Energy Efficient Multi Hop Wireless Network for Ships using Two State Routing Protocol

Selvaganesan.M, Manikandan.R

Department of Computer Science and Engineering (PG), Regional Centre, Anna University, Madurai - 625 002.

Abstract

Problem Statement: The expensive satellite communication currently accessible from ships is ill equipped to meet the needs of a growing number of seafaring Internet users. In order to provide cheap and high-speed Internet access to ships, the radio coverage of existing broadband networks can be extended through a multi-hop network that provides wireless links between neighboring ships. One of the most important issues in such networks is the appropriate choice of a routing protocol that provides efficient and reliable communication. In this paper, a maritime two-state routing protocol for a multi-hop ship network is proposed that provides efficient and reliable communication with a minimum of overhead. The maritime path loss model considered for simulations and the mobility model used in this paper represent real traffic of ships. In this paper, the proposed routing protocol is compared to the leading alternatives and simulation results are presented to quantify the performance.
Results: The proposed work is implemented in NS2 and the performance metrics like throughput, packet delivery ratio; delay and bandwidth are measured and compared with existing protocol. Conclusion/Recommendations: This System shows that our gateway selection to improve the quality of service, network throughput and packet delivery ratio with low energy power consumption per node.

Keywords

VANET, QoS, gateway selection, GMA

I. Introduction

The recent adoption of the various 802.11 wireless standards has caused a dramatic increase in the number of wireless data networks. Today, wireless LANs are highly deployed and the cost for wireless equipment is continuing to drop in price. Currently, an 802.11 adapters or access point (AP) can be purchased for next to nothing. Mobile ad hoc networks (MANET) are one area that has recently received considerable attention. One promising application of mobile ad hoc networks is the development of VEHICULAR AD HOC NETWORKS (VANET).
A MANET is a self forming network, which can function without the need of any centralized control. Each node in an ad hoc network acts as both a data terminal and a router. The nodes in the network then use the wireless medium to communicate with other nodes in their radio range.A VANET is effectively a subset of MANETs. The benefit of using ad hoc networks is it is possible to deploy these networks in areas where it isn't feasible to install the needed infrastructure. It would be expensive and unrealistic to install 802.11 access points to cover all of the roads in the United States. Another benefit of ad hoc networks is they can be quickly deployed with no administrator involvement. The administration of a large scale vehicular network would be a difficult task. These reasons contribute to the ad hoc networks being applied to vehicular environments. Traffic fatalities are one of the leading causes of death in the United States. The Federal Communications Commission (FCC), realizing the problem of traffic fatalities in the US dedicated 75 MHz of the frequency spectrum in the range 5.850 to 5.925 GHz to be used for Vehicle-to-Vehicle and Vehicle-to-Roadside communication.
The creation of Vehicular Ad Hoc Networks (VANET) has also spawn much interest in the rest of the world, in German there is the Fleet Net project and in Japan the ITS project. Vehicular ad hoc networks are also known under a number of different terms such as Inter-Vehicle Communication (IVC), Dedicated Short Range Communication (DSRC) or WAVE. The goal of most of these projects is to create new network algorithms or modify the existing for use in a vehicular environment.

Media Access Control

To create Wide Scale vehicular ad hoc networks, changes need to be made to the media access control (MAC) layer. The objective of media access control layer is to arbitrate the access to the shared medium, which in this case is the wireless channel. If no method is used to coordinate the transmission of data, than a large number of collisions would occur and the data sent would be lost. The ideal scenario is a MAC that prevents nodes within transmission range of each other from transmitting at the same time and no collision occur.
The 802.11 family of protocols use CSMA/CD with acknowledgments to restrict the number of collisions and to reliably transmit packets. The 802.11 standard defines two MAC protocols the Distributed Coordination Function (DCF) and the Point Coordination Function (PCF). The Distributed Coordination Function is a contention based access protocol. In a contention based protocol all nodes that have data to send contend for the

Manuscript received March 5, 2015
Manuscript revised March 20, 2015

channel. Contention based protocols are the easiest to implement but the problem with them is they offer no quality of service (QoS) guarantees. Contention free protocols are achieved by scheduling when a node can transmit. Contention free protocols enable the use of real-time services

Broadcast Messages

A number of challenges exist in providing reliable broadcasts. In vehicular ad hoc networks a majority of the messages that are transmitted will be periodic broadcast messages that announce the state of a vehicle to it neighbors. It is likely that there will be more broadcast messages than unicast messages in vehicular networks.

