Mobile Ad hoc Networks for Ground Surveillance

Mathew McGee, Nirmala Shenoy
Rochester Institute of Technology
Rochester, NY, USA
mxm9106@rit.edu, nxsvks@rit.edu

***Abstract*— Ground surveillance networks are an important application of mobile ad hoc networks. The mobile nodes used in such applications benefit by a compact set of protocols that focus on reliable and timely data delivery. A solution that allows for closely integrated operation of routing, medium access control (MAC) and clustering is presented in this article, where clustering is used to improve data aggregation. The solution is evaluated for its performance and compared with two schemes using Optimized Link State Routing (OLSR) and Ad hoc On demand Distance Vector (AODV) routing over wireless LAN 802.11 MAC. Significant performance improvements indicate the potential for integrated approaches.**

Keywords - ground surveillance applications, MANET architectures, clustering, routing.

I. INTRODUCTION

Data aggregation for surveillance is an important application, which requires normally mobile *data collection nodes* to be deployed over the area of interest. From the collection nodes, *data is then aggregated* at a few nodes from where it may be sent to a center for further analysis. Surveillance applications require that data collection be done in a reliable and timely manner. One such application arises in networks used for rescue operations, where several mobile nodes collect data and aggregate them at one or more rescue centers. Text messages, low bit rate streaming data and voice would be the primary type of traffic in such scenarios. Due to the nature of the application, data aggregation over multiple hops becomes a necessity. Computationally non-intensive algorithms and protocols are preferred in such situations. The criticality of such applications and the constrained operational environment would further benefit if a minimal set of protocols that target the tasks in an efficient manner were used.

Some *desirable features* of such MANETs for surveillance can thus be listed as: 1) Multi-hop clustering for efficient data aggregation at few designated nodes; 2) robust connectivity and redundant paths to minimize data loss 3) a minimal protocol stack with low processing complexity to support timely data delivery. The *challenges* however to achieve the desirable are; 1) most clustering algorithms are single hop; multi-hop clustering require complex algorithms 2) proactive routing protocols, which result in reduced lead latency suitable for timely data delivery normally do not support redundant paths; 3) random access MAC protocols face high collisions and loss of data when nodes are mobile especially where packets have to be forwarded across multiple hops; 4) use of different algorithms for clustering and routing, and a MAC protocol that works independently results in protocol interaction issues and inefficiencies.

In this article, we introduce a novel MANET architecture that is highly suitable for critical surveillance applications that use mobile nodes. The architecture is built on the framework offered by an algorithm called the *Multi Meshed Tree* (MMT) algorithm [1]. This algorithm allows efficient coordination and operation of clustering, routing and MAC protocols in an integrated manner using a single address both at layers 2 and 3. A new protocol stack where clustering, MAC and routing operation are viewed as processes operating at a single layer is used. MMT algorithm allows creation of multiple multi-hop clusters, where in each cluster, the cluster head (CH) is the root of a meshed tree, and the cluster clients (CC) simultaneously reside on several tree branches originating from the root to create meshed trees. The random MAC protocol send bursts of data packets from a CC to the CH using sessions resulting in timely data aggregation.

We model the above using Opnet simulation tool and evaluate its performance in comparison with Optimized Link State Routing Protocol (OLSR) [8] and Ad hoc On demand Distance Vector Routing (AODV) protocol [4], operating over WLAN 802.11 at layer 2. The MMT routing protocols used in this work was the proactive version [1, 2]. Hence it was felt appropriate to compare with OLSR a standard proactive routing protocol. AODV is a reactive routing protocol and is supposed to have low overhead has been included in the studies to show the reduced control overhead with MMT routing protocol.

The rest of the article is organized as follows. In section II, we highlight related work in the different topics covered in the integrated solution. Section III presents the integration framework. Section IV briefly provides the simulation details and models based on Opnet simulation tool. Section V provides the graphs and performance analysis. Section VI concludes this paper by providing the relative performance improvements across the three schemes and their rationale highlighting the significance of the integrated approach.

II. RELATED WORK

To the best of our knowledge, there is no published work that integrates clustering, routing and MAC to operate based off a single algorithm and using a single address for surveillance MANETs. In this section, we hence present some related work conducted separately in the areas of random access based MAC protocols, routing protocols for

 ISBN: 978-1-61208-186-1

large MANETs and clustering techniques and conclude by highlighting the advantages of an integrated approach.

Clustering or zoning can be efficiently employed for the type of convergecast traffic encountered in surveillance networks, were the primary traffic flow is from CC to CH [9, 13]. In such cases proactive routing approaches are recommended as the routing is limited to the cluster or zone. However proactive routing algorithms require the dissemination of link state information to all routers in the zone, which can introduce latency in realizing or breaking a route, and high overhead. In the *Zone Routing Protocol* (ZRP) [10], each node pre-defines a zone centered at itself. and a framework is proposed, where any proactive routing protocol can be adopted within the zone. *Multi path distance vector zone* routing protocol [11] is an implementation of ZRP that uses multi path *Destination Sequence Distance Vector* for proactive routing. Another proactive approach, which works for groups of nodes, is *LANMAR* [12], which uses *Fisheye State Routing* [8].

