Testing Multi-Services Mobile Adhoc Networks Effectively

S.S.Kamal, J.D. Aishman
Science Applications International Corp. [SAIC]
1710 SAIC Drive
McLane, VA 22102

ABSTRACT

This paper's arguments start with the belief that "simple is hard to implement". The literature is packed with concepts that propose mobile platforms (ground or airborne) move freely and unfettered, with little or no hierarchy, to execute their missions. This is notable in defense and in emergency response or disaster relief applications where too rigid a network architecture will cripple operations and be vulnerable to single-points-of-failure and bottleneck congestion. This paper examines how such "free" networks should be tested to meet their intended requirements. With little or no defined topology and rules, how are performance metrics like throughput, latency, message completion rates, Quality of Service (QoS) measured? An argument is presented for how network products cannot & should not be viewed simply as "smart" or "software defined" radios. Ultimately, the paper's intent is to propose a mindset and discipline by which network products in such environments should be tested very differently from how radios are tested today. It discusses performance metrics for MANETs, test tools for MANETs, and programmatics for MANETs.

1. INTRODUCTION

The paper focuses on operational environments where rapid reaction, agility, timely information and situational awareness are critical to allowing such missions to either achieve battlefield superiority or overcome emergency/disaster situations with lower losses and/or fewer assets deployed. The underlying premise in these mobile ad hoc networks (MANETs) is that connectivity is maintained despite routing paths varying due to radios roaming free and radio frequency (RF) links being unstable.

MANETs are often the foundation of even more sophisticated mobile networks. E.g. ***cognizant networks***, whose devices develop awareness of their environment_ some dynamically detect and use vacant radio frequency spectrum (*Dynamic Spectrum Allocation*). At their most sophisticated, MANETs do not merely react to their rapidly changing environment, but can be designed to participate in deciding which information is required where…and when. Such an environment is often referred to as being **Netcentric**: executing the mission on the network, not merely via the network.

This paper discusses a growing problem in delivering such networks. Simply stated, devices for this new class of wireless networks are not being tested or procured effectively. This paper will discuss why this is happening and propose remedies to the stated problem. At stake are millions of dollars spent developing and deploying this new class of networks.

Mobile Ad hoc Networks (MANETs) are self-configuring networks of mobile devices connected by wireless links. MANET devices are free to move independently in any direction, and will therefore change links to other devices frequently. Each device must also forward traffic unrelated to its own use, and therefore be **a router**. Such networks may operate by themselves or may be connected to external networks to form a larger fabric [1]. MANETs usually have a 'routing' networking environment on top of a Link Layer ad hoc network. Table 1 provides some sense of the scope of ongoing research conducted to design routing protocols for MANETs.

1. Pro-active (table-driven) routing
2. Reactive (on-demand) routing
3. Flow-oriented routing
4. Adaptive (situation-aware) routing
5. Hybrid (both pro-active and reactive) routing
6. Hierarchical routing protocols
7. Host Specific Routing protocols
8. Geographical routing protocols
9. Power-aware routing protocols
10. Multicast routing
11. Geographical multicast protocols (Geocasting)

Table 1. Classes of Ad hoc Routing Protocols

Ad hoc architectures have been studied since the 1950s and with recent technological advances witnessed resurgence in the late 1990s. For most of that time focus has been on applications where miniature radio-equipped sensor devices and robots move randomly while maintaining communications. The more recently revived interest has been fueled by:

1. The explosion of consumer mobile devices and the inability of rigid network structures to support their equally exploding multimedia applications;
2. The need for military, security, public safety and disaster-relief operations to adopt looser networking

architectures to execute increasingly complex missions with fewer assets[3]; and

3. The advances in the hardware & software components of software-definable radios (SDRs) [2] [3] [4].

In short, emerging new user applications (multi-services) intersect with emerging technologies to collectively drive this revived interest in network architectures that are not hierarchical or rigidly preconfigured. The participating devices in these applications move unpredictably, and structured architectures have proven incapable of efficiently supporting their growing needs[1].

