

MOBILE AD HOC NETWORK EMULATION ENVIRONMENT

Natalie Ivanic*, Brian Rivera*, Brian Adamson+

U.S. Army Research Laboratory*
U.S. Naval Research Laboratory+

ABSTRACT

To support research in wireless mobile ad hoc networks, the U.S. Army Research Laboratory (ARL) and Naval Research Laboratory (NRL) have worked jointly to establish an advanced emulation environment. A key component is a Mobile Ad-Hoc Network emulation testbed where algorithms and applications can be subjected to representative wireless network conditions. The testbed is based on a version of the "MANE" (Mobile Ad hoc Network Emulator) software originally developed by the Naval Research Laboratory (NRL). Consulting & Engineering Next Generation Networks (CENGEN) has enhanced MANE by implementing a modular, extensible architecture which allows advanced modeling methods and computing technologies to be incorporated into the environment. This paper discusses the current capabilities of Mobile Network Emulation for conducting empirical evaluation and demonstration of MANET technologies and is organized into 5 sections: (1) introduction to the role of emulation in network modeling, (2) mobile network emulators background (3) emulation system components (4) future plans and (5) conclusions.

INTRODUCTION

Computer networks have rapidly become an integral part of today's society, but the science of networking is still a relatively immature area. The complex interactions of multiple protocols and processes within a shared communication system do not readily lend itself to closed-form analyses. The dynamics of mobile, wireless networks further complicates attempts at formulating analytical solutions that characterize network system performance. Thus, modeling, including discrete event simulation and emulation of such systems has been long-recognized[1] and remains the most pragmatic approach to developing an understanding of expected behaviors[2].

Discrete event simulations of mobile network systems have been conducted for some time[3]. However, in the interest of scalability, such simulations have tended to abstract significant portions of the network system, particularly "upper-layer" protocol mechanisms. More recently, the increased capability and relative low cost of computing hardware has made it more practical to pursue "emulation"

systems that enable production software processes (applications and protocols) to be executed while simulating the characteristics of underlying mobile, wireless communication hardware.

Multiple mobile network emulation systems of varying capability and fidelity have been developed by Department of Defense (DoD) and other organizations during the past several years[4][5][6][7][8]. These systems have often served specific project or development goals. Also according to emulation survey papers[9][10][11] there are many different emulation systems available but no single system for all possible application scenarios. More recently there has been interest in developing more common, generalized approaches to mobile network emulation that can be leveraged by DoD and its partners. February 2009, NRL and ARL hosted a Mobile Network Modeling Workshop to facilitate information sharing on this topic. At the workshop, researchers from more than 15 organizations met to discuss their simulation and emulation research efforts. There was a general consensus that developing a set of interoperable tools was in the interest of the research community. This paper describes efforts by ARL, NRL & CENGEN to develop a modular mobile ad hoc network emulation environment to be made available to the broader research community.

I. BACKGROUND OF THE MOBILE AD HOC NETWORK EMULATOR

The Mobile Network Emulator (MNE)[12] system was developed by NRL and designed to run in a distributed fashion across a set of Linux-based hosts. MNE uses simple link-layer connectivity blocking provided by Linux IP (Internet Protocol) Table mechanisms to dynamically emulate mobile network connectivity. While MNE provides an easy way to examine network and application layer protocols' response to dynamics in connectivity, the fidelity of modeling the lower layers of the protocol stack was somewhat low. This was sometimes mitigated to some extent by running the emulator itself over 802.11 wireless interfaces on the Linux hosts.

As the need for more diverse and higher fidelity lower layers and additional capabilities grew, NRL developed the Mobile Ad-hoc Network Emulator (MANE) system[5]. Subsequently, NRL, ARL, CENGEN and other developers

have enhanced MANE to provide a larger-scale, higher fidelity emulation system capable of supporting arbitrary hosts in a “plug-and-play” fashion through use of centralized packet distribution processor(s). Both MNE and MANE have been used in virtual machine environments to stand up larger scale emulations with a modest hardware footprint.