Multi Hop clustering techniques such as the d-hop or k-hop clustering [13, 14] algorithms can offer flexibility in terms of controlling the cluster size and cluster diameter, but are often complex to implement.

Medium Access Control Protocols can be broadly categorized as scheduled and random access. Scheduled protocols require algorithms to schedule transmission turns for the nodes in the network, which could be achieved in a distributed or centralized manner. However in the case of surveillance networks with mobile nodes moving in a random manner, scheduling algorithms can be complex. Hence MAC protocols based on 802.11 are preferred [15].

Advantages of Single Algorithm and Interacting Modules: From the above discussions it would be clear that clustering and routing are normally treated separately and are based on different algorithms. Thus, to combine clustering and routing for an application it becomes essential to define an interworking mechanism that adds processing complexity and control overhead. When the MAC protocol operates independent of other protocols, efficient handling of time and loss sensitive packets diminishes. However, if all these operations can be based off a single algorithm, the complexity and overhead can be reduced considerably resulting in a protocol set that is highly efficient.

A similar approach was investigated for airborne surveillance networks of unmanned aerial vehicles travelling in circular trajectories [3]. The work was subsequently extended to an optimized MAC for ground surveillance networks where nodes are moving using random waypoint mobility models. Given the ground surveillance type of MANETs, the number of aggregation nodes is reduced by half in this article and the data traffic is streaming data instead of one MByte data files in [3].

III. THE INTEGRATION FRAMEWORK

A. The Multi Meshed Tree Algorithm

The MMT algorithm [1-2] to support the integrated approach will be briefly explained first. The formation of a single *meshed tree* based on the MMT algorithm is described with the aid of Fig. 1. The dotted lines connect nodes that are in communication range with one another at the physical layer. The node designated as CH is the root of the meshed tree. For ease in explanation, the meshed tree formation is kept simple and restricted to nodes that are connected to the CH by a maximum of 3 hops. At each node several values or IDs have been noted. These are the virtual IDs (VIDs) assigned to the node when it joins a tree branch in the meshed tree. Assume that the CH has a VID '1'. All nodes connected to this CH will have '1' as the first digit in their VIDs. Extending the above logic, a node gets a VID, which will inherit as its prefix the VID of the node upstream in the tree branch (the parent node), followed by a single (or multiple) digit(s) which indicates the child number under that parent. In Fig. 1, each arrow from CH is a tree branch that connects nodes to the root.

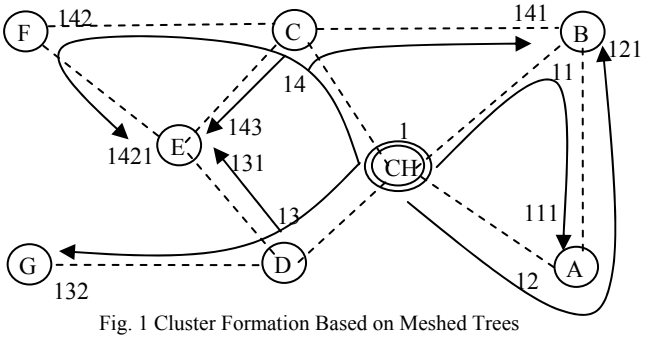

Fig. 1 Cluster Formation Based on Meshed Trees

Flexible Multi hop Cluster Formation: The size of the tree branch can be limited by limiting the length of the VID, which in turn allows control of cluster diameter. Each node that joins the cluster registers with the CH. This allows the CH to accept /reject a joining node to control the cluster size. The number of VIDs allowed for a node can control the amount of meshing in the tree branches of the cluster.

Multiple Dynamic Proactive Paths: The branches of the meshed tree provide the ***route*** to send and receive data and control packets between the CCs and the CH. The branch denoted by VIDs 14, 142 and 1421 connects nodes C (via VID 14), F (via VID 142) and E (via VID 1421), respectively, to the CH. Consider packet forwarding based on VIDs in which the CH has a packet to send to node E. If the CH decided to use E's VID 1421, it will include this as the destination address and broadcast the packet. Enroute nodes C and F will pick up the packet and forward to E. The VID of a node thus provides a virtual path vector from the CH to itself. Note that the CH could have also used VIDs 143 or 131 for node E, in which case the path taken by the packet would have been CH-C-E or CH-D-E respectively. Thus, between the CH and node E there are multiple routes as identified by the multiple VIDs. The support for multiple

 ISBN: 978-1-61208-186-1

proactive routes through the multiple VIDs allows for *dynamic route adaptability* to topology changes, as nodes request for new VIDs and joins different branches as their neighbors change.