From this point forward, it will be useful to agree (or debate) the following view: ***This type of network should neither be viewed as (i) a mobile wireless network trying to behave like a wire line Internet Protocol (IP); nor (ii) as a fixed IP network with an RF radio in front of it. Both are grossly inaccurate and have contributed to primitive products and inadequate test strategies in MANETs.***

It is important to confront this tenet because many efforts incorrectly define a MANET "problem" as:

- How do the devices acquire & maintain routing information bi-directionally with their neighbors, so that robust IP connections are maintained in the network? Or
- How can we isolate the IP network from the MANET?

This paper suggests that the problem needs to be re-phrased: How do we understand how this MANET performs so that we can assess (i) how well our applications will perform? And (ii) what Concepts of Operations (CONOPS) do we need to adjust/modify to recognize that the applications ***are not running on a wire line IP network*?**

Can we realistically specify network metrics like scalability, throughput, latency, message completion rate and QoS for MANETs[2]? Yes, but not in the traditional manner [7] [10].

To some this may seem like the network is wagging the user. This paper argues that traditional 'wire line' IP network design principles and models cannot describe MANET behavior. Forcing such principles on a MANET has cost some programs hundreds of millions of dollars and years in delays. We need to understand how MANETs work and how to measure their performance, so that we can deploy them effectively.

2. METRICS for MANETs

How then should traditional metrics like scalability, throughput, latency, message completion rate and QoS be tested in this class of networks?

SCALABILITY: A metric specifying a network's ability to handle an increasing amount of work. This can be by increasing the number of participating devices, increasing the volume of traffic it supports, or other increases in scale and scope. Neither the number of participating nodes, nor their density (how close they move around each other) is sufficient alone to define scalability[3]. Why?

- **Traffic Density.** Some bandwidth intensive applications can cripple even a 3-node network. Figure 1.
- **Node Density.** Equally, testing a 1000-node network with minimal traffic and minimal mobility doesn't prove much unless that is what its function will be (more typical of sensor networks). Figure 1.

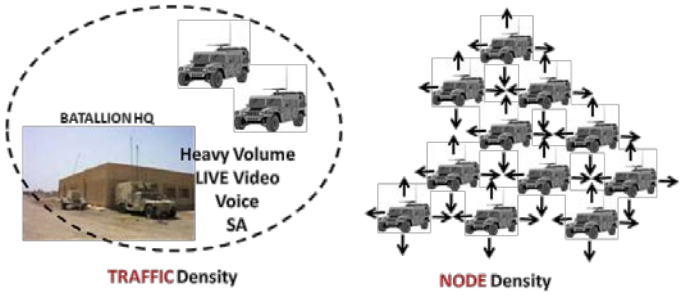

Figure 1. Traffic Density ≠ Node Density

- **Mobility.** Neither do network tests with slow moving vehicles test the devices' need to rapidly discover new routes and recover from severed ones when radio links change rapidly. This becomes more complex if the network has a mix of mobile devices carried by dismounted soldiers, vehicular users and airborne platforms_ all moving at different speeds.
- **RF Conditions.** Testing in flat terrain overlooks severe fragmentation experienced by networks when operating in dense foliage or urban city environments; and cannot exercise a network's self-healing/rejoin capabilities that are vital in critical missions. Figure 2.
- **Deployment Topology.** Testing only dense networks may reveal the behavior under conditions where radio devices may interfere with each other; but overlooks whether sparse, spread-out networks can maintain route-

[1] *Efficiency* here refers to the provisioning of both equipment and wireless spectrum to meet those needs.

[2] These metrics are also called Key Performance Parameters (KPPs) or Netcentric Key Performance Parameters (NRKPPs).

[3] Necessary, but not sufficient [6] [8] [9].

and-relay capabilities under stressed traffic volumes and mobility. Figure 3.