More recently, NRL and ARL, in conjunction with CENGEN, have been collaboratively developing the Extensible Mobile Ad-hoc Network Emulation (eMANE) system based on our experience with MNE, MANE, and other simulation and emulation systems. The goal of eMANE is to provide a scalable system that is capable of modeling network systems with potentially complex, heterogeneous connectivity (i.e. multiple, different radio interfaces). Furthermore a modular approach is being developed to support relatively simple, distributed emulation system deployment, as well as higher fidelity centralized configurations. Support for virtualization has also been added to eMANE. The heart of the eMANE system, as shown in Figure 1, is the Network Emulation Module (NEM) that can be instantiated in multiple and variant forms as needed to model a specific network system.

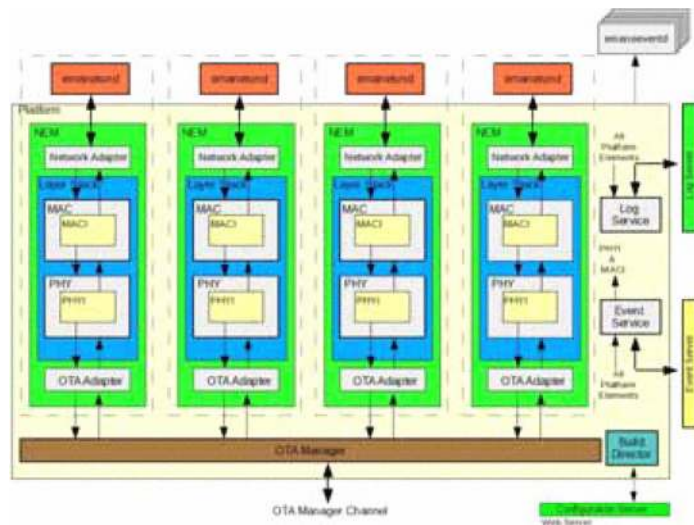


Figure 1 – eMANE Architecture

The NEM portion of eMANE provides a container, implemented as a dynamic loadable module, for software processes that perform packet input/output operations to emulate the characteristics of the link (Layer 2) and physical (Layer 1) layers of a network interface. The “Over-the-Air” (OTA) Manager serves as a bus to interconnect multiple, possibly distributed, NEMs within eMANE. Note that multiple NEMs may be instantiated on a single platform and/or distributed across a set of processing platforms. An NEM represents its emulated network interface to a host system (virtual or real) typically through the virtual interface mechanisms that are commonly available on

most operating systems. These virtual interfaces can be typically used through the host’s Internet Protocol (IP) stack or directly accessed via low level system calls for non-IP use.

Note that both the “Network Adaptor” and the “OTA Adaptor” are sufficiently abstracted and modularized in the code design that alternative mechanisms can be used, enabling the deployment of eMANE and its NEMs in different processing environments. This could include highly centralized deployment of NEMs on a central processor, possibly offering tight interaction for high fidelity link/physical layer modeling. Alternatively, NEMs might be distributed over a cluster of machines interconnected with high performance networking or some combination of centralized and distributed deployment to achieve larger scale emulations.

The eMANE components are designed for distributed, dynamic control and monitoring via the Event Service. Events can include updates to modeled “node” location or other attributes of the emulation environment and NEM parameters. Similarly, the Log Service is accessible by eMANE components so that an emulation experiment can be flexibly and richly instrumented. Additionally, “shim layer” functions can be inserted at different points in the NEM “stack” to augment functionality.

While Figure 1 suggests a typical NEM implementation with input/output modeling actions performed among a set of NEMs interacting via the “OTA Manager Channel”, it should be noted that the design is flexible enough to allow a NEM to be implemented in almost any form desired. For example, with an appropriate NEM implementation, the eMANE system can provide a common framework that can incorporate different (and possibly multiple) real-time Simulation-in-the-Loop (SITL) models of communication systems if such models are available. Thus, “best of breed” models from other tools (e.g. OPNET, QUALNET, ns-2, ns-3, etc) might be incorporated into an eMANE emulation environment as needed. To this end, much of the eMANE development is focused on creating the framework and glue to link the NEM to various components of the environment rather than on the packet communication “engine”, itself (i.e. NEM). For example, the control interfaces of eMANE are being developed to be modular and pluggable so that various forms of test control may be asserted including run-time, dynamic variations of emulation system parameters in response to network communication successes or failures. This will enable modeling of distributed, autonomous network-centric systems that are envisioned for future tactical edge systems.