Broadcast messages cannot use the RTS/CTS exchange, because it would flood the network with traffic. As a result of not using the RTS/CTS exchange, the network exhibits the hidden terminal problem as discussed above. Also, it isn't practical to receive acknowledgments from all of the nodes that receive a broadcast message. Without receiving an ACK the sender of the broadcast has no way of determining if the broadcast was successfully received by its neighbors.

Network Simulator

NS (version 2) is an object-oriented, discrete event driven network simulator developed at UC Berkley written in C++ and OTcl. NS is primarily useful for simulating local and wide area networks. Although NS is fairly easy to use once you get to know the simulator, it is quite difficult for a first time user, because there are few user-friendly manuals. Even though there is a lot of documentation written by the developers which has in depth explanation of the simulator, it is written with the depth of a skilled NS user.

The purpose of this project is to give a new user some basic idea of how the simulator works, how to setup simulation networks, where to look for further information about network components in simulator codes, how to create new network components, etc., mainly by giving simple examples and brief explanations based on our experiences. Although all the usage of the simulator or possible network simulation setups may not be covered in this project, the project should help a new user to get started quickly.

Ii. Related Work

In [1], QoS problem caused by routing is very important. In this paper, we propose a method based on AOMDV protocol providing a route recovery mechanism when a link breaks in an active route to reduce lost packets. The results show that the proposed method can reduce packet loss ratio and delay time compared with the AOMDV. In an Ad-hoc network using AOMDV routing protocol is shown. Source node S is transmitting data to destination node D, node F detected that link F-J is broken. Node F will start a processing "local repair" [4] to discover new route to destination node by generating RREQ packet with that destination if it has not other route to D. If node F cannot receive any RREP packet, node F will transmit a RRER packet for this destination. Therefore, data packets are cached at node F and if source node S must restart route discovery these data will be deleted. On the other hand, S does new route discovery, it will increase data transferring delay time.

In the last years many routing protocols proposals have been made considering the particular VANET characteristics. From the many proposals that came up, the protocols based on the vehicles positions were found to be the most adequate to VANETs due to their resilience to handling the nodes position variation. In this study we will survey the existing position-based routing protocols. Unlike other studies we will emphasize on their applicability to different environments. We start by characterizing the vehicular network environment, namely the urban and the highway environments. Afterwards, topology-based protocols are compared to position-based protocols and to the latter are identified the different used strategies and their performances are qualitatively evaluated relatively to different metrics. The different position-based routing proposals are described including a pseudo-code specification, and a comparison is made based on different perspectives. To conclude, the main constrains to urban and highway environments are characterized and the adaptability of each protocol to each of the environments is evaluated. Equipment manufacturers have recognized the opportunity of enhancing the surface transportation by using the communication capabilities of the Vehicular Ad-hoc Networks (VANET) to offer an Intelligent Transportation System (ITS) to the drivers. The major goal of this system is to improve the driver’s safety by informing them about dangers and situations that they cannot see. It will also be used to support other services such as broadcast of weather or traffic conditions or infotainment to make a trip more pleasant to the passengers.

Multi-hop relay technology[3] is designed to provide capacity enhancement and coverage extension for wireless broadband access system such as WiMAX and LTE-Advanced. However, overall system performances worsen as the number of hop increases. For this reason, resource control function specifically Route selection problem should be tackled precisely so that better system performance can be achieved. Route selection or routing is a process to identify the best route to deliver information from source to destination by considering the constraints of available radio resource of the route. In this paper, we proposed a new route selection scheme named as Link Aware Route Selection Scheme (LARSS) for WiMAX Mobile Multi-hop Relay Networks aiming at maximizing network throughput and minimizing end-to-end delay. The proposed scheme exploits link quality and hop count as route metric. We conducted simulation study to evaluate the performance of the proposed scheme. Through the simulation, our proposed scheme outperformed the existing scheme in term of throughput and end-to-end delay.

