronous packet is measured. If it reaches its limit and the packet cannot be sent to its destination it is treated as useless and removed from the station buffer.

Asynchronous mode of operation

This mode of operation is very similar to that of IEEE 802.11 DCF. The station proceeds with its asynchronous transmission if the medium is determined to be idle for an interval that exceeds the distributed interframe space (DIFS). In the case when the medium is busy the transmission is deferred until the end of ongoing transmission. A random interval (backoff interval) is then selected and the backoff timer is run. The backoff interval can be changed from the lower limit determined by the parameter CWmin to upper limit CWmax. In order to reduce the probability of collision, after each unsuccessful transmission attempt the expected value of the random backoff interval is increased exponentially up to the predetermined maximum. The backoff timer is only decremented when the medium is idle after DIFS time. It is frozen when the medium is busy. A station initiates a transmission of RTS packet when the backoff timer reaches zero.

In the case, when a CTS packet is not received within the SIFS time, it means that collision happened and the RTS is retransmitted following the backoff rules. The successful reception of CTS packet guarantees collision-free DATA packet transmission. Each station should update their NAV vector after receiving RTC or CTS. The packet train mechanism has been implemented in asynchronous transmission in the same manner like in the case of isochronous transmission. The data with additional CTS packets separated by SIFS intervals are transmitted in sequence without collision after the successful medium reservation by a RTS/CTS packets exchange. The only difference is that ACK packet finishes each transmission cycle.

Isochronous-asynchronous traffic scaling

The isochronous packets have a higher priority than asynchronous ones. It can happen that a large number of stations start to realize time-bounded services and then they kill asynchronous traffic (telnet, www, ftp, etc.). Such a situation is very undesirable. So, it is necessary to introduce a mechanism, which permits to control the minimal amount of asynchronous traffic (in IEEE 802.11 this is guaranteed by the super frame but it can only be used in infrastructure networks).

The idea is to use an additional timer, which is used to measure the life-time of asynchronous packets located in source station buffer. An asynchronous packet located in the head of queue in the buffer gets a higher priority if its lifetime is reached (its priority is equal to the priority of isochronous packets). This packet will be certainty sent in the near future. All other asynchronous packets located in the buffer have the same low priority and after reception of ACK the life-time is measured using timer T2 for the next asynchronous packet (always located on the top of the buffer). PUMA permits to regulate the minimum level of realized asynchronous traffic for every station. This level, of course, depends on the number of contending stations sending isochronous traffic. The additional rule has been introduced in PUMA protocol to preserve an asynchronous station sending packets with packet-train mechanism enabled after T2 timeout. The asynchronous station can send then only one asynchronous data packet. This rule is necessary to assure the proper QoS level for isochronous transmission.

The name of the protocol – Priority Unavoidable Multiple Access (PUMA) came into being from the idea of timer T2 usage. The station sending asynchronous packets independently of the isochronous traffic load, will get after certain time (determined by T2 timeout) the possibility of contention with stations sending isochronous packets. Timeout T2 can be dependent on the number of contending stations. It allows make the isochronous throughput independent from a number of stations sending asynchronous packets.

Advantages:

- It allows sending specified packets with priorities, which makes possible provision of isochronous services.
- It completely solves the hidden station problem through CTS over RTS packets domination in radio channel.
- It guarantees collision free data packet transmission.
- Some enhancements were introduced to increase the efficiency of operation, by adding a new backoff scheme called DIDD. The packet-train mechanism was used to improve the protocol efficiency while sending very short data packets.

- It provides JAM signaling for realization of time-bounded services.
- It has additional T2 timer for isochronous-asynchronous traffic scaling.
- It has optional packet-train transmission mechanism.
- PUMA can easy be implemented in wireless network cards.
- PUMA protocol is fair, efficient, stable and allows for provision of time bounded services.