In the interest of supporting DoD objectives in the development of effective mobile network technologies, the MNE, MANE, and eMANE emulation systems are being

made publicly available as open-source code[13]. Additionally, NRL is hosting a collaborative Mobile Network Modeling site available to DoD participants to support continued shared development and information exchange.

II. EMULATION SYSTEM COMPONENTS

After investigating different architectures for emulating MANETs, key aspects of an overall emulation system became apparent for present and future development. Development of eMANE has allowed for a very modular design and framework leading to a cohesive environment for emulation. Along with the core run-time emulation capability, additional supporting tools and applications are necessary to create a complete emulation environment. These include scenario generators and formal languages to design scenarios, test control systems, visualization tools, traffic generators, data collection and analysis tools, and virtualization and simulation-in-the-loop capabilities for providing more scalable networks. Creating such an environment requires the participation of multi organizations to ensure such applications are appropriate and functional. Because organizations have different requirements, the emulation environment needs to be fairly modular to allow different tools to be inserted. For example, it is possible with proper design that the tools here can be applied to other emulation systems aside from eMANE, or with discrete event simulations tools (e.g., OPNET, Qualnet, ns-3), or even to support field test activities.

SCENARIO Generation/Scripting and Visualization

An integral portion of an emulation environment is a tool for initializing and establishing a scenario for testing different parameters and mobility patterns. Several organizations have developed tools in this area, each focusing on different aspects of scenario creation, test control and visualization. ARL has developed a tool known as Topodef (Topology Definition)[14] which is a mobility scenario generator. The tool provides graphical methods for designing, animating and verifying a dynamic ad-hoc topology. The tool is also used for generating event-driven communication scenarios using NRL's RAPR[15] (The Real-Time Application Representative) tool. Recently, Topodef has been enhanced to support the computation of propagation loss using the DoD Terrain Integrated Rough Earth Model (TIREM)[16].

Another tool COMPOSER[17] (Communications Planner for Operational and Simulation Effects with Realism) was developed by Booz, Allen and Hamilton and Lockheed Martin. This tool provides real time realistic communications effects for network operations planning and virtual exercises allowing for the understanding of communications effects. This tool enables the in-theatre network manager to dynamically predict, plan and visualize wire-

less communications performance based on realistic simulation of maneuver and environmental effects. Another tool being examined for similar use is the NRL Interactive Scenario Builder tool that provides a number of propagation prediction algorithms and a graphical user interface for scenario planning[18].

Once a scenario has been defined and is running, a visualization tool is useful for verifying that the network is operating as expected. A useful tool which continues to evolve as the emulation environment develops is a real-time visualization tool developed by Cobham Analytic Solutions known as "Watcher". "Watcher" evolved out of efforts to understand the effects of attacks on MANET routing protocols and has since become a generalized network visualization tool. "Watcher" runs on the MANE-based testbed and displays geographic, test node status, and debugging information in both real-time and post-test-run playback modes. Display characteristics are controlled by the test nodes, which pass status information to "Watcher" via a custom API that supports multiple types of information presented in distinct visual layers. Watcher continues to be improved and efforts are underway to make it compatible with eMANE and incorporate real-time analysis.

Similarly, the NRL-developed Scripted Display Tool (SDT) provides a general-purpose visualization capability that supports features suitable for mobile network status display. More recently, the SDT family includes a three-dimensional (3D) variant (sdt-3d) that is built around the NASA World Wind Java SDK for geospatial visualization.

MIT Lincoln Labs is in the process of creating a DoD Wireless Network Library repository[19] that will store channel models, mobility traces, traffic traces and data sets from various DoD field exercises and experiments. This repository will allow the research community to access data that is much more relevant to the military operating environment. By helping develop more realistic experiments, the repository will facilitate the sharing of data sets and help leverage other research efforts.

It is expected that a number of different tools will be available to define and visualize various aspects of mobile network scenarios. It is planned that some standard semantics and scenario description formats will be developed for these anticipated tools to commonly use.

Test Control Systems

As the fidelity of mobile network experimentation environments increase, the complexity of configuring the system, conducting trials, and collecting and analyzing results may approach (or, in some cases, even exceed!) that of deploying an operational system. A number of tools are needed to support configuration control, orchestration, and archival of experiments data and results collected. Some

progress has been made in establishing some tools for these purposes.