It provides a survey of routing protocols in vehicular ad hoc networks. The routing protocols fall into two major categories of topology-based and position-based routing. The chapter discusses the advantages and disadvantages of these routing protocols, explores the motivation behind their design and trace the evolution of these routing protocols. Finally, it concludes the chapter by pointing out some open issues and possible direction of future research related to VANET routing.

In[5], Maritime wireless communication is different from terrestrial wireless communication, due to difference of environments. In this paper focuses to simulate on IEEE 802.16j for ship to ship maritime communication under various sea states. We describe BER level according to sea states under Rician fading channel model. BER patterns are designed using MATLAB. Conventional maritime wireless communication is based on voice communication, which is using radio devices of MF, HF, VHF and satellite system. The radio devices loaded to a vessel are decided according to the size and sailing area of vessels. IMO and ITU defined these radio types for sailing vessels. ESA (European Space Agency) introduced Wired Ocean Project. They suggest low cost and broadband IP based TV, internet and communication services using hybrid satellite and DVB (Digital Video Broadcast) systems in the ocean. But the cost is still expensive due to Wired Ocean needed TVRO (TV Receive Only) antenna for implementation.

In[6], Vehicular Ad hoc Network (VANET), a subclass of mobile ad hoc networks (MANETs), is a promising approach for the intelligent transportation system (ITS).

The design of routing protocols in VANETs is important and necessary issue for support the smart ITS. The key difference of VANET and MANET is the special mobility pattern and rapidly changeable topology. It is not effectively applied the existing routing protocols of MANETs into VANETs. In this investigation, we mainly survey new routing results in VANET. We introduce unicast protocol, multicast protocol, geocast protocol, mobicast protocol, and broadcast protocol. It is observed that carry-and-forward is the new and key consideration for designing all routing protocols in VANETs. With the consideration of multi-hop forwarding and carry-and-forward techniques, min-delay and delay-bounded routing protocols for VANETs are discussed in VANETs. Besides, the temporary network fragmentation problem and the broadcast storm problem are further considered for designing routing protocols in VANETs.

The temporary network fragmentation problem caused by rapidly changeable topology influence on the performance of data transmissions. The broadcast storm problem seriously affects the successful rate of message delivery in VANETs. The key challenge is to overcome these problems to provide routing protocols with the low communication delay, the low communication overhead, and the low time complexity. The challenges and perspectives of routing protocols for VANETs are finally discussed.

III.PROPOSED SYSTEM

Unicast and multicast routes are established in coordination with the scheduling of transmissions and bandwidth reservations in a way that bandwidth and delay guarantees can be enforced on a per-hop and end-to-end basis. The routes established in STORM are shown to be loop-free and realtime packets forwarded along these routes are shown to have bounded end-to-end delays. Results from detailed simulation experiments show that, compared to a protocol stack consisting of 802.11 DCF for channel access, AODV or OLSR for unicast routing, and ODMRP for multicast routing, STORM attains similar or better performance for elastic traffic, and up to two orders of magnitude improvement in end-to-end delays, with twice the amount of data delivery for real-time traffic while inducing considerably less communication overhead.

STORM MODEL:

The routing meshes established by STORM provide a fast and efficient way of repairing routes, because they contain extra paths that can be used in case of link breaks. This reduces the impact of node mobility on the quality of service perceived by real-time flows. In addition, the routing algorithm establishes enclaves, which restrict the dissemination of control information to those nodes that are likely to participate as forwarders of a given data flow, rather than the entire network.It uses four type of Interface Requirements are accessed.

User Interfaces

Graphical User Interfaces not in this product.

Users are communicated with Buttons with network animator.