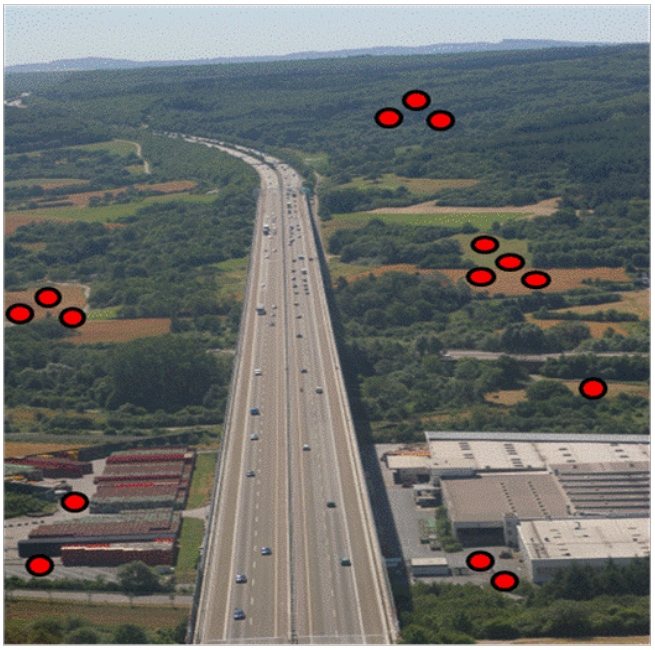

Figure 2. DENSE Terrain Cut-off of Connectivity

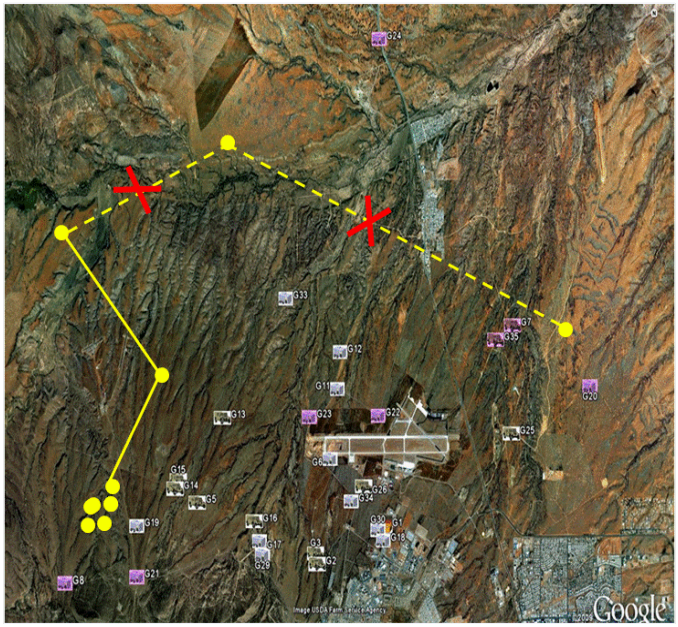

Figure 3. SPARSE Terrain Cut-off of Connectivity

In reality,

SCALABILITY = function Fs{N, D_N, T_{FC}, M, T_{RR}, …..}

Where:
N: Number of Nodes
D_N: Node density
T_{FC}: Traffic profile {volume, directivity}
M: Mobility of participating devices
T_{RR}: Terrain profile

To make things even more challenging, all the variables of scalability change with time. Scalability in a MANET is itself a compound function of time-variable parameters. i.e. N is N(t) as nodes leave and join the network, density D_N is D_N (t) as mobile nodes move closer and further apart, T_{FC} is T_{FC}(t) as the RF channel at any instant in time varies; etc. Consequently, no one scenario or test case is sufficient to characterize the network's performance. Understanding this fact will influence how testing for Key Performance Parameters (KPPs) can be adequately conducted. It also should avoid the inclination for the buyer to specify an absolute network size, or for vendors to craft 3 or 4 custom tailored scenarios under which KPPs can be met. This is discussed in Section 3.

THROUGHPUT: Used to reflect how efficiently the network can transport traffic from source(s) to destinations(s) devices. It is a measure of how well capacity, bandwidth and spectrum allocated to the network is being used to support its users. In a MANET this fluctuates with time [6]. Why? Because the wireless connections transporting that traffic change as the network moves. See figure 4.

