7-1-2016

Context Aware Routing Management Architecture for Airborne Networks

Mark Silvius
Air Force Institute of Technology

J. Addison Betances
Air Force Institute of Technology

Kenneth M. Hopkinson
Air Force Institute of Technology

Recommended Citation

Silvius, Mark; Betances, J. Addison; and Hopkinson, Kenneth M., "Context Aware Routing Management Architecture for Airborne Networks" (2016). *Faculty Publications*. 115.
https://scholar.afit.edu/facpub/115

IET Networks

Research Article

Context aware routing management architecture for airborne networks

ISSN 2047-4954
Received on 12th June 2015
Revised on 21st February 2016
Accepted on 7th March 2016
doi: 10.1049/iet-net.2015.0064
www.ietdl.org

Addison Betances, Kenneth Mark Hopkinson ✉*, Mark Silvius*

Department of Electrical and Computer Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH 45433, USA
✉ *E-mail: kenneth.hopkinson@afit.edu*

Abstract: Military environments require highly dynamic mobile *ad hoc* networks (MANETs) to meet operational mission requirements. Decision makers rely on the timely delivery of critical battlefield information to make informed determinations quickly and as accurately as possible. However, traditional MANET routing protocols do not provide quality of service (QoS). Furthermore, they do not implement active controls to minimise the impact of network congestion. This study proposes the use of the information embedded in an air tasking order (ATO) during the planning phase of military missions to optimise the network performance. The trajectories of relevant nodes (airborne platforms) participating in the MANET can be forecasted by parsing key information contained in the ATO. Using this idea it is possible to optimise network routes to minimise edge overutilisation and increase network throughput. In one simulated test case, there was a 25% improvement of network throughput, and 23% reduction on dropped packets. Using this technique, the authors can selectively preserve the QoS by establishing network controls that drop low-priority packets when necessary. The algorithm improves the overall MANET throughput while minimising the packets dropped due to network congestion.

1 Introduction

There are several challenges that are inherent to airborne networks. Protocols designed to be used with the Internet do not perform well in an exclusively military environment. The main objective for the protocols that are currently used to provide services over the Internet is to deliver electronic data between network nodes reliably by detecting and avoiding faulty network paths. Traditional Internet protocols do not incorporate measures to react quickly enough to dynamic networks where the topology changes frequently. Network management architectures designed for terrestrial wire-line networks cannot be easily ported to airborne network due to their reliance on persistent connections [1]. Classical routing algorithms are inefficient in high mobility and scalability scenarios [2] as normally encountered in military operations. Additionally, adversaries might resort to interference or jamming to degrade network performance. Existing protocols cannot handle this range of challenges.

There are several algorithms designed to operate efficiently with the constant topology changes, which are inherent in mobile *ad hoc* networks (MANETs). MANET routing algorithms can be classified as: reactive, proactive, and hybrid. Reactive protocols discover the routes as needed in order to minimise traffic overhead. Proactive protocols create routing tables to every other node participating in the network, and optimise traffic among nodes by computing the shortest path. Hybrid routing protocols combine techniques from proactive and reactive protocols in an attempt to achieve fast route discovery while minimising overhead traffic. The performance of these MANET routing protocols varies depending on the operating conditions. There is no single routing protocol that outperforms every other routing protocol in all circumstances [3].

The United States Department of Defense (DoD) is currently interested in the development of new MANET routing solutions and has funded research efforts to develop an algorithm that performs optimally in a military environment [4]. During military operations node mobility is driven by mission requirements. These preplanned requirements are documented in the air tasking order (ATO) [5]. During the ATO generation phase, it is possible to gather information about the trajectories of the participating airborne platforms. Additionally, specifics about the type of network support required to execute the mission are obtained. By considering the capabilities of each participating node and how they are used, it is possible to generate efficient network routes optimising the performance of the overall network. This information can be used to prioritise packets and move data more effectively.

The DoD is currently implementing the global information grid (GIG): a globally interconnected network that provides capabilities for collecting, processing, storing, disseminating, and managing information on demand to warfighters, policy makers, and support personnel. GIG capabilities will be available from all operating locations: bases, posts, camps, stations, facilities, mobile platforms, and deployed sites [6]. One of the primary GIG objectives is to ensure that the networks used to support military operations perform optimally so they can provide mission feedback in a timely manner.

This paper describes an algorithm that maximises bandwidth utilisation in a military airborne network environment. This algorithm uses an off-line environment to compute optimum routes prior to mission execution. The algorithm seeks to take advantage of the data collected during the mission planning phase for military tactical environments such as the ATO generation process. The algorithm determines network packets priority and scheduling for all participating nodes to maximise utilisation. A prioritised list of periodic and predicted unicast and multicast network packets necessary to conduct the mission is stored in an information exchange requirements (IERs) database. The algorithm uses the IERs available during the mission planning stage as input to compute an optimum schedule with respect to available and accessible bandwidth.

Network traffic can be significantly optimised when the topography and traffic pattern of the source nodes of the network can be determined ahead of time. An obvious approach to compute an optimal routing path is to find the shortest paths between all pairs using Floyd–Warshall algorithm, or Dijkstra's algorithm in the case that all traffic has a single destination. However, a solution based on shortest paths algorithms may fail to provide optimum routing tables when the network traffic is high enough to saturate some of the edges of the network. This paper proposes a solution based on the maximum concurrent multi-commodity network flow that satisfies the network traffic for all nodes without violating the capacity constraints of the edges.