Scalability: Lastly, a surveillance network can comprise of several tens of nodes; hence the solutions for surveillance networks have to be scalable [12]. We assume that several 'data aggregation nodes' are uniformly distributed among the non-data aggregation nodes during deployment of the surveillance network. Meshed tree clusters can be formed around each of the data aggregation nodes by assuming them to be roots of the meshed trees. Nodes bordering two or more clusters are allowed to join the different meshed trees and thus reside in branches originating from different CHs. Such border nodes will inform their CHs about their multiple VIDs under the different clusters. When a node moves away from one cluster, it can still be connected to other clusters, and thus the surveillance data collected by that node is not lost. By allowing nodes to belong to multiple clusters, the single meshed tree cluster can be extended to *multiple* overlapping *meshed tree* clusters that can collect data from several tens of nodes deployed over a wider area with very low probability of losing the captured data.

B. Burst Forwarding (BF) MAC

The VIDs acquired by a CC defines a path from the CC to the CH. Given that the paths in a MANET are transient and have short life times, the proposed MAC forwards several data packets (a burst) in a sequence (a session) over multiple hops. BF_MAC opens multi-hop data sessions using VIDs issued by the MMT algorithm. This allows for BF_MAC operation without additional 'address' overhead. The access is similar to the CSMA/CA protocol (WLAN 802.11), except that a data session between a CC and CH is started when a node succeeds in getting the channel. An exponential back off process is adopted. Contention window (CW) is handled as explained below.

EXP_ACK - The explicit acknowledge mode. A node that is either the final destination for a data session or is unable to forward a Request To Send (RTS) is in this mode.

CLEAR_TO_SEND: When a node senses an idle medium after its backoff count down, it goes into this mode. It continues in this mode till it senses a busy medium.

NOT_CLEAR_TO_SEND: A node that overhears transmissions in its neighborhood will go into this mode

DATA_SEND: when a node has data to send that originated from its application and receives an explicit or implicit CTS (explained below) is in this mode.

DATA_FORWARD: when a node that is not the originator for the data session receives a CTS for the RTS packet that it forwarded will enter this mode.

Session Establishment Procedure: When a node has data that arrives from its application layer it will first initiate the backoff process. On countdown to zero it will send an RTS packet, which contains the source VID, destination VID, and a hold time which is the total session time. This is calculated to include the time that the nodes in the path will be sending and forwarding packets as well as the time to establish the session.

If a node receives an RTS packet and determines that it is the next hop in the data session then it will forward the RTS onto the next hop node in the path specified by the VID. An RTS packet thus forwarded is overheard by the previous node on the session path and is treated as an Implicit CTS (IMP_CTS) packet. If a node receives an RTS packet and determines that it is the final destination for the session then it will send an explicit CTS (EXP_CTS). If a node receives an RTS and determines that it doesn't have the VID to involve in the session it will go into NOT_CLEAR_TO_SEND mode and remain silent for the hold time specified in the RTS packet.

Fig. 2 is used to explain the use of the modes of operation in BF_MAC. Node 'A' which has a VID 124 which defines its path to the CH (VID=0) has data to send to the CH. First node 'A' sends an RTS packet to node B (with VID 12). Node B receives the RTS as it is the next hop node in the session in the path (124) towards the CH. B forwards the RTS packet to node C. Node 'A' hears the forwarded RTS packet from 'B' and treats it as IMP_CTS. On receiving the IMP_ CTS, node A calculates the time it will take for the rest of the session to be established until an EXP_CTS is issued from CH and queues the first data packets to be sent. It then goes to DATA_SEND mode.

When node 'C' receives RTS from B it determines that it is on the path but is not the final destination so it forwards the RTS packet. When node 'B' receives the IMP_CTS it enters the DATA_FORWARD mode and is ready to receive and forward data packets. When the CH receives the RTS packet from node 'C' it sends an EXP_CTS, and the session is considered to be established.

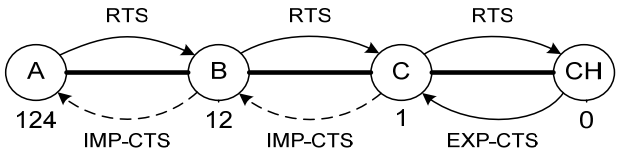

Fig. 2 BF-MAC Multi-hop Multi-packet Data Session

Non-Session and Non-Source Nodes: A node that overhears activity by it neighbors enters the NOT_CLEAR_TO_SEND mode. It also resets its contention window (CW), to the lowest value of 31. A node that is on a data session path but isn't the source node for the session will also set its CW to the minimum value at the end of the session for which it is currently forwarding the packets. This gives the non-session and non-source nodes a fair chance to get the media next time.