Hardware Interfaces

Linux environment of system and basic need of system feature like random access memory etc

Software Interfaces

- This software is interacted with the TCP/IP protocol.
- This product is interacted with the and linux
- This product is interacted with the ServerSocket
- This product is interacted with TCL

Communications Interfaces

The TCP/IP protocol will be used to facilitate communications between the nodes.
STORM assumes that nodes share a single wireless channel organized into time frames consisting of a fixed number of time slots. The objective in STORM is to orchestrate the scheduling, routing, and traffic management functions of a multihop wireless network in a way that sources and destinations of flows perceive the network as a virtual link dedicated to the dissemination of those flows. Accessing the time slots of each frame is based on a combination of distributed elections of available time slots and reservations of time slots. For those time slots that have not been reserved, nodes use a distributed election algorithm based on hashing functions of node identifiers. A virtual link is created to support an individual real-time data flow and is implemented by a set of nodes located at directed meshes connecting sources to destinations. These meshes are computed by means of an interest-driven routing algorithm that establishes an ordering over the nodes based on their distances to the destination and the bandwidth available to them. To provide the abstraction of a virtual link, the routing algorithm also computes an end to- end channel access schedule for each data flow. The schedules generated by STORM are such that delay guarantees can be enforced on a per-hop and end-to-end basis. The end-to-end schedules are instantiated by the reservation protocol when the first data packet traverses the flow's routing mesh.

EVENT SCHEDULER:

This section talks about the discrete event schedulers of NS. As described in the Overview section, the main users of an event scheduler are network components that simulate packet-handling delay or that need timers. Figure 5 shows each network object using an event scheduler. Note that a network object that issues an event is the one who handles the event later at scheduled time. Also note that the data path between network objects is different from the event path. Actually, packets are handed from one network object to another using send (Packet* p) {target_->recv(p)}; method of the sender and recv(Packet*, Handler* h = 0) method of the receiver.

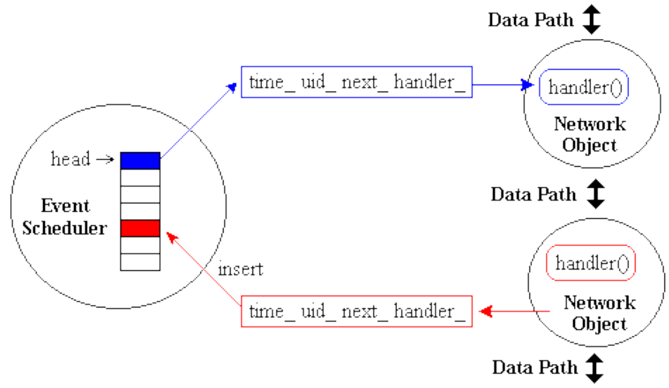

Fig .Discrete Event Scheduler

NS has two different types of event schedulers implemented. These are real-time and non-real-time schedulers. For a non-real-time scheduler, three implementations (List, Heap and Calendar) are available; even though they are all logically perform the same. This is because of backward compatibility: some early implementation of network components added by a user (not the original ones included in a package) may use a specific type of scheduler not through public functions but hacking around the internals. The Calendar non-real-time scheduler is set as the default. The real-time scheduler is for emulation, which allows the simulator to interact with a real network. Currently, emulation is under development although an experimental version is available.
The following is an example of selecting a specific event scheduler:

```
.            .            .
set    ns    [new    Simulator]
$ns    use-scheduler    Heap
. . .
```

Another use of an event scheduler is to schedule simulation events, such as when to start an FTP application, when to finish a simulation, or for simulation scenario generation prior to a simulation run. An event scheduler object itself has simulation scheduling member functions such as at time "string" that issue a special event called AtEvent at a specified simulation time. An "AtEvent" is actually a child class of "Event", which has an additional variable to hold the given string. However, it is treated the same as a normal (packet related) event within the event scheduler. When a simulation is started, and as the scheduled time for an AtEvent in the event queue comes, the AtEvent is passed to an "AtEvent handler" that is created once and handles all AtEvents, and the OTcl command specified by the string field of the AtEvent is executed. The following is a simulation event scheduling line added version of the above example.

```
.            .            .
set    ns    [new    Simulator]
```

```
$ns use-scheduler Heap
$ns at 300.5 "complete_sim"
. . .

proc complete_sim {} {
. . .
}
```