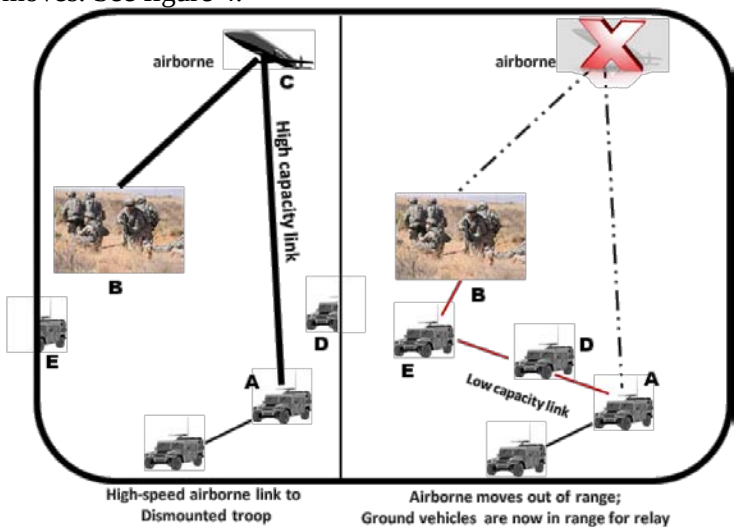

Figure 4. Shifting Network Conditions

- Devices change speed and direction.
- The relay capability of devices between source & destination changes as they move [11].
- The applications using the network change the direction and volume of the traffic carried; and therefore the routes taken from source to destination.
- The RF channel characteristics change with time and mobility, forcing routes to change on-the-fly.

Getting from A to B via node C in 1 instant is not the same as going via nodes D and E in the next instant. i.e.

THROUGHPUT= function Fμ{N, D_N, T_{FC}, M, T_{RR}, ….}

LATENCY: A measure of the delay incurred by the information as it traverses the network. This metric should be specified such that (a) meaningful information arrives in a timely manner and (b) interactive communications do not become crippled by the stuttering of the information exchange between source(s) and destination(s). Voice, data, interactive applications… all have a mix of precise scientific and empirical rule-of-thumb values for acceptable delays between source & destination. However, in a MANET routes and route conditions (capacities) vary constantly. Therefore

- Fluctuating node density and instantaneous positions of the network devices affect instantaneous routes selected, and therefore overall distances.
- Traffic volume & direction affect congestion levels at (relay) nodes. This affects instantaneous routing, as traffic attempts to route around congested relay nodes.
- Different applications at different nodes are processed at different speeds and this impacts lower priority traffic relay and delivery.

Therefore, it is no surprise that here too

$$\textbf{LATENCY = function } F_L\{N, D_N, T_{FC}, M, T_{RR}, \ldots.\}$$

MESSAGE COMPLETION RATE (MCR): This is a metric used to reflect a network's ability to allow applications that use it to finish what they started. Applications inject traffic into the network that cannot be lost or destroyed by congestion or excessive delays that trigger timeouts. Otherwise, packets get dropped; lost forever. MCR itself is influenced by a network's throughput & latency; and by node mobility, traffic profiles and RF channel variations. But also by design factors like buffer management rules, rerouting algorithms and QoS policies.

$$\textbf{MCR = function } F_{MCR}\{N, D_N, T_{FC}, M, T_{RR}, \ldots.\}$$

QUALITY of SERVICE (QoS): This is discussed last for a reason. By now the discussions of constant flux in a MANET's environment should make it clear that traditional QoS metrics like committed bit rate and priority services cannot apply as they do in wire line IP networks. In fact, with no central control, preempting services because an emergency connection is needed is very complex (but not impossible) to implement.[6]. i.e.

$$\textbf{QoS = function } F_{QoS}\{N, D_N, T_{FC}, M, T_{RR}, \ldots.\}$$

In reality, none of the above metrics can be measured effectively using test plans we modify from those of traditional IP networks, or wireless devices. Figure 5 graphically illustrates the compound dependencies that represent the relational functions given above. As a networking and T&E (test & evaluation) community, we have yet to assemble the effective test environments for MANETs. This has largely been due to misunderstanding the nuances of MANETs, and a gross underestimation of the eventual costs of deploying them.

