

Missile System Modeling in a Distributed Virtual Environment

Greg Tackett

U.S. Army Aviation & Missile Research, Development,
and Engineering Center
Redstone Arsenal, AL 35898
greg.tackett@rdec.redstone.army.mil

Tim McKelvy

U.S. Army Aviation & Missile Research, Development,
and Engineering Center
Redstone Arsenal, AL 35898
tim.mckelvy@rdec.redstone.army.mil

William Greenleaf

Gleason Research Associates
Redstone Arsenal, AL 35898
William.Greenleaf@rdec.redstone.army.mil

In April 2003, The U.S. Army Research, Development, and Engineering Command (RDECOM) conducted an experiment called the First Application (1stApp), using a distributed virtual environment across a wide area network. The two objectives of this HLA-based distributed simulation were to establish a modeling and simulation cost/schedule/performance baseline for the development of a target distributed virtual environment for Future Force technical experimentation and to match tactics to physics to produce realistic expectations of the technical performance of networked fires for the U.S. Army Future Combat Systems (FCS).

The core set of models and simulations for this event was developed and provided by the networked fires using Non-Line-of-Sight Launch System (NLOS-LS) Science and Technology Objective (STO). In particular, NLOS-LS developed models and virtual simulations of the NLOS-LS missiles and Mission Management Application (MMA), including modeling of the LADAR seeker on the Loitering Attack Missile (LAM) and the man-machine interfaces to display and utilize LADAR imagery in the battlefield context.

This paper describes the LADAR models, missile flight models, MMA virtual simulations, LADAR visualization, and networked fires context that was used to represent NLOS LS in the 1stApp event. It also describes the distributed virtual environment and the simulation architecture that was used to represent the networked fires functionality, through the integration of simulations from a variety of organizations within the RDE Command.

Keywords: Seeker, LADAR, missile, NLOS-LS, LAM, 1st App

1. Introduction

In October 2002, the Research, Development, and Engineering Command (RDECOM) was established to integrate the Research, Development, and Engineering Center (RDEC) components of the Army Materiel Command. A virtual distributed lab for modeling and simulation was initiated and selected to execute RDECOM's First Application (1stApp). The objectives of 1stApp were to provide insights into the networked fires process and performance for Future Combat Systems (FCS) and to define the baseline capability of RDECOM to conduct distributed simulations.

1stApp was designed as a distributed real-time simulation architecture utilizing established and emerging models and simulations at key RDECOM modeling and simulation (M&S) facilities, linked with representative test and user communities. Key M&S functions were provided by the Army Aviation and Missile RDEC (AMRDEC), the Armaments RDEC (ARDEC), Tank and Automotive RDEC (TARDEC), Communications and Electronics RDEC (CERDEC), the Army Research Laboratory (ARL), and the Simulation and Training Center (STC).

While the overall objectives of 1st App covered a broad range of technical issues, this paper focuses on work performed to develop modeling and simulation tools for the Non-Line-of-Sight Launch Systems (NLOS-LS). The

NLOS-LS represented the majority of non-line-of-sight and beyond-line of sight effectors. The architecture federated over a dozen dissimilar simulations in a classified real-time distributed simulation environment for the express purpose of identifying and testing baseline NLOS-LS Command, Control, and Communications requirements in a Unit-of-Action context. As such, primary emphasis was placed on modeling the sensor/system concept for the Loitering Attack Missile (LAM) and the Precision Attack Missile (PAM). This paper specifically discusses the M&S tools developed to address the LAM reconnaissance function.