CHANNEL STRUCTURE AND TRAFFIC MANAGEMENT

Nodes share the same frequency band, and we assume that clock synchronization among the nodes in the network is achieved through a multihop time synchronization scheme such as the one implemented in Soft-TDMAC [6] which is a TDMA-based MAC protocol that runs over commodity 802.11 hardware. Nodes access the common channel assuming that it is organized using a time-division multiple access structure, which we call STORM frame. Each STORM frame is composed of N time slots (from slot 0 to slot N _ 1) and we use the position of a slot within the STORM frame as the identifier of the slot. A STORM frame does not have any particular structure and any time slot can be used to transmit a sequence of packets (signaling or data). There is only one special purpose time slot used to admit new nodes to the network. These admission time slots occur every A time slots, with A _ N, and are used by nodes to transmit their first hello packets on a contention basis. When a node is allowed to transmit over a time slot, it fits as many packets as possible in it.

Packets are selected from the local transmission queues, which are FIFO and are served using a priority-based algorithm. Reservation packets have the highest priority (pRsv), because quick consensus is needed on which nodes should have access to which time slots. The next priority is given to network-layer signaling packets (pctr), and data packets waiting in data queues have the lowest priority. Data queues can be either elastic or real-time, and real-time queues are assigned higher priority (pRT) than the priority given to elastic queues (pelastic), given that jitter and latencies are not as important for the latter. The priority of a real-time queue created for flow f is increased from pRT to pRTþ if the current time slot t was reserved on behalf of flow f. Hello packets are transmitted with the lowest priority (pHello_ < pelastic) if more than hello period=2 seconds but fewer than hello period seconds have elapsed since the last time a hello packet was transmitted, because there is no need for the information yet. However, if more than hello period seconds have elapsed, then the neighborhood information must be refreshed and hence the priority of the hello packet is set to pHelloþ > pctr. To summarize, during a time slot allocated to a node, the relationships among traffic priorities are: pHello_ < pelastic < pRT < pRTþ < pctr < pHelloþ < pRsv.

NEIGHBOR PROTOCOL

Routing, reservations, and transmission scheduling in STORM use distributed algorithms that require each node to know the nodes within its two-hop neighborhood. The neighborhood of a node consists of those nodes whose transmissions the node can decode, which we call one-hop neighbors, and the one-hop neighbors of those nodes are called two-hop neighbors. More formally, let G ¼ ðV ;EÞ be an undirected graph with a set of vertices V representing the set of nodes present in a wireless ad hoc network and a set of edges E. Any two nodes u and v share an edge ðu; vÞ 2 E if they are one-hop neighbors (i.e., within radio transmission range) of each other. For any node u 2 V, we denote INðuÞ ¼ fv : ðu; vÞ 2 Eg as the one-hop neighborhood of u and INðINðuÞÞ as the two-hop neighborhood of u. To gather two-hop neighborhood information, each node transmits hello messages periodically every hello period seconds, and each such message contains a list of tuples for the node itself and for each of its one-hop neighbors. Each tuple is composed of a node identifier, a list of the identifiers of the time slots reserved by the node, and the length of the list of reserved slots. Each node stores the last hello message received from each one-hop neighbor (or simply neighbor) in its neighbor list.

A neighbor is deleted from the neighbor list if no hello message is received from that neighbor in three consecutive hello periods. It is worth noting that the neighbor protocol in STORM is very similar to approaches used in traditional routing protocols that also require neighborhood information (e.g., OLSR) in that hello messages are transmitted unreliably but persistently, and convey information about local neighborhoods. The neighbor protocol is also used to detect when two nodes in a two-hop neighborhood have reserved the same slot. To resolve a conflicting reservation, the node with the larger identifier keeps its reservation over the particular slot, whereas the node with the lower identifier has to give up its current reservation and start a new reservation transaction over a different slot. The main source of these conflicting reservations is node mobility, which changes the neighborhood of nodes. The neighborhood information contained in hello messages allows nodes to detect these collisions before the conflicting nodes become one-hop neighbors.