Data Sending and Forwarding: When the session's source node receives an IMP_CTS or EXP_CTS from the next node in the data session path it knows the data session is open to the next hop. If the packet was an IMP_CTS packet then the node calculates the time for session establishment and queues the first data packet for transmission at that point. If the packet was an EXP_CTS packet then the node will

 ISBN: 978-1-61208-186-1

begin sending the data packets because it knows the entire session path is just one hop.

The source node sends its first data packet and then waits for an EXP/IMP_ACK from the next node in the session path. When the next node in the data session receives the data packet it will modify the packet changing the sender's VID and the sender's UID to its VID and UID. It then checks it's mode, and if in DATA_FORWARD mode the node will forward the packet onto the next hop. When the previous node in the data session receives the forwarded data packet it interprets the packet as an IMP-ACK packet. A node in the session path that is in EXP_ACK mode will send an EXP_ACK packet back to the previous hop in the session path. If a node is the final destination then the BF_MAC will send the packet to the upper protocol layers. Else BF_MAC will queue the data packet in its own queue with a higher priority than it own data packets.

A source node will continue the above pattern of sending data packets until it has sent every packet for the session or has sent all of the remaining packets in the session. When the session ends at the source node it will double its CW. By doubling the contention window at the source node and resetting the contention window at all other nodes in range of the session, the BF_MAC effectively gives non source nodes a higher priority to begin their own data sessions.

Partially Established Sessions: Nodes in a session path will respond to RTS if they are in the CLEAR_TO_SEND mode only. Using the same example as above if node 'C' was in NOT_CLEAR_TO_SEND mode, perhaps from a session that was already established on the other side of the CH, node 'C' wouldn't respond to the RTS that node 'B' sent. Node 'B' would then timeout waiting for IMP_CTS and would hence enter EXP_ACK mode. The session would still be established from node A's perspective so it would send data packets to node 'B'. However, since 'B' is now in EXP_ACK it wouldn't forward the packets onto node 'C'. Instead it would modify the data packet's sender UID and VID fields just the same way it would if it were forwarding the packet, but it would put the packets on the top of its data queue and subsequently try to establish a session to the CH to forward the packets. Once node 'A' receives EXP_ACK for its data packets it would clear all related information.

Priority Queues: The BF_MAC maintains three queues and in each queue packets are inserted based on their type to provide a second level of priority within that queue.

- One queue stores the configuration or hello packet and has the highest priority. Only the latest packet is stored.
- *Route Break* packets, initiated by a parent node on discovering that one of its children is not connected are placed at the top of a high priority queue followed by disconnect packet, generated by a child node on detecting disconnection from its parent nodes. Next in the queue are data packets that are in transit and have to be forwarded for other nodes. Last in this queue are the Registration Reply used for MMT cluster formation operations. In surveillance application most of the traffic is travelling from a CC to a CH, while the Registration Reply packets go from a CH to a CC i.e., in the opposite direction. Hence these packets were included in the high priority queue.
- Data packets originating from a node are placed at the top of the normal priority queue, which is the third queue. This is followed by Registration Requests, followed by Registration Update packets sent by nodes to inform other CHs, when they join a new branch in a cluster, and finally Registration Acknowledgements by a newly joining node, confirming to the CH its success in joining a cluster. The MMT routing protocol will only pass a data packet down to the BF-MAC layer if a route exists. Hence data packets were given a higher priority over creating new routes to prevent the already established routes from expiring while new routes were being created.

SIFS Timer: BF_MAC uses short inter-frame space (SIFS), between the end of transmission of a packet and the beginning of the next packet to avoid collisions. In Fig. 3, when node 'B' sends an RTS packet and node 'A' receives the RTS, it will wait for SIFS time before forwarding the RTS onto the CH. CH waits SIFS time before sending EXP_CTS. This applies through the entire data session.

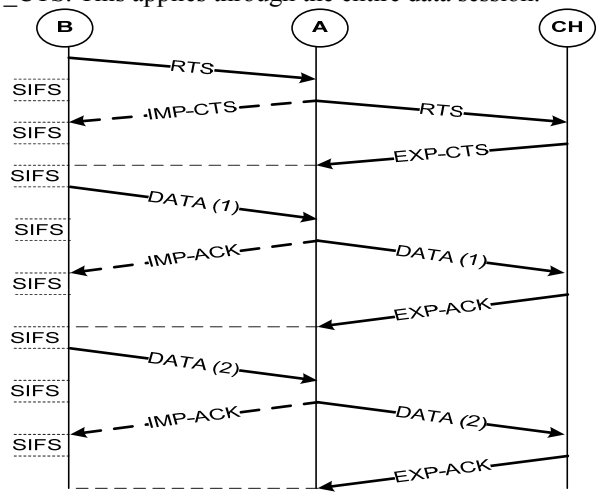

Fig. 3 Use of SIFS in BF_MAC

IV. SIMULATIONS

The proposed solution was modeled using the Opnet. Aggregation nodes were designated and allowed to move within a coverage area of surveillance network as shown in Fig. 4, to limit path distance between data collection and aggregation nodes. This would provide a best case scenario of data collection for all schemes.