The first application is a tool developed by ARL known as the Network interdisciplinary Computational Environment (NiCE). NiCE is a new software architecture for assessing and researching Mobile Ad-Hoc Wireless Networks and leverages the HPC CSE (Computational Science Environment) [20]. NiCE provides a stable suite of 64-bit Linux (adaptable to 32-bit Linux) wireless network analysis and assessment tools, applications and libraries that are easily activated via modules[20]. The environment allows users/developers to leverage previous work completed in the HPC environment and helps solve portability and flexibility issues. NiCE ensures that software is built using the correct versions of dependent libraries for the system. This tool is currently being beta tested at ARL and is not yet available for public release.

An existing tool that was useful in the original MANE development is the Test-bed Control System (TCS)[13] created by NRL. TCS is used to integrate, organize, control and automate the testing process of the MANE environment. Such a tool will need to be adapted to the new eMANE architecture to allow for the same capabilities.

NRL is also creating an XML-based schema for describing mobile network scenarios and experiment plans. This consists of a set of “planning” document types and “scripting” format for respectively specifying and executing an experiment, including node mobility and other aspects of dynamic environments and network operation. It is hoped that a common “language” to describe mobile network scenarios and experiments can allow cross-fertilization across a modular set of tools for planning and controlling emulation-based testing.

The goal is to simplify the creation of useful and relevant scenarios in a way that can also take advantage of real-world data collection. This allows multiple applications to interact together and leverage the work between tools. Our hope is to create a common language that others can use to develop scenarios. A common scenario description will allow sharing between different organizations and tools. This effort will leverage existing work on a military scenario description language[24] and eventually be tied into ARL’s Network Data Model and Format (NetDMF) language which is an extension of ARL’s existing eXtensible Data Model and Format (XDMF) language[25].

Network Traffic Generation

Controlled network traffic generation is needed to conduct meaningful experimentation in these environments. This is necessary to understand the effects a loaded network has on applications, algorithms, and protocols. To support the performance evaluation of mobile, wireless networks,

NRL has developed some scriptable traffic-generation tools. These tools can be used to quantitatively measure system performance in terms of goodput, data delivery latency, and success of transfer. The Message Generator (MGEN) [26] tool is capable of supporting dynamic unicast and multicast data flows including use of UDP (User Datagram Protocol), TCP (Transmission Control Protocol) or externally-available transport protocols. The Real-Time Application Representative tool (RAPR) is an extension of the MGEN tool that allows interactive communications (e.g. query/ response) to be scripted and the outcome of those interactions to influence subsequent traffic loading and behaviors. Data analysis programs are available for data reduction and visualization of results obtained with these tools.

The Massachusetts Institute of Technology’s Lincoln Laboratory (MIT-LL) has developed a comprehensive system to support traffic generation in network emulations. The Lincoln Realtime Information Assurance Testbed (LARIAT) [27] suite provides mechanisms that consist of virtual users exercising real applications including the ability to model enterprise-level networks of more than 100,000 users. This includes profiles derived from real networks as well as other traffic models. The LARIAT system continues to evolve with planned extensions for military tactical applications (e.g. Situational Awareness) in addition to the existing enterprise application suites. Currently, LARIAT does not interoperate with the eMANE environment. However, once LARIAT is extended to incorporate tactical military applications, LARIAT could become a good source of network traffic.

Data Collection and Analysis

A key component for any emulation environment is a tool which can provide accurate and understandable network analysis. Some open source tools for data collection and analysis have been developed by Dark Corner Software LLC[28]. This software allows users to run tests using representative applications with a high degree of information gathering on MANET networks. Supporting analytical tools are also provided. One example is the Monitaur a suite of tools that provide a network-based application scripting and monitoring system. The tools provide distributed testing, monitoring and control of applications on a multiple node network. An essential part of this tool is the Monitaur Analysis tool, which provides post-processing analysis. It focuses on providing a framework for data collection, reduction and report generation. The logs gathered are from NRL’s MGEN program. This suite provides the ability to perform IP network performance tests and measurements using UDP/IP traffic. The toolset generates real-time traffic patterns so that the network can be loaded in a variety of ways and generates reports on the network’s performance. The report generation provides

extensive plotting and reporting abilities, including: generating time plots, histograms, scatter plots, comparing data across multiple tests, and aggregate statistics.