Multicast Destinations and Core Elections

Upon reception of a MR, a multicast group member first determines whether it has received a MA from the core of that group within the last two MA-periods. If that is the

case, no further action is needed; otherwise, the receiver considers itself the core of the group and starts transmitting GARCIA-LUNA-ACEVES AND MENCHACA-MENDEZ: STORM: A FRAMEWORK FOR INTEGRATED ROUTING, SCHEDULING, AND TRAFFIC... 1351
MAs to its neighbors, stating itself as the core of the group. Nodes propagate MAs based on the best MA they receive from their neighbors. An MA with a higher core id is considered better than one with a lower core id. Therefore, if a node receives a MA advertising a core with a larger id than the current core, then the new core is adopted and a new MA advertising the new core is transmitted. Eventually, each connected component has only one core.

PERFORMANCE:

We present simulation results comparing STORM with ODMRP for the case of multicast traffic, as well as AODV and OLSR for the case of unicast traffic. In our experiments, ODMRP, AODV, and OLSR run on top of IEEE 802.11 DCF and all the protocols use a 802.11b physical layer.Weselected these protocols because they have become de facto baselines for performance comparisons of multicast, unicast, and channel access protocols. Even though they were not designed for real-time traffic, they are a good reference that allows us to highlight the performance gains of our approach. We use packet delivery ratio, generalized group delivery ratio, end-to-end delay, and total overhead as our performance metrics. To measure total overhead, we count all the packets generated by each protocol stack, which for the case of STORM includes data packets, MRs, MAs, hellos, and reservation packets.
The generalized group delivery ratio is a multicast-specific metric in which a data packet is considered as delivered, if and only if it is received by at least a given proportion of the multicast group members. This metric emphasizes the importance of group delivery by not considering packets that are received by a small subset of the group members. For this paper we set a threshold of
80 percent. The total overhead is computed as the average total number of packets transmitted by each node.

IV. Implementation & Results

Evaluate the impact of the Virtual transmissions approach presented. We measured the performance of STROM when the number of parents is varied from one to three. The results are obtained from based on the above topology. An extensive simulation model having scenario of n (user defined) mobile nodes and n UDP/TCP connections is used to study inter-layer interactions and their performance implications.

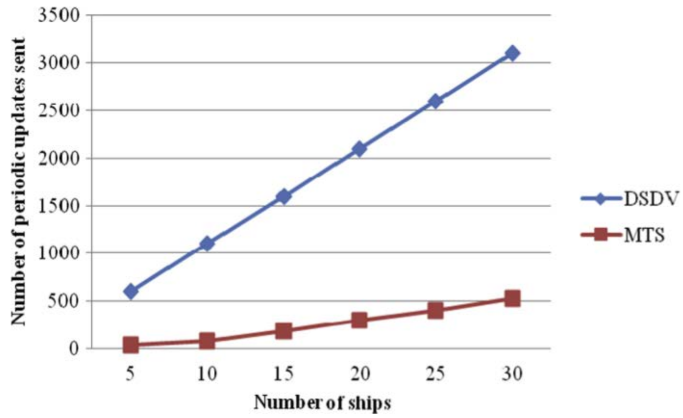

Fig.1: Comparison of periodic updates sent in the network

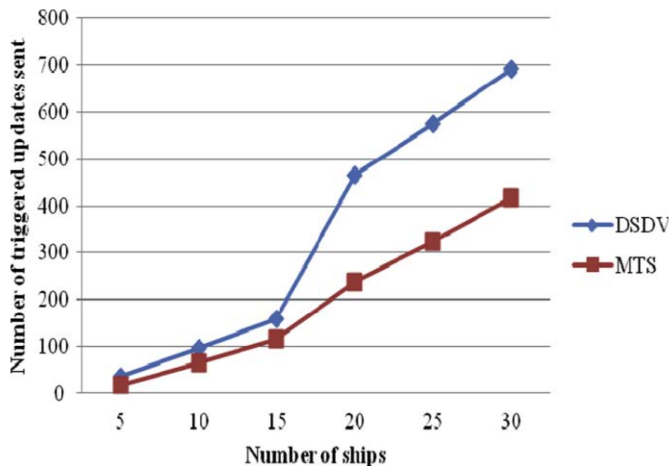

Fig.2: Comparison of triggered updates sent in the network.