The NLOS-LS Weapon System consists of two missile systems that can be launched from a Container/Launch Unit. One missile is referred to as the LAM. The LAM can conduct a combination of reconnaissance and attack mission profiles. This report focuses on the reconnaissance mode. Following fly out to the target area, the missile will loiter and search for candidate targets and transmit potential targeting information via a radio frequency link to the system controllers. The LAM can engage a variety of targets after transitioning to the attack mode. The LAM will provide flexibility to the force by allowing several search missions. The search patterns and the target priorities will depend on the mission, but the LAM can follow specified routes, monitor specific areas, or perform a large area search in the target area. Potential search strategies include serpentine motion, racetrack patterns, and rosette patterns. Targets found by the LAM can be engaged with the PAM, or with other weapons systems available to the battle commander. The LAM missile will employ a laser radar (LADAR) sensor to perform target search and engagement. The PAM will employ an uncooled imaging infrared sensor to engage targets autonomously. PAM also will employ a Semi-active Laser sensor to engage designated point targets. This paper describes modeling and analysis tools for analyzing several LADAR concepts and simulating them for the LAM mission in the Interactive Distributed Engineering Evaluation and Analysis Simulation (IDEEAS). IDEEAS was employed as a simulation driver and also was configured as a missile server to respond to remote launch commands.

2. Simulation Architecture and Components

The 1stApp architectural design was based on a requirement to utilize existing RDECOM facilities and simulations, which necessitated the physical backbone and simulation infrastructure to derive from those currently in use to support RDECOM customer experiments. A High Level Architecture (HLA) simulation backbone was laid across the physical connectivity to provide simulation interactions. Sub-networks of Distributed Interactive Simulation (DIS) traffic were bridged into the HLA architecture at three sites, then the appropriate HLA federates and DIS simulations were connected to provide the overall digital architecture.

2.1 Wide Area Network (WAN) Architecture

Geographic distribution of the event was accomplished by linking four simulation sites with one wide area network monitoring and collaboration server site, and physically bringing resources from the other organizations to the four simulation sites. These distributed simulation sites are all connected to the Defense Research and Engineering Network (DREN), and have historically utilized the DREN to connect their real-time simulation capabilities, particularly in support of the RDEC Federation Calibration Experiment (CalEx) [1], [2].

A driving requirement for the physical architecture was classified operation. Encryption devices added complexity while they improved connectivity experienced during CalEx, by allowing the network to bypass metropolitan area network firewalls and access control lists.

Using this approach, WAN network performance, in terms of latency and throughput, was demonstrated as 2-36 ms latency from AMRDEC to the other sites by ping test, and less than 0.3% throughput loss as measured by nuttcp tests with AMRDEC as server. Bandwidth utilization during runs was observed to be less than 0.37 Mb/s on average, with an observed peak of up to 9.67 Mb/s. A more detailed discussion of the 1stApp network design and performance is given in [3].

2.2 Simulation Architecture

The 1stApp simulation architecture supported the execution of 70 DIS workstations, three gateways, and six HLA federates for 40 hours of record runs of scenarios running up to three hours each, involving a maximum of 2,500 entities per run. Up to 2 million Protocol Data Units were generated for runs at these levels. With this amount of traffic, and the WAN performance described above, 1stApp appeared not to reach any performance thresholds which would inhibit the experimental evaluation of networked fires, with the exception of some line-of-sight fires with simulation traffic that had to pass between multiple gateways.

Most of the existing RDECOM simulations were developed using DIS protocols for interconnectivity, and have demonstrated compliance with HLA using the Real-time Platform Reference (RPR) Federation Object Model (FOM) and DIS/HLA gateways.

During CalEx, the RDEC Federation experimented with RPR FOM extensions to pass non-DIS compliant information, such as target acquisition truth data, and distributed data collection information. However, during the design of 1stApp, it proved possible to configure all data exchanges into DIS compliant Protocol Data Units and corresponding RPR FOM data without the need for extensions. This allowed use of commercial DIS/HLA gateways without customization.

In addition to the gateway federates, there were several native HLA federates. In order to control and limit WAN

traffic and avoid feedback loops, only HLA traffic was allowed to pass across the WAN. Consequently, any DIS-to-DIS traffic between sites had to pass through two gateways to encode and decode data packets, which impacted performance in terms of data latency. This led to the addition of a small red cell at a remote site to allow for local blue/red force interactions to avoid the multiple gateway latencies mentioned above.