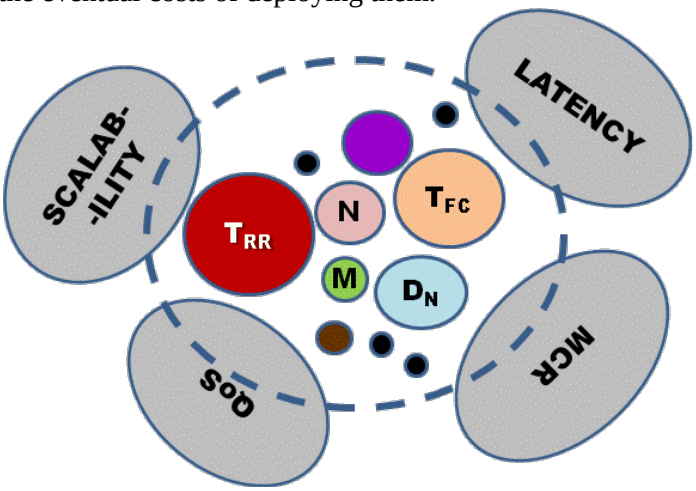

Figure 5. Compound Functions of MANET KPPs

In MANETs these KPPs are determined largely (not entirely) by the performance of Layers 2 & 3 of the protocol stack: The Media Access Control (MAC) layer and the Mobile Intranet (MI) layer[4].

1. ***The Media Access Control layer:*** This governs how MANET devices share the available network capacity (or spectrum). How well the capacity allocation algorithms react to the MANET variations discussed above, will also determine the performance of throughput, latency, QoS etc. If mobile devices in MANETs cannot share the airwaves efficiently, performance will suffer.
2. ***The Adaptive Link Control:*** These are algorithms are implemented in the MAC layer to adjust a device's transmissions (power, modulation, diversity, etc.) as they move in and around each other. Do they mitigate interference with each other or add to it because they can't keep up with the rapid changes in the RF environment? How well the algorithms perform also impact the network KPPs.
3. ***The Mobile Intranet Layer:*** This is where the devices "discover" their neighbors and form routing connections to each other. How this layer is designed and how rapidly it reacts to the MANET variations in routes affect performance. Moreover, how much

[4] It is in these 2 layers where some of the cognizant radio features are also implemented. E.g. dynamic spectrum assignment, and geo-location awareness.

control overhead it injects into the network traffic to do its work will determine how much of the network capacity it consumes; thereby affecting overall network throughput, latency, MCR etc…

Mobility, direction, connectivity, instantaneous relative locations of devices, the applications they run… all influence how a MANET performs. They all vary by the minute, if not by fractions of a second.

3. TEST TOOLS for MANETs

Such complex interdependency of MANET factors may appear to make testing hopeless. It is important not to lose sight of the big picture. Too often complex concepts are allowed to bleed into complex program processes and complex test strategies. Suddenly acquisition projects are drowned in circuitous labyrinths of technical and program reviews_ all fabricated to essentially downgrade the original requirements and "get what you get". Costly initiatives are launched to measure headers on packets and elaborate test instrumentation is unfolded to capture every heartbeat of the MANET. Is this necessary? For product development? Yes. For system performance test and deployment? No. So we need to step back and regain focus of our key objectives.

The toolsets discussed in this section aim to provide a Test & Evaluation strategy for MANETs that should be adopted from program inception. The authors have witnessed years and millions of dollars wasted in programs, as they try to recover from not doing so.

Toolset #1: Define the requirements effectively. i.e. define them at the applications level; what do the ultimate users of this network need to do their job? E.g.

Traffic: *The network shall be capable of simultaneously supporting 36 voice connections, 28 internet connections, 3 video streaming connections, and 2 file transfers of documents NLT 8MB each in NMT 45 seconds*[5].

Mobility: *Devices in the network will be moving at speeds of NMT 50kmph (ground), 20knots (maritime) and 180knots (airborne); in any mix of variable speeds and directions*[6].