The rest of this paper is organised as follows. Section 2 discusses related research in the area of network predictions. Section 3 describes the concept of a network tasking order (NTO). Section 4 presents a brief description of multi-commodity flows. Section 5 describes the maximum concurrent multi-commodity flow heuristic that we make use of when adjusting routes based on traffic flows and edge constraints. Section 6 describes the experimental methodology and the scenario used in conducting the experiments. Section 7 presents experimental results of the experiments. Section 8 provides conclusions and recommendations for future work.

2 Network predictions

Traffic prediction plays an important role in network control and protocols. It is possible to implement automatic network controls that adjust network routes to minimise congestion based on accurate projections of the future network's state. To implement such a system, the prediction interval should be large enough to offset delays caused by traffic measurements, network topology changes, modify traffic priority etc.

Significant bandwidth in MANETs is consumed by messages broadcast to advertise topology changes, network routes, traffic changes etc. A solution to this problem is to predict the future state of the system far enough in advance so that network controls can avoid congestion and utilise the available network capacity efficiently. It is important to determine how far into the future can network traffic be predicted with confidence because the prediction accuracy decreases significantly as the prediction interval is increased. Automatic network controls that make decisions based on erroneous predictions can severely impact network performance [7].

Stuckey *et al.* [8] developed an algorithm that can predict future network status based on the present conditions. Their prediction algorithm is implemented using a Kalman filter to predict future network queue sizes. This algorithm can accurately predict queue size several seconds into future. These queue sizes predictions can be used in network control algorithms to optimally manage the network, thus optimising metrics of interest such as delay or throughput.

Haught *et al.* [9] performed simulations using a static scenario to validate the effectiveness of Stuckey's Kalman filters network congestion prediction algorithm in scenarios involving various Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) flows with exponential arrival rates and a number of Kalman filter prediction points. The Kalman filters were able to accurately predict the network flow several seconds into the future, making a prediction every second for the 500 s of simulation time. It was noted that predictions too far into the future were inaccurate.

Haught performed additional experiments using static scenarios adding a dynamic routing queue controller (DRQC) that managed the network traffic into the network. The DRQC adjusted network traffic based on current queue sizes as well as predicted queue sizes and network flows. A significant improvement was noted in the overall network throughput when the DRQC was introduced into the network measured by a drastic change in network congestion. The accuracy of the queue size prediction significantly affected the performance of the DRQC [9].

3 Network tasking order

The ATO is a human/machine readable document created by Joint Air Planners that define the daily schedule of air related operations. ATOs specify targets, callsigns, air controlling agencies, type, and number of aircraft allocated to a target, mission types, routes (including time and position), frequencies in use [10] etc. The ATO is detailed enough to determine what type of network support is necessary to accomplish the mission. An analogue to the ATO, called the NTO, could be created that specifies network assets and operations extracted from the ATO.

A simple example, depicted in Fig. 1, illustrates the NTO's benefits. In this example, we have two nodes: Source 1 and Source 2. Each node has data to transmit to the same destination node. Source 1 is transmitting a video at 30.6 kilobits per second (kbps) and based on the requirements specified in the ATO it has been determined to be a high-priority transmission. Source 2 is collecting data from a remote sensor at 7.2 kbps and based on the ATO requirements it has been classified as low priority. Both sources are connected to the destination node via the same router. The two sources, the destination, and router are each slow moving or fixed, particularly when compared with the depicted aircraft. There is a single link between the router and the destination node. The capacity on this link can only accommodate a maximum data transfer of 36 kbps. From the information above, the combined data rate for both sources is 37.8 kbps. Hence, the available bandwidth is not enough to allow data transmitted from both sources simultaneously to be delivered to the destination node.

One of the challenges typically encountered in MANETs using traditional routing protocols is highlighted above when not all of the available resources are utilised. However, using information collected from the NTO, an optimal alternative topology can be generated. The NTO provides enough information to leverage networking resources not normally considered by traditional routing protocols. For example, in the previous scenario the airborne platform can be used to ferry data between Source 2 and the destination.

4 Multi-commodity flow

Traditional network environments use an implementation of a shortest-path algorithm to generate optimum routes, but this type of algorithm does not achieve optimal results in a network environment where the nodes are moving, as in a MANET. These limitations to shortest-path algorithms are well known.

In a dynamic network environment under heavy traffic load, shortest-path routing algorithms, particularly those that attempt to adapt to traffic changes, frequently exhibit oscillatory behaviours, and cause performance degradation [11].

A multi-commodity flow is a type of network flow with multiple sources and destinations that maximises the amount of flow travelling from the various sources to their corresponding destinations subject to the capacity constraints. A multi-commodity flow instance can be defined as a set of ordered pairs of vertices (S_1, T_1), (S_2, T_2), …, (S_K, T_K), where each pair (S_i, K_i) represents a commodity with source S_i and target T_i. For each commodity (S_i, K_i) a non-negative demand d_i is specified [12]. The objective is to maximise the amount of flow travelling from the sources to the corresponding destinations. There are several