2.3 Experiment Control and Data Collection Tools

1stApp utilized the Mak HLA Run-time Infrastructure and Mak gateways as commercial products. The Run-time Infrastructure Execution process was run from the experimentation control and data collection cell at AMRDEC. The commercial product HLA Results was used as the primary distributed data collection tool. HLA Results was run on several platforms to allow simultaneous data collection during runs, post-processing, and analysis queries.

1stApp also utilized the AMRDEC-developed Data Collection and Analysis Tool (DCAT) for real-time monitoring of battlefield statistics. DCAT was originally developed for DIS, and was converted to an HLA interface in support of CalEx. Since the underlying data structures for DCAT are still DIS based, it was closely aligned to RPR to support 1stApp. While DCAT focused on effects data, another run-time data collection tool was developed from a rules-based engine, which provided real-time monitoring of target acquisition and fires process data, user selectable during the runs.

OneSAF Test Bed (OTB) and the Vulnerability/Lethality server were also used in data collection modes, even though those simulations typically are used as simulation truth data generators. Since OTB was not the primary scenario generator simulation for 1stApp, it was used to monitor battlefield entity status and enumerations to reduce risk for later use of OTB. Likewise, the Vulnerability/Lethality server eventually will be responsible for setting the damage states; but for 1stApp, it was used to monitor state changes for verification purposes.

Other typical data collection and viewing tools like plan view displays and stealth viewers were also used as needed. A more thorough discussion of lessons learned and results from this data collection effort is given in [4].

2.4 Truth Data Simulations

1stApp was scoped to represent the quantities and types of systems composing an FCS Unit-of-Action, based on the Army Systems Book, using the then-current version 1.6 document.

The primary scenario engine for 1stApp, providing platform-level performance for the bulk of the red and blue

forces, was the AMRDEC-developed Interactive Distributed Engineering Evaluation and Analysis Simulation (IDEEAS). IDEEAS is a constructive simulation which is synchronized with real-time to support virtual experimentation, with a low level of runtime user interactions. All entities in 1stApp were generated by IDEEAS, except as identified with the other models described below.

FIRESIM from Fort Sill was used to represent all red and blue indirect fire platforms, with the exception of about half of the Non-Line-of-Sight Launch System (NLOS-LS) platforms which are otherwise represented by the NLOS-LS Mission Management Applications (MMAs). FIRESIM was also utilized to represent all counter-battery radar systems.

OTB represented a small number of air sensor platforms used to populate fused sensor data. OTB also represented the local red ground force at CERDEC for interaction with sensor simulations.

When launch platforms fired missiles and armaments, the flights, interactions, and detonations of those entities were generated by the Missile Server from AMRDEC and the Armament Servers from ARDEC.

The TARDEC manned simulator and Human Performance Model represented two recon vehicles, which controlled Armed Reconnaissance Vehicles. The CERDEC manned simulator suite represented a single recon vehicle, which also controlled a Class I Unmanned Aerial Vehicle (UAV), a Small Unmanned Ground Vehicle, an Unattended Ground Sensor suite, and an Intelligent Munition System. There also were several desktop simulators of Class I-IV UAVs, and dismounted forward observers at AMRDEC.

2.5 Perceived Data Simulations

Command, Control, and Communications (C3) functionality was focused on those functions critical to networked fires, which included target reporting, sensor fusion, situational awareness, weapon-target pairing, calls for fire, fire missions, and Battle Damage Assessment.

Communications representations were limited to two key areas: communications between ground sensor suites and the recon vehicle, which were simulated by a CERDEC model, and communications between NLOS-LS missiles and their control stations and relays, which were simulated by the AMRDEC Comms Server.

Many of the existing tools from the RDECOM organizations had overlapping functionality in Command and Control, so the challenge was to scope meaningful roles for these simulations so fire missions could be executed at various levels of command. Collectively, these products provided the functionality of a Network-centric system without explicitly modeling the individual nodes and interfaces of that system.