Random walk mobility model was used for node movement as the study focused on ground surveillance. Node speeds were varied from 3 m/s, 5 m/s to 10 m/s. 'Hello' interval for all schemes was set to 2 seconds. Three different scenarios, one with 20 nodes and 2 aggregations nodes, second with 40 nodes and 4 aggregation nodes and lastly 80 nodes and 8 aggregation nodes.

The graphs presented capture the performance when all data collection nodes are each sending 10 Kbyte files in intervals of 1 second for the 20 node scenario and 2 seconds for the 40 and 80 nodes scenario. The 2 second interval was selected for the higher node scenario to reduce congestion.

 ISBN: 978-1-61208-186-1

Due to the sequential file transfers, the traffic pattern resembles streaming data, with the exception that packet arrival was modeled as five packets (size 2 Kbytes) per one (or two) seconds. At the physical layer, packets with single *bit error rates* were dropped and *forward error correction* was not enabled. All other physical layer parameters were as provided in the standard Opnet 802.11 WLAN models. The data rates were maintained at 11 Mbps.

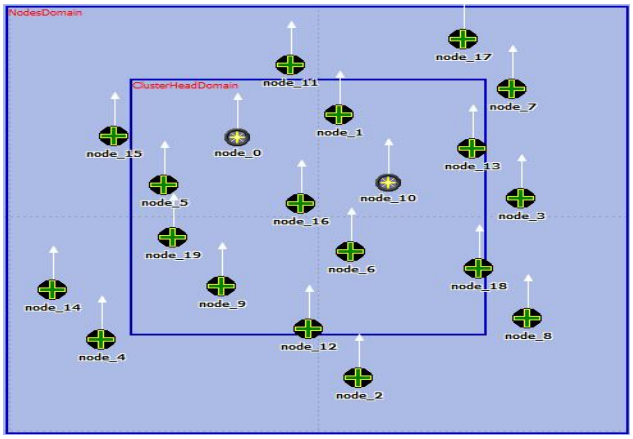

Fig. 4 CHs and CCs in a 20 node scenario

V. PERFORMANCE ANALYSIS

Success rate and latency in packet delivery and control overhead were recorded. Control overhead was calculated as percentage of control traffic to total traffic in bits.

Success Rate: Figures 5A, B and C respectively are plots of success rate achieved with OLSR, AODV and MMT when the mobile node speeds were varied from 3, 5 to 10 m/s. On the x axis is the number of nodes in the scenario. So the plot shows the performance variations as the number of nodes increase in the network and thus its scalability.

With node speeds of *3 m/s,* the success rate of MMT based solution drops from 98% to 96% as network size increases from 20 to 80 nodes. Success rate for OLSR drops from 94 to 91%, while AODV dropped from 93% to 85%.

As the scenario is one of data aggregation, and the data aggregation nodes are explicitly identified as the destination nodes and zone restricted OLSR scales better that AODV. Moreover reactive routing schemes do not perform well when the number of sending nodes is high – and the tests conducted in these cases had all data collection nodes sending files simultaneously to the data aggregation nodes.

When the speeds of the nodes were increased to *5 m/s,* the gap between MMT and OLSR based schemes shows an increase. MMT still maintains a success rate between 97% with 20 nodes to 93% with 80 nodes. While OLSR drops from 90 % scenario to 86% for the 80 node scenario, AODV success rate dropped down to 81% for the 80 node scenario.

OLSR performance degrades faster with increasing node speeds compared to MMT and ADOV, which is further noticed in the plot where node speeds were increased to *10 m/s* in Fig. 5C. The plot for OLSR gets closer to the plot for