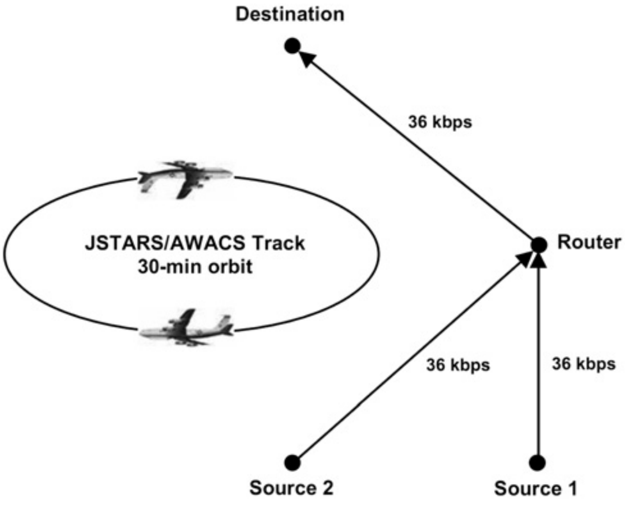


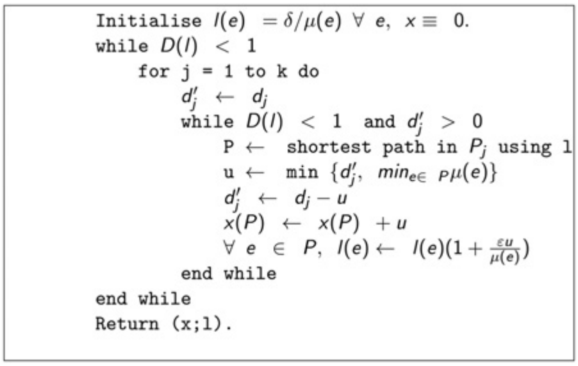
Fig. 1 *Sample NTO scenario*

algorithms that can solve some specific instances of the multi-commodity flow problem in polynomial time using linear programming if we allow fractions for the flow demands [12].

A network routing solution based on multi-commodity flow is normally problematic because such problems are intractable if we constrain the flow demands to be integers. Shimon Even from The Weizman Institute of Science demonstrated in 1975 that the multi-commodity integral flow problem is non-deterministic polynomial-time complete even if the number of commodities is two [13]. The problem can be solved in polynomial time using linear programming if fractional flows are allowed. In reality, an algorithm that computes the fractional solution to within 1% is normally adequate for a practical implementation that can be used to determine network routes [14]. It is possible to obtain an approximation of the optimum solution arbitrarily close in polynomial time much faster than algorithms based on linear programming [15].

5 Maximum concurrent multi-commodity flow

The objective of this process is to compute optimum routes for the MANET given the predicted network topology and traffic flows. The algorithm computes these routes using a fully polynomial-time approximation scheme for the maximum concurrent multi-commodity flow paradigm. The algorithm computes an approximation using fractional flows, as opposed to the actual solution, to minimise the computing resources required. However, the approximation scheme used in this paper establishes an error bound that can be set arbitrarily close to the actual solution [16]. The running time of the algorithm increases as the error is minimised. In practicality, an approximation that is within 1% of the actual solution is sufficient to determine optimum MANET routes. The complete pseudocode of the maximum concurrent flow is shown in Fig. 2.



```
Initialise $l(e) = \delta/\mu(e)\ \forall\ e,\ x \equiv 0.$
while $D(l) < 1$
    for j = 1 to k do
        $d'_j \leftarrow d_j$
        while $D(l) < 1$ and $d'_j > 0$
            P ← shortest path in $P_j$ using l
            u ← min $\{d'_j,\ min_{e\in P}\mu(e)\}$
            $d'_j \leftarrow d_j - u$
            $x(P) \leftarrow x(P) + u$
            $\forall\ e \in P,\ l(e) \leftarrow l(e)(1 + \frac{\varepsilon u}{\mu(e)})$
        end while
end while
Return (x;l).
```

Fig. 2 *Maximum concurrent flow algorithm [16]*

6 Experimental methodology

This section outlines the methodology used to determine the applicability of network predictions obtained via the NTO process to improve MANET performance when compared with traditional routing solutions. An outline of the experiments performed is given along with the expected results and performance factors used to measure the suitability of the proposed solution.

The objective of this paper is to develop an algorithm that capitalises on the information obtained via the NTO process. The algorithm computes optimum routes based on predicted network traffic and topology. This part of the algorithm is executed in an off-line environment prior to mission execution. The routes generated in this process are preloaded in all participating nodes.

The NTO process provides estimates of the network performance over a course period of time based on limited information available several hours before mission execution. It is possible to encounter significant topology or network traffic deviations from these estimates during actual mission execution. Therefore, the algorithm implements agents that monitor the network traffic during mission execution to detect deviations from the NTO process. The online agent generates network routes that minimise the impact of inaccurate network predictions whenever deviations from planned routes are detected.

The measures of merit used to determine the suitability of the proposed solution are: the algorithm's ability to minimise packet loss while improving network throughput. Packet loss will be measured by the amount of dropped packets throughout mission execution. Network throughput will be measured by the average kBps during mission execution.

6.1 Simulation scenario

A scenario was developed to measure the performance of the proposed solution. The scenario simulates the deployment of 15 remotely piloted vehicles (RPVs) that act as mobile information collection nodes and three base stations. Once deployed, the RPVs form a MANET that disseminate the information collected by the sensors. The base stations disseminate the information collected by the RPVs to a remote location via a secured wired network. A detailed description of the mobility for the different airborne platforms is illustrated in Fig. 3. Additionally, the scenario specifies the network traffic generated by each RPV as the information is collected. The network traffic for each RPV is prioritised allowing planners to specify the importance of the information collected.