The ability to analyze the large amounts of data gathered in an emulation experiment and display the analysis in a meaningful manner is crucial to the comprehension of these complex networks. A modular approach to data collection and analysis, to include common data formats, makes it much easier to conduct multiple analyses on a single set of data. In particular, common data formats such as NetDMF allows data to be visualized using a variety of visualization tools without large amounts of preprocessing to reformat the data.

Virtualization and Simulation-in-the-Loop

In order to conduct meaningful analysis of large networks, the emulation of 1000's of nodes representing a brigade combat team. In order to avoid purchasing, installing and powering thousands of PCs, we have looked to virtualization to reduce the amount of hardware necessary to create an emulated network of $O(1000)$. This is useful to reduce the amount of cost for purchasing hardware and simplifies the initial setup of the systems. We are currently optimizing the systems being used to maximum performance given the amount of virtual machines on a particular system.

Another area we are investigating is the use of simulation-in-the-loop in our experiments. This will allow us to leverage the military radio models that have already been developed for simulation tools such as OPNET and Qualnet. Simulation-in-the-Loop capability would provide us the ability to rapidly incorporate validated models of tactical radios without having to develop and validate a model specifically tailored for emulation. It is still unclear whether the simulation models can be scaled up to 1000's of radios and still run in real time.

III. FUTURE CAPABILITY ENHANCEMENTS

Many organizations within the DoD are developing emulation testbeds. In February 2009, NRL hosted an Emulation Workshop which drew participants from more than 15 different industrial, academic, and governmental organizations. While many of the organizations were leveraging each others work, there was a significant amount of work that was not being leveraged. ARL & NRL would like to develop the eMANE environment into a modular set of tools which the broader research community could use. In particular, we'd like to facilitate the interoperability of various scenario development, visualization and analysis tools as well as enable the reuse of various radio/protocol models. This would allow researchers to swap out components and assemble an emulation environment that meets their research needs without having to recreate a particular capability.

In addition to creating a common modular environment, we would like to increase the scalability and fidelity of the emulation environment. The real time processing requirement limits the fidelity and potentially the scalability of the emulation environment as the size of the emulated network grows to the 1000's of nodes.

One way to get around this is to use high performance computing (HPC) hardware. The DoD has a significant investment in HPC hardware that can be leveraged to scale the emulation capability to beyond a 1000 nodes. In addition to the availability of HPC systems with 1000's of processors, HPC environment has the software tools in place to facilitate the rapid transfer of data between processors, which can increase scalability significantly. HPC resources can also be used to run detailed high fidelity RF propagation and physics models in real-time.

The DoD High Performance Computing Modernization Office has funded ARL to establish a Software Application Institute to develop multi-disciplinary expertise and software tools that transform the ways in which DoD models, simulates, emulates, and experiments with dynamic reconfigurable mobile tactical networks.

ARL's Mobile Network Modeling Institute[29] will reengineer the eMANE emulation environment to operate on the HPC resources installed at ARL's DoD Supercomputing Resource Center, and extend the size of the networks that can be emulated to $O(1000)$ nodes. In the mid term, the Institute will extend the HPC eMANE environment to support real-time interactions with force-on-force and command and control (C2) applications and the implementation a real-time high fidelity RF propagation modeling component into MANE.

IV. SUMMARY

In summary, we are announcing the release of developing a modular extensible emulation environment. The goal is to develop an openly-available suite of tools and applications that can be combined and extended as necessary to provide a high-fidelity, scalable emulation environment for mobile wireless networks. The modular design will allow for synergy amongst multiple organizations. Our realistic emulation environment allows for the verification of protocols and algorithms being developed for mobile ad hoc networks. Developing a common taxonomy and semantics for describing mobile network scenarios and experiments from the conceptual stage to the execution stage will increase our ability to thoroughly examine this emerging technology area in a meaningful way. It is hoped that a sufficient understanding of the technology in a broad set of relevant and perhaps even mission-driven scenarios and environments will enable its adoption with a high level of assurance for increased capability in mission-critical tac-

tical systems. We are seeking the support of other organizations to join in the effort to make this an open modular set of tools which the broader research community can use and extend.