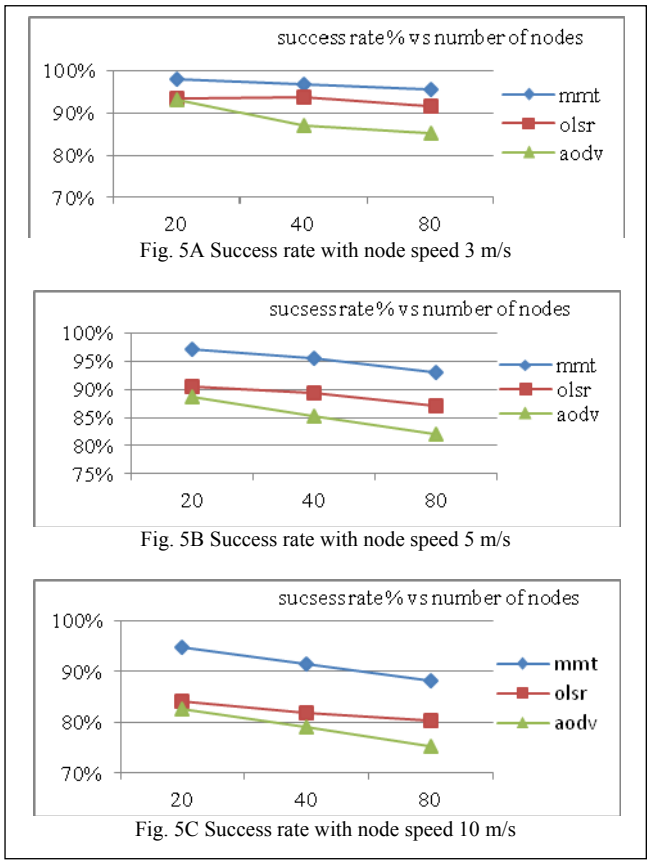

Fig. 5A Success rate with node speed 3 m/s

Fig. 5B Success rate with node speed 5 m/s

Fig. 5C Success rate with node speed 10 m/s

AODV, while the gap between MMT plot and OLSR plot increases. To summarize in Fig. 5C, MMT success rate is 8% higher than OLSR success rate for the 80 node scenario.

The inference from the three plots would be that with the same settings when node speeds (in this case all nodes) increase MMT performance deteriorates by 3 to 5% (20 nodes to 80 nodes), OLSR deteriorates by 9 to 11% (20 nodes to 80 nodes) while AODV deteriorates around 10%.

Overhead: Figures 6A, B and C are respectively the plots for control overhead expressed as a percentage to the total traffic as node speeds were varied from 3, 5 to 10 m/s. MMT based solutions show an overhead of 10% maximum with 80 nodes with node speeds of 3 m/s which goes to 19% when the node speeds were increased to 10 m/s. OLSR shows a consistent overhead of nearly 60% even in the 80 node scenario. This is because OLSR is not adaptive to dynamic topology changes as MMT i.e., OLSR sends hello packets and TC packets at a certain interval, which are independent of node movement or topology changes. AODV exhibits an overhead that varies from 60 to 68%, because it is also an adaptive protocol. However it is unable to cope in successful packet delivery as the node speeds and the network size increases, which was apparent in Fig. 5.

 ISBN: 978-1-61208-186-1

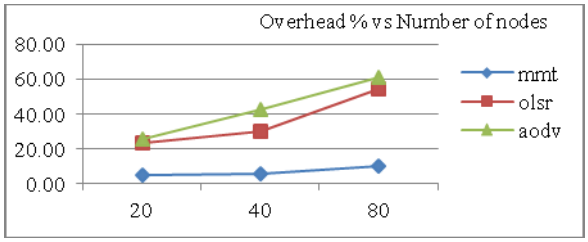

Fig. 6A Overhead % with node speed 3 m/s

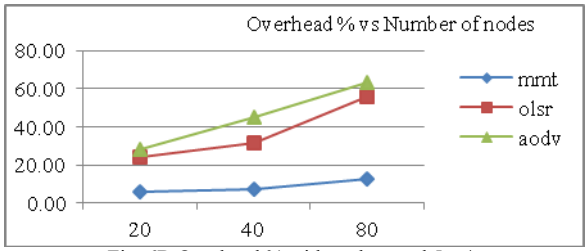

Fig. 6B Overhead % with node speed 5 m/s

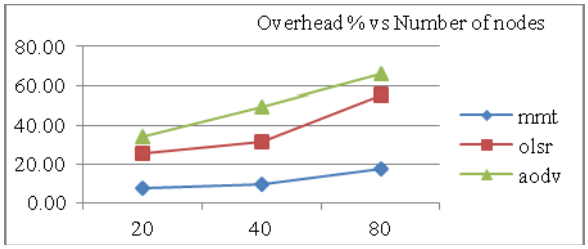

Fig. 6C Overhead % with node speed 10 m/s

Latency: Figures 7 A, B and C are the plots of average end to end latency incurred by the successfully delivered packets for varying node speeds.

MMT based solutions have an average end to end packet delivery latencies of 0.01 seconds when the node speeds were maintained at 3 m/s. OLSR exhibits slightly higher latencies. However ADOV latency varies from 0.04 seconds to 0.12 seconds when the network size increases from 20 nodes to 80 nodes. This can be explained when the average hops encountered in each case is considered next.

The average end to end packet delivery latency with OLSR is slightly better than that of MMT when the node speed is increased to 10 m/s. However t MMT successfully delivered 8 to 12 % more traffic. The increase in latency can also be accounted for when one looks at the average number of hops that the packets were delivered over in the case of MMT and OLSR. AODV on the other hand is significantly disadvantaged at higher node speeds and larger network sizes due to the fact that all non-aggregation nodes are sending data traffic. Fig. 7D is the plot for the maximum latency encountered when the node speed was maintained at 3 m/s. This graph has been provided just to show that while OLSR and MMT still maintain low maximum latencies AODV packets experience very high latencies – up to 30 seconds. However when the network size increases, this latency drops which can be attributed to the fact that the traffic delivered successfully has dropped considerably and in the case of a large network as the path length increases (discussed next) many packets are not delivered.