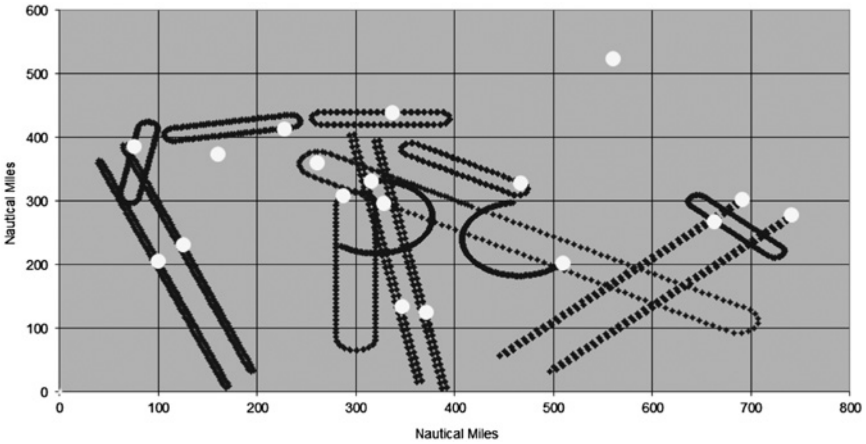


Fig. 3 *Mobility tracks for the simulation scenario*

Table 1 Distance in miles among nodes

Node_ID	Time, s	*X* position, feet	*Y* position, feet	*Z* position, feet
1	8100	174,449	709,160	6096
2	8100	436,748	769,637	6705
3	8100	577,265	777,840	6096
4	8100	788,872	609,752	6096
5	8100	1,081,194	695,136	6096
6	8100	333,360	348,673	18,288
7	8100	763,127	623,568	18,897
8	8100	494,884	467,064	3048
9	8100	851,892	550,564	3657
10	8100	872,555	1,031,564	4572
11	8100	923,453	1,021,378	5486
12	8100	972,761	1,008,174	4572
13	8100	872,555	168,532	4572
14	8100	923,453	178,718	5486
15	8100	972,761	191,922	4572
16	8100	296,320	694,500	0
17	8100	481,520	666,720	0
18	8100	1,037,120	972,300	0

6.2 NTO model implementation

This paper assumes that the mobility of the different airborne platforms was derived from information contained in the ATO. Using the NTO process we can use this information to generate a forecast of the trajectories for all participating nodes. To realise an improvement on network performance based on NTO predictions, it is necessary to discretise individual node's mobility information. This information is stored in tables with discrete time intervals, as depicted in Table 1. This technique permits the prediction of network topology throughout mission execution as identified in the NTO.

This paper assumes that all nodes are equipped with radio-frequency (RF)-based networking devices capable of adjusting the data rate to increase connectivity range. Radios with this capability implement several different modulation techniques. The radio selects a modulation scheme optimised for increased data throughput when operating in a high signal-to-noise ratio (SNR). However, a low data rate modulation is used in low SNR environments, because this type of modulation provides better performance for these conditions. Radios with automatic rate adaption select the modulation scheme that provides the optimum data throughput for a given channel condition [13]. For this specific simulation scenario, the algorithm generates links prediction based on the distance of the nodes as follows:

- If the distance is <15 miles, the algorithm predicts that there will be network link between those two nodes with a capacity of 8 MBps.
- If the distance is <25 miles, the algorithm predicts that there will be network link between those two nodes with a capacity of 4 MBps.
- If the distance is <45 miles, the algorithm predicts that there will be network link between those two nodes with a capacity of 2 MBps.
- If the distance between the nodes exceeds 45 miles, the algorithm will generate a prediction that the two nodes are out-of-range.

The pairwise distance and angle between all participating nodes is determined using the information stored in the tables that describe the position of the nodes at specific times. The distance between two nodes is computed utilising the Euclidian formula: $\sqrt{(A_x - B_x)^2 + (A_y - B_y)^2 + (A_z - B_z)^2}$. The inclination between two points can be computed using

$$\theta = \arccos\left(\frac{A_z - B_z}{\sqrt{(A_x - B_x)^2 + (A_y - B_y)^2 + (A_z - B_z)^2}}\right), \qquad (1)$$

and the azimuth can be obtained using

$$\varphi = \arctan\left(\frac{A_y - B_y}{A_x - B_x}\right). \qquad (2)$$

This paper assumes that all nodes are equipped with isotropic antennas; therefore, the inclination and azimuth between nodes are not used. However, it is possible to use the inclination and azimuth in between nodes to model the antenna radiation pattern for the different participating nodes. Using this information we can predict the network topology at a specific time as illustrated in Table 2.

6.3 Network traffic for simulation scenario

The scenario simulates an MANET that disseminates information collected from several different Intelligence, Surveillance, and Reconnaissance (ISR) sensors for further processing. Nodes 17, 18, and 19 are used as destination nodes because they simulate the base stations. The vast majority of the traffic uses is directed to a remote ground location that oversees the entire battlefield via a secured wired network connected to the base stations.

All simulations were conducted using the traffic pattern specified in Table 3. Nodes acting as traffic sources transmit 50% of the time, and are quiet the remaining 50%. All traffic sources transmit at a constant bit rate when active. Transmitting and quiet times are determined using a pseudo-random number generator (PRNG). The seed used for the PRNG corresponds to the respective experiment number. For example, experiment 18 uses a seed of 18. This procedure ensures that the same traffic pattern is generated for all different routing solutions.