Disadvantages:

The only problem is to determine the number of stations in the neighborhood of each station. However, the solution seems to be very simple since each station is obliged to listen the RTS/CTS packets and these packets carry information about the source and destination addresses, so it can create dynamically updated list of known addresses. A station can estimate the number of stations in its neighborhood on the basis of registered addresses. Obviously, if a given address does not appear in RTS/CTS packets for a long time it should be removed from the list. The described method permits to assure the desirable level of isochronous throughput. The growth of number of stations sending asynchronous packets should not influence the isochronous throughput.

Conclusion

Multicast routing is an essential component of communication protocols in mobile ad hoc networks. The design of the protocols are driven by specific goals and requirements based on respective assumptions about the network properties or application area. The Research Paper tries to review typical tree-based and mesh-based multicast routing protocols and reveal the characteristics and trade-offs. Generally the tree based protocols are efficient than mesh based ones from the perspective of energy efficiency generated by the minimization of transmission redundancy, whereas mesh based protocols provide better reliability at the cost of redundancy. The subject of our future investigations will be test the related packet delivery ratio, packet overhead, link throughput and average latency of PUMA and BABEL protocol.

REFERENCES

1 Ahmed Al-Maashri and Mohamed Ould-Khaoua. "Performance Analysis of MANET Routing Protocols in the Presence of Self-Similar Traffic". IEEE, ISSN- 0742-1303, First published in Proc. of the 31st IEEE Conference on Local Computer Networks, 2006.

2 A review of current routing protocols for ad-hoc mobile wireless networks, E.Royer, C.K. ToY, 2000.

3 AODV Routing Protocol Implementation Design Ian D. Chakeres, Elizabeth M. Belding-Royer, 2000.

4 An Experimental Comparison of Routing Protocols in Multi Hop Ad Hoc Networks David Murray, Michael Dixon, Terry Koziniec,2009.

5 Anuradha Banerjee et. al. / International Journal of Engineering Science and Technology Vol. 2(10), 2010, 5594-5604.

6 Survey of Multicast Routing Protocols For Mobile Ad Hoc Networks An Interest- Driven Approach to Integrated Unicast and Multicast Routing in MANETs Rolando Menchaca-Mendez 2008 IEEE.

7 Classification and Review of Security Schemes in Mobile Computing, Vol.2 No.6, 2010.

8 Classification of Ad Hoc Routing Protocols Petteri Kuosmanen Finnish Defence Forces, 1999.

9 Clausen, T., Dearlove, C., Dean, J., and C. Adjih, "Generalized Ad Hoc Network Packet/Message Format," RFC 5444, 2009.

10 Charles E. Perkins, Elizabeth M. Belding-Royer, and Ian Chakeres. "Ad Hoc On Demand Distance Vector (AODV) Routing." *IETF Internet draft*, draft-perkinsmanet- aodvbis-00.txt, Oct 2003 C.W. Wu, Y.C. Tay, AMRIS: a multicast protocol for ad hoc wireless networks.

11 Proceedings IEEE MILCOM'99, Atlantic City, Nov. 1999.

12 DPUMA: A Highly Efficient Multicast Routing Protocol for Mobile Ad Hoc NetworksRolando Menchaca-Mendez, Ravindra Vaishampayan, J. J. Garcia-Luna- Aceves and Katia Obraczka 2005, Volume 3738/2005.

13 D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris, “A high throughput path metric for multi-hop wireless routing,” in *MobiCom’03:* Proceedings of the 9th annual international conference on Mobile computing and networking. New York, NY, USA: ACM, 2003, pp.134–146.

14 J. Chroboczek, “Babel a loop-free distance-vector routing protocol.” [Online]. Available: http://www.pps.jussieu.fr/□jch/software/babel/

15 R. Talpade, A. McAuley, and E. Bommaiah, AMRoute: Ad-hoc multicast routing protocol. Technical Report, CSHCN T. R. 99-1, University of Maryland.