Mission threads were represented as follows:

Sensor reports from various recon assets were sent to

the CERDEC Sensor Exploitation and Management System (SEAMS) simulation for fusion and forwarding to assets including the Mobile Command and Control (MC2) system and Protocommand. Other sensings were sent directly into the network systems without the fusion function occurring.

MC2 then updated situational awareness and Protocommand utilized an automated Attack Guidance Matrix and the dynamic organizational allocation of fires to determine weapon-target pairings. MC2 also sent target information to FIRESIM for processing.

From Protocommand and FIRESIM, fire requests were issued, either directly to firing platforms or for further allocation to the ARDEC Combat Decision Aiding System, depending on which echelon of command was executing the fires.

For fires allocated to the NLOS-LS system, depending on which batteries were assigned, FIRESIM or the NLOS LS Mission Management Application determined the weapon flight paths and executed the fires.

2.6 Terrain Representation

The terrain database used in 1stApp was a geotypical Caspian Sea terrain, which includes several cities, in a 131x131 km playbox. The terrain also contains many crop fields and powerlines that are located around these cities, and a central airport. The bounding coordinates of the terrain are as follows:

West-bounding coordinate:	46.49749546 E
East-bounding coordinate:	48.08519989 E
North-bounding coordinate:	40.15330927 N
South-bounding coordinate:	40.00829279 N

The source data for the terrain was primarily DTED level 1 and DFAD level 1, with geotypical urban features and terrain textures added notionally. No geospecific imagery was used in the creation of the database. The model feature set consisted of 24 building models and three powerline pylon models at three levels of detail. The overall polygonal density was approximately 5180 polys/km at the highest level of detail, which is driven by cultural feature density with open areas less dense than urban regions. There were 105 geotypical textures used in the terrain. The native formats that were generated to support the heterogeneous simulation set consisted of Open Flight, CTDB, ELE, Janus, TerraPage, JCATS, and XIG.

3. Methodology for Simulating the LAM Mission in IDEEAS and the 1st App

For the 1st App experiment, a detailed LADAR simulation was incorporated into the Interactive Distributed Engineering

Evaluation and Analysis Simulation (IDEEAS)-based Missile Server, which was responsible for instantiating the NLOS-LS missiles. The complete system representation of the NLOS-LS included three major simulation components to create NLOS-LS engagement threads: the Missile Server, the Mission Management Application, and the Communications Server.

3.1 IDEEAS Missile Server

The Missile Server provided the NLOS-LS missile service to the 1stApp. Based on a legacy validated effectiveness model, the Battlefield Effectiveness Weapon System Simulation, the Missile Server integrated the missile three degree of freedom kinematics models, the missile seeker models, and the interface to battle command elements. The three-degree-of-freedom flight simulation flies the LAM based on missile aerodynamics design data, mass properties, guidance and navigation, and propulsion properties. At each time step (default 1 second), the LADAR field-of-view is checked for targets whose truth data is known by the simulation via the distributed simulation interface. If targets are in the field-of-view, the laser range equation is checked to determine if the LADAR can range to the target. If the laser signal-to-threshold is greater than one, based on the range equation, then the number of pixels on the target is computed and the probability-of-acquisition is computed. A random number is drawn to determine if the target is detected. If successful, a message is created that informs Battle Command of a target detection and appropriate action is taken. While the Missile Server employs several simplifying assumptions to incorporate the LAM LADAR model, it simultaneously employs a higher fidelity model of the missile trajectory, terrain, and targets, thus creating a balanced, credible representation.