6.4 Online agent to react to inaccurate NTO predictions

The algorithm implements an online agent that monitors network traffic real-time to detect network congestions caused by inaccurate

Table 2 NTO prediction for network topology at 8100 s

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
node 1		2														4		
node 2	2		4													2	4	
node 3		4		2			2										4	
node 4			2				8		8									
node 5									2									
node 6								2										
node 7			2	8					4									
node 8						2											2	
node 9				8	2		4											
node 10											8	4						2
node 11										8		8						4
node 12										4	8							8
node 13														8	4			
node 14													8		8			
node 15													4	8				
node 16	4	2															2	
node 17		4	4					2								2		
node 18										2	4	8						

Table 3 Network traffic for simulation scenario

Source	Destination	Start, s	Finish, s	Packet size, bytes	Rate, s	Priority
0	4	60	120	100	0.001	00
2	1	60	120	100	0.001	01
3	1	60	120	100	0.001	02
4	1	60	120	100	0.001	03
5	21	60	120	250	0.001	04
5	1	60	120	100	0.001	05
6	1	60	120	100	0.001	06
7	1	60	120	100	0.001	07
9	21	60	120	100	0.001	08
10	21	60	120	100	0.001	09
11	21	60	120	100	0.001	10
12	21	60	120	100	0.001	11
13	21	60	120	100	0.001	12
14	21	60	120	100	0.001	13
3	6	60	120	100	0.001	14

NTO predictions or unaccounted events. This agent is executed at predefined intervals that coincide with Kalman filter prediction rate. For example, if the Kalman filter prediction rate is 1 s ahead of time, this online algorithm will be executed one time per second. This ensures complete synchronisation of the agent throughout mission execution. The flowchart that describes the functionality of the online agent is shown in Fig. 4. The main functionality of the agent is implemented in the following five key processes: generate queue predictions, compute maximum concurrent multi-commodity flow, generate and implement routes, stop low priority flows, and re-enabled stopped flows.

6.4.1 Generate queue predictions: The agent uses a Kalman filter to monitor the queue that holds incoming packets. The Kalman filter samples the queue size of the incoming link at predetermined intervals to generate estimates. The size of the interval equals the prediction size. For instance, if we sample the queue every 5 s the Kalman filter generates a prediction of the queue size 5 s into the future.

6.4.2 Maximum concurrent multi-commodity flow: The online agent uses the maximum concurrent multi-commodity flow algorithm previously described, to compute new routes when congestion is detected. Network congestion is detected before queues are overflowed and nodes are forced to drop incoming traffic by monitoring the predictions generated by the Kalman filter. The agent needs to know the entire network topology and network flows before running the algorithm. It is possible to disseminate this information throughout the network within a second in an MANET consisting of 20 active nodes. This algorithm is processor intensive; however, it is possible to compute solution for a 20 nodes MANET with 40 commodities in a fraction of a second using a modern personal computer.

6.4.3 Generate and implement routes: The output of the maximum concurrent multi-commodity flow algorithm provides the information necessary to generate optimum routes. The new routes have to be disseminated throughout the entire MANET before they become effective. It is possible to disseminate the new routing information in an MANET consisting of 20 nodes in a fraction of a second. There will be times where the flow demands exceed the capacity of the MANET. However, the routes generated by the multi-commodity flow algorithm will be optimum even under these conditions.

6.4.4 Stop low priority flows: The online agent will detect edge congestion via Kalman filter predictions. The threshold used to determine edge congestion is a Kalman filter prediction exceeding 60% queue capacity. Previous research has demonstrated that

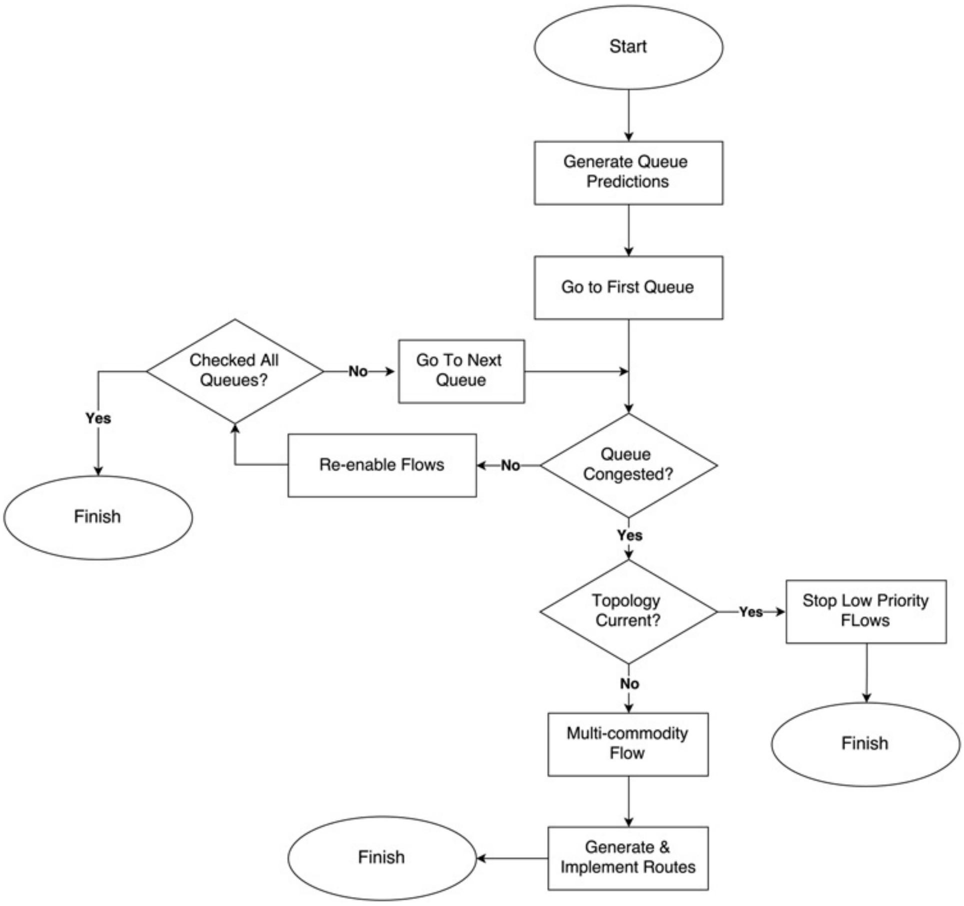


Fig. 4 *Online agent to react to inaccurate NTO predictions*

Kalman filters can be used effectively to determine a condition where flow demands exceed actual network capacity [12]. A conservative approach to this scenario is to stop the low-priority flows associated with this link. This will ensure that high-priority traffic is delivered to its destination.