REFERENCES

- [1] F. Zurcher and B. Randell, "Iterative, multi-level modeling – a methodology for computer system design," in Proc. IFIP Congress '68, 1968, pp. 867-871.
- [2] L. Breslau, D. Estrin, K. Fall, S. Floyd, J. Heideman, A. Helmy, P. Huang, S. McCanne, K. Varadhan, Y. Xu, and H. Yu, "Advances in Network Simulation", IEEE Computer 33(5), 59-67, May 2000.
- [3] J. Short, R. Bagrodia, and L. Kleinrock, "Mobile Wireless Network System Simulation", in ACM Wireless Networks, Vol 1, Issue 4, pp 451-467, 1995.
- [4] W. Chao, J. Macker and J. Weston "NRL Mobile Network Emulator" IEEE MILCOM 2003, 2003.
- [5] N. Ivanic, B. Rivera, R. Gopaul, B. Luu, D. Gwyn, R. Hardy, K. Marcus, L. Scott, G. Tran, and B. Nguyen, "A Scalable Testbed for Emulating Wireless Mobile Ad-hoc Networks", IEEE MILCOM 2008, 2008.
- [6] M. Kaplan et al, "WISER: Realistic and Scalable Wireless Mobile IP Network Emulator", available at http://wiser.research.telcordia.com/images/WISER_Whitepaper_2008.pdf
- [7] S. Doshi, U. Lee, R. Bagrodia and D. McKeon, "Network Design and Implementation Using Emulation-based Analysis", in Proc. of IEEE MILCOM 2007, 2007.
- [8] <http://www.tel.fer.hr/imunes/>
- [9] Wolfgang Kiess and Martin Mauve, "A Survey on Real-World Implementations of Mobile Ad-Hoc Networks", in Elsevier's Ad Hoc Networks 5 (3), pp. 324-339, April 2007.
- [10] Matthias Kropff, et al, "A Survey on Real World and Emulation Testbeds for Mobile Ad hoc Networks", in TRIDENTCOM 2006.
- [11] Erek Gokturk, "A Stance on Emulation and Testbeds, and a survey of Network Emulators and Testbeds", in European Conference on Modeling and Simulation 2007.
- [12] J. Macker, W. Chao, and J. Weston, "A Low-cost IP-based Mobile Network Emulator (MNE)", in proceedings of IEEE MILCOM 2003.
- [13] <http://cs.itd.nrl.navy.mil/products>
- [14] B. Nguyen, "The ARL Topodef Tool for Designing Mobile Ad-Hoc Network Topologies to Support Emulation," Military Communications Conference, 2007. MILCOM 2007. IEEE, vol., no., pp.1-6, 29-31 Oct. 2007.
- [15] <http://cs.itd.nrl.navy.mil/work/rapr/index.php>
- [16] <http://www.alionscience.com/index.cfm?fuseaction=products.view&productid=19>
- [17] O. Ukrainsky, et al., "An Open Environment for Rapid Embedded Planning of on-the-move Communications Networks using Multit-Level abstraction," Military Communications Conference, 2005. MILCOM 2005. IEEE, vol. 4, pp. 2631-2636, 17-20 Oct. 2005.
- [18] <https://builder.nrl.navy.mil>
- [19] <http://wnl.ll.mit.edu>
- [20] J. Renteria, et al, "The Computational Science Environment (CSE)", HPC User Group Conference, 2008
- [21] J. Furlani et al, "Abstract Yourself With Modules", Lisa'96: Proceedings of the 10th USENIX conference on System Administration, 1996
- [22] L.M. Rossey, et al, "LARIAT: Lincoln adaptable real-time information assurance testbed," IEEE Aerospace Conference Proceedings, 2002. IEEE Vol 6, pp 6-2671-2682, 2002.
- [23] <http://www.darkcornersoftware.com/index.html>
- [24] J. Abott, "Military Scenario Definition Language Overview"
http://msdl.pergu.eu/Specifications/presentations/Military_Scenario_Definition_Language_Overview.pdf
- [25] J. Clarke and R. Namburu, "A Distributed Computing Environment for Interdisciplinary Applications" Concurrency and Computation: Practice and Experience, 2002
- [26] <http://pf.itd.nrl.navy.mil/mgen/mgen.html>
- [27] L.M. Rossey, et al, "LARIAT: Lincoln adaptable real-time information assurance testbed," IEEE Aerospace Conference Proceedings, 2002. IEEE Vol 6, pp 6-2671-2682, 2002.
- [28] <http://www.darkcornersoftware.com/index.html>
- [29] <https://hsai-web.arl.army.mil/pub/mnmi>