Density: *The network shall operate within an AOR of NMT 400 sq.miles; regardless of where the nodes are with respect to each other.*

Terrain: *The network shall operate in the frequency range A→B MHz, in all types of outdoor terrain, foliage and urban environments.*

Spectrum: *For the traffic volume specified, the network shall use NMT 5MHz of contiguous or non-contiguous spectrum to execute its missions; anywhere in the A→B MHz range.*

Quality of Service: *The network shall allow traffic to be classified in up to 8 levels of priorities, and always ensure that the available network capacity is serving higher priority traffic first.*

Other: S*pecific threat survivability requirements, security requirements (incl. Low Probability of Interception- LPI, Low Probability of detection- LPD and access control), and net management requirements should also be specified at the user level: What needs to be done; not how it is done.*

What is equally critical for testing is that it be clear that these requirements are required simultaneously; not to be tested individually. KPPs are derived from this level of requirements. Resist specifying **how the network meets its KPPs**. Too often contracts specify technical requirements they shouldn't. From a testing perspective this has wasted time and money verifying necessary but severely insufficient performance. Moral of the story: Stick with defining clear operational requirements.

Finally, refrain from using CONOPS (Concepts of Operation) as "requirements". They are vital for running field tests and simulation models_ but only as practical situations which the MANET must support. Verification and Acceptance testing of the MANET should seek to "break" the network, to identify its performance limits. Write your contracts that way.

Toolset #2: Developers and Operators must invest in developing both Behavioral Models[7] (BMs) and Shared Code Models[8] (SCMs) for the network.

The scarcity of recognized and proven analytical models for MANETs has crippled many programs from meeting their requirements. Modeling & Simulation (M&S) will prove to be one of the most vital toolsets we need to plan and execute missions on the MANET.

[5] NMT: No more than; NLT: No less than.
[6] 1 knot = 1 nautical mph≈1.852 km/h or 1.151 mph

[7] BMs: Algorithmic models that represent the proposed design of the devices and their mobile routing implementation.
[8] SCMs are simulation environments in which the actual mobile device software is inserted in a simulated environment.

The bottom line in realizing the role of M&S is to understand that running ***some scenarios*** in the lab/field will not tell you what the MANET can support for your next mission. A close re-read of Section 2 should convince us of that. Too many things change the outcome. Therefore…..

- **Developers & Operators:** During development, BMs guide the design of the MANET algorithms and protocols. But after development BMs can guide mission planners in defining what network assets can execute the missions envisioned.
- **Developers:** During development SCMs assist code debug and design verification. During developmental testing and system performance testing SCMs become a component of a Hardware-In-The-Loop (HITL) environment for testing scalability [6], [10].
- **Developers & Operators:** HITL test environments are essential when providing a sufficient number of actual radio devices for testing scalability and network stressing is cost prohibitive. One program learned the hard way, spending tens of millions of dollars scrambling to assemble actual radio devices, and conducting months of field testing_ yet failing to thoroughly cover all the KPPs.

Both BMs & SCMs can also play an important role in characterizing the MANET's interface to its external environment. Rarely does the MANET exist in a vacuum. In Defense applications, MANETs are used to extend the Global Information Grid (GIG) to the tactical battlefield [4]. In Emergency & Disaster/Relief applications they interface to backbone and rear command centers across a county, state, country, continent or globally. Models should be used to give early warnings of design concepts that do not recognize the bigger picture. Models will augment lab & field testing (see Toolset#3).

A strong distaste for simulation lingers with those in the Defense sector who have experienced Modeling & Simulation efforts that wound up costing more than the program they were intended to assist; and still failed to be useful. This is the wrong lesson to learn. Effective use of models is essential to keeping MANET programs on track, and allowing operators/ planners to conduct endless "what if" scenarios and refine their Tactics, Techniques & Procedures (TTPs) to effectively use the network long after it has been delivered and paid for.

Toolset#3: Recursive Lab & Field testing.

Figure 6 depicts their respective roles and how they fit together with the other tools. Together, the toolkit embodies a "***test a little, verify, fix, test some more***" approach[9] [10]. All test plans should demand that the network devices be stressed and the network "broken". This test data is invaluable in characterizing the network's system performance limits. Despite their good intent, too often contracts dissuade suppliers from documenting the limits of their devices' performance. Meeting minimum performance levels is not a sound foundation for deploying complex networks such as MANETs.