6.4.5 Re-enable stopped flows: The algorithm utilises Kalman filter predictions to determine if a link is saturated. A link is considered underutilised if the prediction for the queue size falls below 40%. All flows that were stopped to prevent network congestion are reactivated in this process.

7 Results

This section describes the experiments performed to demonstrate an implementation of the NTO concept to improve an MANET's performance in a military environment. Additional experiments were conducted to demonstrate the use of Kalman filters to predict network congestion. These predictions show how to establish network controls that minimise any adverse performance impact on the network due to edge overutilisation.

7.1 Network performance using the NTO concept

Two sets of experiments were conducted to contrast the performance of the proposed solution against traditional routing mechanisms. The first sets of experiments were conducted using Ad hoc On-Demand Distance Vector (AODV) routing protocol routing protocol. The performance of the AODV routing protocol was determined by running the same scenario 30 times in accordance with the central limit theorem. A PRNG was used to determine the transmission intervals. As a result, all traffic sources were transmitting 50% of the time on average. The packets sizes and the sending rate remained constant for all transmitting sources.

The second sets of simulations were conducted using the NTO concept to generate optimum routes. All routes used for these experiments were pre-computed using the maximum concurrent multi-commodity flow. All other simulation parameters remained unchanged from the first set of experiments.

Simulation results demonstrated that network routes obtained using the multi-commodity flow algorithm outperforms traditional routing solutions such as AODV as illustrated in Fig. 5. Simulations that incorporated NTO optimisations dropped 12,845 packets per second on average. Simulations using AODV as the routing solution dropped 59,841 packets per second on average. It is important to note that the performance of both routing solutions is identical for the last 2 s of simulations due to the fact that there is only one possible route for all active traffic. Consequently, AODV- and NTO-based solutions generate the same routes for the last 2 s. The throughput for simulations that incorporated NTO optimisations was 843 kBps on average. Simulations based on AODV routing solutions averaged 758 kBps. These results are illustrated in Fig. 6.

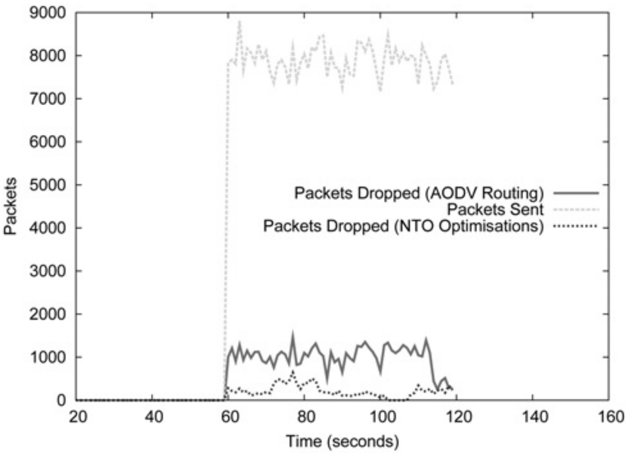


Fig. 5 *Packets sent against packets dropped per second – average of 30 simulations*

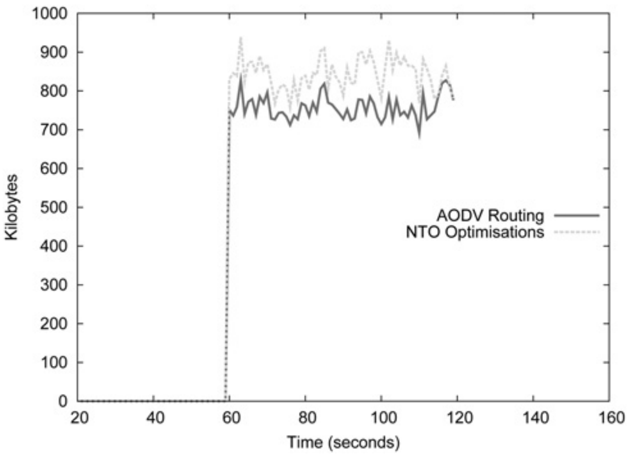


Fig. 6 *Network throughput – average of 30 simulations*

7.2 Deviations from predicted routes

Routes generated using the maximum multi-commodity flow algorithm are highly optimised for a given topology. These routes are static and they do not have mechanism to adjust for changes in topology. It is possible to end up with routes that are unfeasible by just changing the quality of one of the MANET's link. One of the most challenging aspects of designing a new MANET routing solution is to incorporate network controls to adjust the routing solution to sudden topology changes.