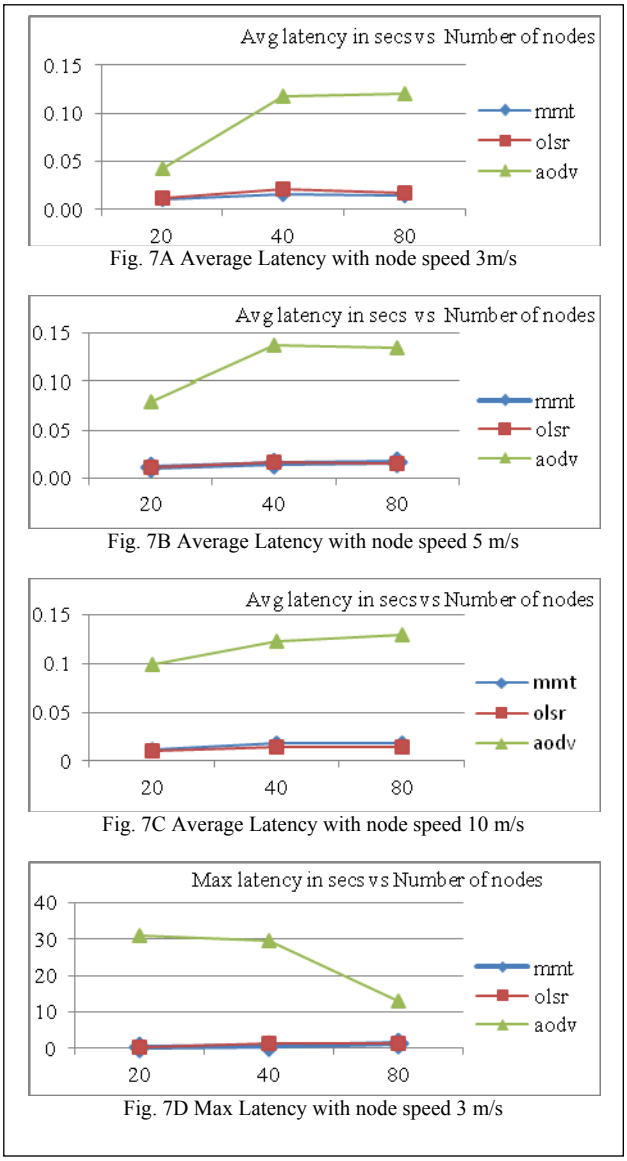

Fig. 7A Average Latency with node speed 3m/s

Fig. 7B Average Latency with node speed 5 m/s

Fig. 7C Average Latency with node speed 10 m/s

Fig. 7D Max Latency with node speed 3 m/s

Path Length in Hops: Figures 8A and B are the plots for path lengths encountered in the three schemes. These plots help understand the performance trends in previous graphs. We include only one set in this case when the node speed was maintained at 3 m/s and record the average and maximum hops encountered. Fig. 8A is a plot of maximum hops for the three schemes, when the node speed was maintained at 3 m/s. AODV records maximum hops of 12 with a network size of 80, which goes to show the lower success rate and high latencies encountered by AODV.

Fig. 8B on the other hand is the plot of the average hops. MMT recorded an average of 1.5 hops for the 80 node scenario, OLSR recorded average hops of 1.2, while AODV recorded 2.1 hops. It is worth noting that OLSR recorded and used the shortest paths among the three schemes. This is

 ISBN: 978-1-61208-186-1

because it collects the topology information and uses Dijkstra's algorithm to compute the shortest path. MMT on the other hand focuses on route robustness and quick dynamic adaption to topology changes with an intention to keep nodes connected. Hence while in OLSR the routing table was not updated due to the lack of timely collection of topology information, MMT continued to dynamically maintain multiple routes for every node, which were updated as nodes moved and the neighbors changed (with lower overhead). AODV recorded a high value in maximum path lengths but, the low average value indicates that such situations were rare.

VI. CONCLUSIONS

In this article, we introduced a new architecture for MANET use in critical surveillance applications. The goal is include the minimal set of functions across the protocol layers to facilitate timely and reliable delivery of data at a few aggregation nodes. For this purpose an integration framework was developed based on the MMT algorithm. The algorithm allowed integrated operation of clustering, proactive routing and MAC using a single address. A new technique of random access is introduced.

The proposed solution was evaluated and compared with standard OLSR and AODV routing protocols operating over WLAN 802.11 MAC. To the best of our knowledge this is the first article that compares such MANET performance under stressful operating conditions. Furthermore the article also brings to light the good performance achievable with OLSR for surveillance type MANETs, if the operating conditions are set accordingly. Reasons for AODV's performance deterioration under such scenarios are also explained. Lastly the proposed MMT based solution is shown to outperform both AODV and OLSR when node speeds and the network size increases, given that the operational parameters are maintained constant.