Toolset#3's value lies in the duality of lab & field testing. But let's examine each separately.

Toolset#3a: Lab Testing. Should always be additive and recursive, not merely regressive: No linear test strategy should be adopted where individual design features are checked in isolation. Concepts of testing hundreds of devices in the lab should be abandoned.

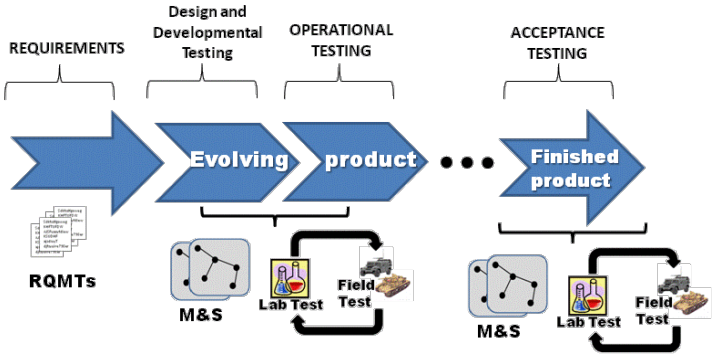

Figure 6. The Proposed Toolkit

Instead, lab tests should be used to verify and validate the BMs and SCMs [MIL-STD 3022], and then used, along with HITL test environments, to address large scale network performance [9], [10]. When possible, some level of testing should be performed by independent parties; parties other than those who supply the MANET products.

Toolset#3b: Field Tests. Save your money. They should never be used to compensate for lack (or poor use) of toolsets#2 and #3a. They are very costly and cannot realistically test too many variations or mission threads. Field tests≠ Demonstrations. For MANETs they can be confined to (a) testing the implementation of the devices' physical layer performance with their RF components (antennas, amplifiers, filters, etc.); and (b) operational tests examining the network's performance in the hands of its actual users. Most other testing can (should) be done at far less cost with channel emulators and well-designed HITL environments.

[9] This approach is being adopted by modern product development processes, ranging from Agile Computing to Xtreme Design.

Toolset#4: Developers: BITs & BIDs.

The cost savings accrued by requiring the mobile devices' software design to include Built-in-Tests (BITs) and Built-in Diagnostics (BIDs) cannot be overstated. This paper's authors have repeatedly witnessed programs waste months and millions of dollars outfitting primitive breakout boxes and poring over millions of lines of test data; chasing design flaws that could have been isolated by embedding self-tests into the software's protocol layers. It is also important to recognize that today's software-defined networking devices can (should) be designed to meet their system performance metrics while their code includes the built-in test & diagnostics required. Time and again these tools have paid for themselves by reducing the cost of instrumentation during lab and field tests.

Together, the toolsets discussed here sound obvious; perhaps textbook-like. They are largely ignored when placing contracts for networks like these. Yet they help define a clear acquisition structure, keep it on track with respect to its performance objectives, and tightly manage the risk of mushrooming budgets and runaway schedules.

The next section focuses on how the acquisition strategy itself can assist or hinder the testing of complex MANETs.

4. PROGRAMMATICs for MANETs

The role that the acquisition strategy plays in testing this class of complex networks may not be obvious… but is critical. The recent revision of the Defense Acquisition Guide (DAG) was partly influenced by the recognition that complex software-driven devices could no longer be procured using antiquated rules. Consequently, specific programmatic tenets need to be understood and adopted:

1. The contract itself cannot be linear. i.e. milestones cannot segment requirements and then proceed to prove compliance with each one in sequence. Algorithms in Layers 1-3 of MANET protocols cannot be verified in isolation. Toolset#1 advises that all requirements defined be tested simultaneously. Incrementally, yes; but also simultaneously before any final system sell-off is attempted.
2. No Critical Design Review (CDR) of a MANET design can be considered complete without Behavioral Models verifying the KPPs. Too often programs result in vendors providing deliverables [Contract Data Requirements List or "CDRL" documents] that merely regurgitate user requirements and propose paper designs. Thanks to the requirements traceability of Dynamic Object-Oriented Requirements System (DOORS), every requirement seems covered, with virtually no modeling, lab or field testing to verify if the paper designs meet KPPs and requirements. Terms like "*preliminary CDR*", "*interim CDR*" and "*CDR Part-1*" are legitimate artifacts if they help guide incremental and agile development; but not as means of segmenting requirements and verifying them in isolation. Complex network architectures cannot be delivered successfully by "checking boxes".
3. No Formal Qualification Test (FQT), Formal Acceptance Test (FAT) or any "final acceptance testing" should be conducted before (a) all models (BMs & SCMs) have been verified and validated, and (b) the entire set of system performance limitations (the limits of what the designed network can and cannot do) has been compiled and reviewed. Such outer limits must be for all KPPs simultaneously.
4. There should be no contemplation of assessing system level performance of MANET devices without the Network manager(s) being in place. Concurrent stressing of both Network Manager(s) and network is vital to ensuring rugged performance in rugged environments.
5. Defining CONOPS and Information Exchange Requirements (IERs) is a legitimate way to define test scenarios for the MANET. However, they are necessary but insufficient for system acceptance or sell-off. Why? They define what the network must do to support missions ***how they are conducted today***. Netcentric environments are supposed to challenge these very procedures, giving rise to CONOPs that exploit netcentric awareness to do more with less. Today's CONOPS may ignore advancements in technologies that suggest new capabilities and therefore new ways to exploit them.

CONCLUDING REMARKS

Let us end with a reality check: A few years ago a Government Accounting Office (GAO) report noted that: "*DOD does not have an effective process for testing or certifying newly developed C4I systems…*" [5]. The report discussed many reasons for this, but the one most relevant to this paper is that the defense community does not yet understand how to test emerging and complex military networks being procured. This should serve as warning and provide valuable "lessons learned" to the public safety and emergency response sectors.

This paper attempted to deconstruct the problem for MANETs and netcentric environments. Clear requirements, essential toolsets and disciplined programmatics are all components of a proven test strategy that mitigates this problem. It sounds simple; but simple is hard to implement.

REFERENCES

[1] Wikipedia.http://en.wikipedia.org/wiki/Mobile_ad-hoc_network#Further_reading

[2] "*MANET Architecture Design Principles: The Protocols or the Applications*?" Dr.S.S.Kamal. To be published.

[3] "*The Joint Tactical Radio System (JTRS) and the Army's Future Combat System (FCS): Issues for Congress*". CRS Report For Congress. A. Feickert. 17 November 2005.

[4] Homepage of the JTRS Joint Program Office: http://jpeojtrs.mil/

[5] "*Joint Military Operations: Weakness in DOD's Process for Certifying C4I Systems' Interoperability*". Letter Report 3/13/98, GAO/NSIAD-98-73.

[6] "*Scenario-based Performance Analysis of Routing Protocols for Mobile Ad-hoc Networks*," P. Johansson et al., Proc. Mobicom '99, pp. 195-206.

[7] "*MANET Simulation Studies: The Incredibles*," S. Kurkowski, et al., ACM SIGMOBILE Mobile Computing and Communication Review, Vol. 9, Issue 4 (October 2005), pp. 50-61.

[8] "*Scalable routing protocols for mobile ad hoc networks*" X. Hong et al., *IEEE Network* , Vol. 16, No. 4 (July-Aug. 2002), pp. 11 -21.

[9] "*On the Scalability of Ad Hoc Networks*" B.-J. Kwak, et al., IEEE Communications Letters, vol. 8, pp. 503-505. (August 2004)

[10] "*Quantitative Lessons From a Full-Scale Multi-Hop Wireless Ad Hoc Network Testbed*,"D. A. Maltz, J. Broch, and D. B. Johnson, Proc. IEEE Wireless Communications and Networking Conference, September 2000.

[11] *"Making Ad Hoc Networks Density Adaptive"* R. Ramanathan, Proc. Milcom 2001, pp. 957-961.