The first two sets of simulations computed the quality of the network edges based on the position of all participating nodes. Each node's position was computed by interpolating perfect trajectories. Such levels of accuracy cannot normally be achieved outside simulation environment. Two additional sets of simulations were conducted that incorporate an induced error in each node's position. The error is calculated using the following formula

$$\text{sim\_pos} = (\text{actual\_pos} \cdot ((\text{PRNG}() \quad \bmod 15) - 7)) \tag{3}$$

The next two simulations incorporate the node positioning error model in (1). The induced error is non-cumulative and will not alter a node's overall trajectory. All other simulations parameters remained unchanged from the first two sets of experiments. This error model simulates deviations from predetermined routes as well as variations on signal strength for RF transmission.

The performance of these two sets of simulations degraded when compared with the first two sets. This change in performance is expected, because the topology changes are more frequent. The optimised routes computed using the maximum concurrent multi-commodity flow algorithm were not adjusted to reflect the topology changes induced by the node's position error.

The simulations that incorporated NTO optimisations dropped 15,217 packets per second on average. Simulations that used a routing solution based on AODV protocol dropped 68,104 packets per second on average. The optimised routing solution outperforms the AODV-based solution significantly during the last 2 s of simulation time in contrast to the results shown in Fig. 5. This is the case because AODV saturates few network edges by dynamically adjusting network routes to use shortest path between source and destination. These results are illustrated in Fig. 7.

The 95% confidence intervals for the simulations are illustrated in Fig. 8. Visual inspection shows that the confidence intervals do not overlap, demonstrating that the two means are statistically different at the 95% confidence level. Routing solutions that implement

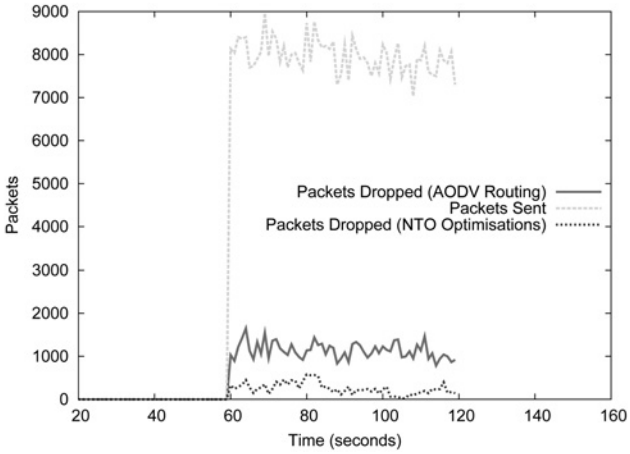


Fig. 7 *Packets sent against packets dropped per second with topology errors – average of 30 simulations*

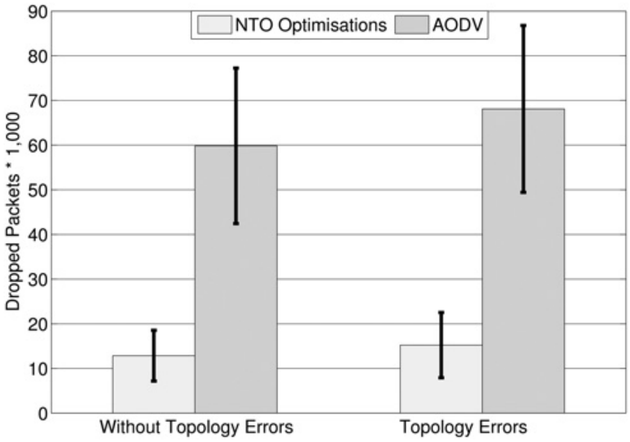


Fig. 8 *Dropped packets – 95% confidence interval comparison with and without topology errors*

NTO optimisation minimise dropped packets because it uses a maximum concurrent multi-commodity flow paradigm to model the network traffic and topology. This technique minimises edge overutilisation. In contrast, AODV routing protocol uses the minimum cost path between any given source–destination pair. This routing solution saturates network edges that are part of the minimum cost path for multiple source–destination pairs.

7.3 Network performance using an online agent

The final set of simulations utilises a Kalman filter-based online agent to detect deviations from routes computed using the NTO concept and refreshes the network topology accordingly. Nodes detect deviations from NTO preplanned routes whenever the plan requires forwarding traffic via non-existing network edges or when edge saturations occur. Detection of network edge saturation can take up to 1 s under this condition because all nodes in the network path use queues to buffer excess traffic.

The network topology is refreshed using traditional node discovery methods. About 180 experiments were conducted to determine the time it takes to replicate the network topology for an MANET with 20 nodes. In all cases, the MANET topology was obtained in <0.30 s after transmitting <300 packets. Table 4 shows the total time required to refresh network routes to all participating node.

The preplanned topology loaded for this set of simulations is offset by 100 min to simulate deviations from preplanned routes. This ensures that network routes preloaded to nodes are not consistent with the actual topology. The online agent detects this failed state and implements network controls to recover. The online agent detects deviations from NTO preplanned routes in the form of non-existing network edges, saturated edges, or both and computes optimum network routes in real-time. The performance of the network using the online agent is illustrated in Fig. 9. Results indicate that the online agent generates routes that provide marginally better throughput performance than AODV routing protocol. Additionally, the online agent stops low-priority traffic when network demands exceed actual capacity minimising high-priority traffic loss.