ACKNOWLEDGMENT

This work was supported by funding from Office off the Naval Research (ONR), USA.

REFERENCES

[1] Nirmala S., Pan Y., Narayan D., Ross D. and Lutzer C., "Route Robustness of a Multi-meshed Tree Routing Scheme for Internet MANETs", Proceeding of IEEE Globecom 2005. 28 Nov – 2nd Dec. 2005 St Louis, pp 3346-3351.

[2] Martin N., Al-Mousa Y. and Shenoy N, "An integrated routing and medium access control framework for surveillance networks with mobile nodes", ICDCN 2010, Bangalore, India. pp 315-323.

[3] Abolhasan M., Wysocki T. and Dutkiewicz E., "A review of routing protocols for mobile ad hoc networks", Journal of ad hoc networks, Elsevier publications, 2004, pp 1-22.

[4] Perkins C., E., Royer E. M., and Das S. R., "Ad Hoc On-Demand Distance Vector (AODV) Routing", IETF Mobile Ad Hoc Networks Working Group", IETF RFC 3561

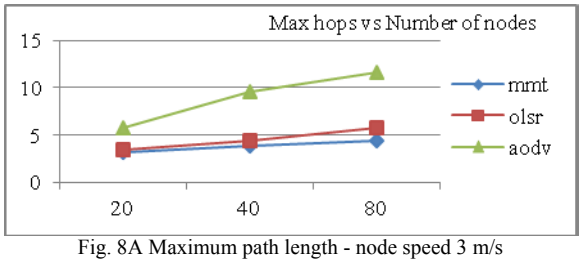

Fig. 8A Maximum path length - node speed 3 m/s

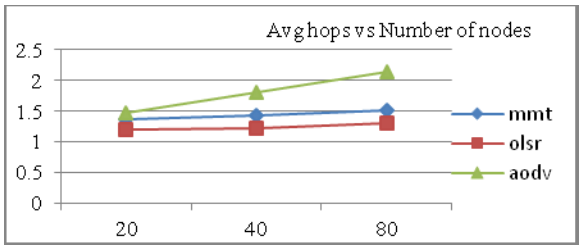

Fig. 8B Average path length – node speed 3 m/s

[5] Clausen T., Ed. and Jacquet P., Optimized Link State Routing Protocol (OLSR), Network Working Group, RFC: 3626

[6] Basagni S., Chlamtac I., and Farago A., "A generalized clustering algorithm for peer-to-peer networks", Workshop on Algorithmic Aspects of Communication, July 1997, pp 1-15.

[7] Hong X., Xu K. and Gerla M., "Scalable Routing Protocols for Mobile Ad Hoc Networks", IEEE Network Journal, July/Aug 2002, Vol 16, issue 4, pp 11-21.

[8] Pei G., Gerla M., and Chen T.-W., "Fisheye State Routing: A Routing Scheme for Ad Hoc Wireless Networks," IEEE International Conference on Communications, 2000, Vol 1, pp 70-74.

[9] Pei G., Gerla M., Hong X., and Chiang C. -C., "A Wireless Hierarchical Routing Protocol with Group Mobility," in Proceedings of IEEE WCNC'99, New Orleans, LA, Sept. 1999. pp 1583-42.

[10] Haas Z.J. and Pearlman M.R., "The Performance of Query Control Schemes for the Zone Routing Protocol," ACM/IEEE Transactions on Networking, vol. 9, no. 4, August 2001, pp. 427-438.

[11] Ibrahim, I. S., Etorban, A. and King, P.J.B., "Multipath Distance Vector Zone Routing Protocol for Mobile ad hoc networks (MDVZRP)", The 9th PG Net, Liverpool John Moores University, UK, pp. 171-176, 23-24 June 2008

[12] Pei G., Gerla M. and Hong X., "LANMAR: Landmark Routing for Large Scale Wireless Ad Hoc Networks with Group Mobility," in Proceedings of IEEE/ACM MobiHOC 2000, Boston, MA, Aug. 2000, pp. 11-18.

[13] Lin, C.R. and Gerla, M., "Adaptive clustering for mobile wireless networks," Selected Areas in Communications, IEEE Journal on , vol.15, no.7, pp.1265-1275, Sep 1997

[14] Basagni, S., "Distributed and mobility-adaptive clustering for multimedia support in multi-hop wireless networks," Vehicular Technology Conference, 1999. VTC 1999 - Fall. IEEE VTS 50th , vol.2, no., pp.889-893 vol.2, 1999

[15] Grönkvist J., A. Hansson A. and Nilsson J., "A comparison of access methods for multi-hop ad hoc radio networks", IEEE Vehicular Technology Conference, 2000, pp. 1435–1439.

 ISBN: 978-1-61208-186-1