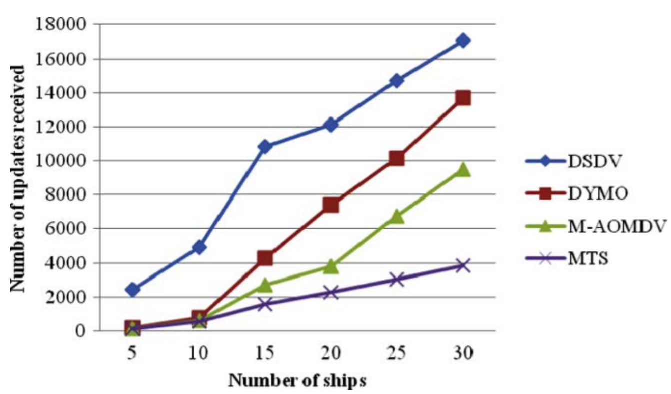

Fig.3: Comparison of total updates received in the network.

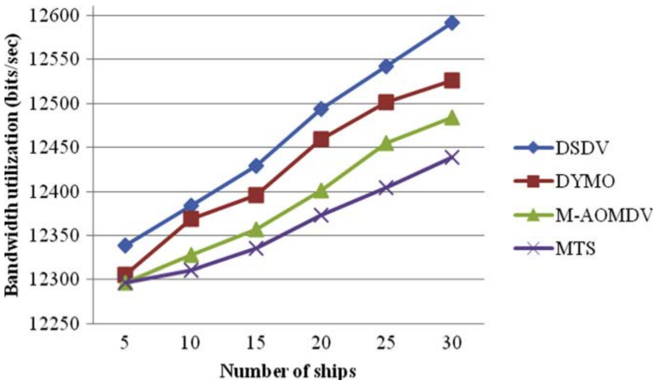

Fig.4: Comparison of bandwidth utilization for DSDV, DYMO, M-AOMDV, and MTS.

V. Conclusion

This project defines that a simple form of node cooperation can bring significant benefits to Vanet. Further, the benefits are higher for (i) heterogeneous networks than homogeneous ones and (ii) networks with random node placement than controlled deployments. In leveraging these benefits, project identified an inherent limitation in exploiting cooperation by directional and adaptive transmitters. This project then proposed a cooperation strategy that adapts to fading conditions and balances the trade-off between cooperation and antenna gain. In this project also proposed an efficient yet simple, distributed MAC protocol that incorporates the proposed cooperation mechanism.

References

[1] Daeseung Yoo and Gwangja Jin, “A Modified AOMDV Routing Protocol for Maritime Inter-ship Communication,” IEEE, International Conference Technology Communication (ICTC), 2011.

[2] [2]. Fonseca Antonio .A and Teresa Vazao .A, “Applicability of Position-Based Routing for VANET in Highways and Urban Environment,” INESC-ID/ Institution Superior Tecnico, 2011.

[3] [3]. Fisal .N, Satiman .N and Yusof .S.K.S, “An Efficient Link Aware Route Selection Algorithm for WiMAX Mobile Multi-hop Relay Networks,” Proc. IEEE Int’l Journal for Computing Applications, Volume 27-No 2, 2011.

[4] [4]. Mauve .M, Winder .A and Hartenstein .H, “A survey on position based routing in mobile ad hoc networks,” IEEE Newt 2011.

[5] [5]. Myeong Soo Choi, Beom-mu Kim and Kyung Ho Kim, “Ship to Ship Maritime Communication for e-Navigation Using IEEE 802.16j,”Proc. IEEE Int’l Advanced Science and Technology Letters (ASTL), pp. 110-115, 2013.

[6] [6]. Sing-Ling Lee, Yuh-Shyan Chen and Yun-Wei Lin, “Routing Protocols in Vehicular Ad Hoc Networks: A Survey and Future Perspectives,” Journal of Information Science and Engineering (JISE), 2013.