Table 4 Time required to refresh network routes

detect deviation from preplanned routes	1 s
refresh network topology	0.3 s
compute optimum network routes	0.3 s
disseminate routes to participating nodes	0.3 s
total time	~2 s

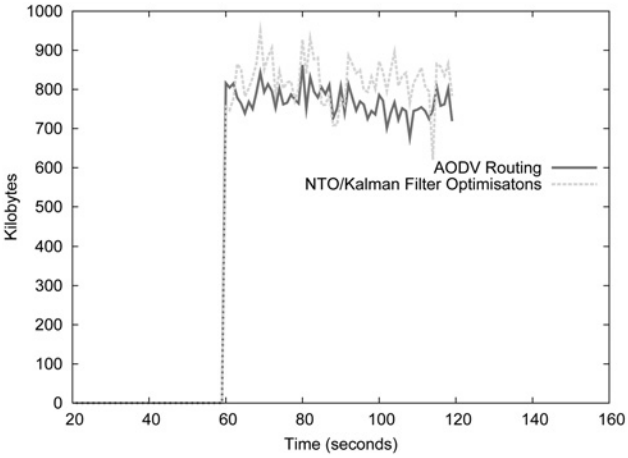


Fig. 9 *Network performance using online agent*

8 Conclusions

The objective of this paper is to present an algorithm that capitalises on the network information implied in the ATO. Using the NTO concept, this paper demonstrated that it is possible to optimise the performance of highly dynamic MANETs used to deliver critical battlefield information to the warfighter. The NTO process implements network controls to provide quality of service in highly dynamic network environments. The increased information flow can be used to achieve information superiority.

The algorithm specified in this paper computes optimum routes based on predicted network traffic and topology. Additionally, this paper provides instructions for an online agent that monitors network traffic real-time to detect and react to inaccurate network predictions minimising any adverse impact.

Though results in this paper are promising, there is much work that can be done to further refine the methods specified in this paper. The NTO concept should be evaluated using real-world data obtained from military planning documents (e.g. ATO) to quantify the differences between mission planning and actual execution.

The results illustrated in paper demonstrate better network performance when the proposed solution is implemented when compared with traditional routing protocol. However, this routing solution cannot be used for every scenario. Consideration must be given to the additional resources required to implement and manage this solution. Nodes are required to have significant computer power to implement the maximum concurrent multi-commodity flow algorithm. This paper was adapted from

Addison Betances' Master's Thesis [17] which provides additional information about this research effort.

9 Acknowledgments

The views in this document are those of the authors and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the U.S. Government.

10 References

1 Willard, D.: 'Network management architecture for the objective airborne network'. IEEE Military Communications Conf., October 2004, vol. 1, pp. 388–393

2 Naveen, K., Karuppanan, K.: 'Mobile cluster assisted routing for urban vanet'. Int. Conf. on Recent Trends in Information Technology (ICRTIT), June 2011, pp. 296–300

3 Pathak, S., Jain, S.: 'A survey: on unicast routing protocols for mobile *ad hoc* network', *Int. J. Emerging Technol. Adv. Eng.*, 2013, **3**, (1), pp. 2250–2459

4 Tiwari, A.: 'Context aware routing and management architecture (carma) for airborne networks'. Technical report, Utopia Compression, 2010

5 JCS: 'Joint publication 3–30. Command and control for joint air operations'. Technical report, Joint Chiefs of Staff, 2014

6 DoD: 'Department of defense directive (DODD) 8100.1'. Technical report, Department of Defense, 2002

7 Sang, A., qi Li, S.: 'A predictability analysis of network traffic'. INFOCOM 2000. Proc. IEEE 19th Annual Joint Conf. of the Computer and Communications Societies, 2000, vol. 1, pp. 342–351

8 Stuckey, N., Vasquez, J., Graham, S., *et al.*: 'Stochastic control of computer networks', *IET Control Theory Appl.*, 2012, **6**, (3), pp. 403–411

9 Haught, J., Hopkinson, K., Hemmes, J.: 'The modeling and simulation of a network controller based on a router queue-size predictions', *J. Def. Model. Simul., Appl. Methodol. Technol.*, 2013, **10**, (2), pp. 105–115

10 Murray, M., Quigley, J.: 'Automatically generating a distributed 3D battlespace using USMTF and XML-MTF air tasking order, extensible markup language (XML) and virtual reality modeling language (VRML)'. Master's thesis, Naval Postgraduate School, 2000

11 Wang, Z., Crowcroft, J.: 'Analysis of shortest-path routing algorithms in a dynamic network environment', *ACM SIGCOMM Comput. Commun. Rev.*, 1992, **22**, (2), p. 71

12 Scheideler, C.: 'Johns Hopkins University CS 600.348-Lecture Notes in Theory of Network Communication: Network Flows III – Multicommodity Flows, Fall 2003. Available at http://www.cs.jhu.edu/_scheideler/courses/600.348F03/

13 Even, S., Itai, A., Shamir, A.: 'On the complexity of time table and multi-commodity flow problems'. 16th Annual Symp. on Foundations of Computer Science, 1975, October 1975, pp. 184–193

14 Goldberg, A., Oldham, J., Plotkin, S., Stein, C.: 'An implementation of a combinatorial approximation algorithm for minimum cost multicommodity flow', in Bixby, R.E., Boyd, E.A., Rios-Mercado, R.Z. (Eds.): 'Integer programming and Combinatorial Optimization' (Springer, vol. 1412, 1998), pp. 338–352

15 Kolliopoulos, S.G.: 'Edge-disjoint paths and unsplittable flow' in Gonzalez, T. (Ed.), 'Handbook of approximation algorithms and metaheuristics' (Chapman & Hall, Boca Raton, FL, USA, 2007), pp. 7–11, ch. 57

16 Fleischer, L.: 'Approximating fractional multicommodity flow independent of the number of commodities', *SIAM J. Discrete Math.*, 2000, **13**, (4), p. 505

17 Betances, J.: 'Context aware routing management architecture for airborne networks', Master's thesis, Air Force Institute of Technology, 2012