3.2 Mission Management Application (MMA)

The MMA prototypes the functionality required for human operators to plan, execute, and maintain NLOS-LS missions. This functionality is logically and physically decomposed into two distinct functions; mission planning and sensor imagery display. The mission planner ultimately is responsible for the command and control of the NLOS-LS system, the planning and execution of its fires, and the interrelationship with the other components of the networked fires mission. It consists of a graphical user interface with a two-dimensional plan view display, facilitating maneuver planning for the fire units (container/launch units) and route planning/updating for missiles. As the effective footprint of the LAM seeker is nominally less than 200m wide, the availability of and correlation quality of feature data (i.e., roads, rivers, structure etc.) is of paramount importance. The complimentary sensor imagery display is primarily concerned with the display of missile seeker imagery for both LAM (LADAR) and PAM

(Uncooled Infrared). For LAM missions the Missile Server generates target detections from a LADAR search using the model described previously. However, no synthetic imagery is actually passed via the distributed simulation interface. Rather, the sensor display generates the imagery locally on the fly, based on the geometry of the detection.

The LADAR imagery was presented in three distinct methods. First, the default mode presented LADAR “chiplets” to the operator, centered on the detected target, and cropped to a small image size of approximately 40x40 pixels. As LADAR data contains both reflectance and range data, these chips were drawn in three simultaneous frames (reflectance, false color rollover in range, and false color rollover in height above terrain). The second mode, called a LADAR Swath, generated a stream of these chips, simulating a continuous feed of the LADAR data for short bursts of time. In this mode, there were no pixels on target (POT)-based Automatic Target Recognition (ATR) modeling running; instead, the operator was observing the raw data stream in quasi-real time. Finally, a third mode, called LADARVis, captured the data in the continuous stream and used it to build a three-dimensional view of the collected data set. In the execution of the 1st App experiment, the sheer workload of the MMA Operators (managing up to 20 missiles in the air at any given time) precluded the use of all but the default mode.

3.3 Communications Server

As the NLOS-LS missiles were launched and flew their assigned missions, they created an ad hoc mobile area network. One of the key research topics of the NLOS-LS Full System Simulation was to derive the network traffic load to be expected on the NLOS-LS communications subnet in order to drive the development of the NLOS-IS waveform. The Communications Server represented the network in the simulation experiments, dynamically introducing the effects (latency and bandwidth restrictions) of the load on the missile ad hoc network. Each command and control message and seeker imagery message passed between the MMAs and the missiles were in essence passed through the Comms Server, which not only injected the effect of bandwidth and latency performance, but also the line-of-sight effects for the longer range missions. The data populating the Comms Server for the NLOS-LS network was generated offline using a very high fidelity OpNet communications model (including radio characteristics, antennae patterns, and the networking protocols).

4. LADAR Sensor Modeling and Simulation

The LADAR sensor consists of a low-power 1.06 micron laser operating at a very high pulse repetition frequency (PRF). The sensor does not have a field-of-view in the sense that other sensors do, rather it paints the terrain in front of

the missile by using a combination of high frequency sensor nodding, and rotating the sensor head in a yawing motion as the missile flies at a constant speed and altitude. A graphical depiction of this process is illustrated in Figures 1 and 2.

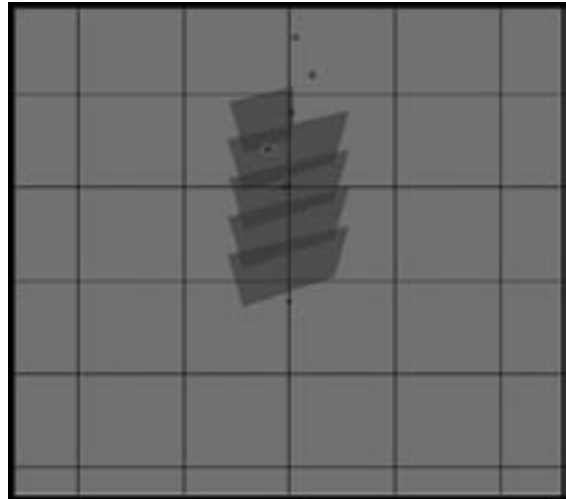


Figure 1. Top-down perspective

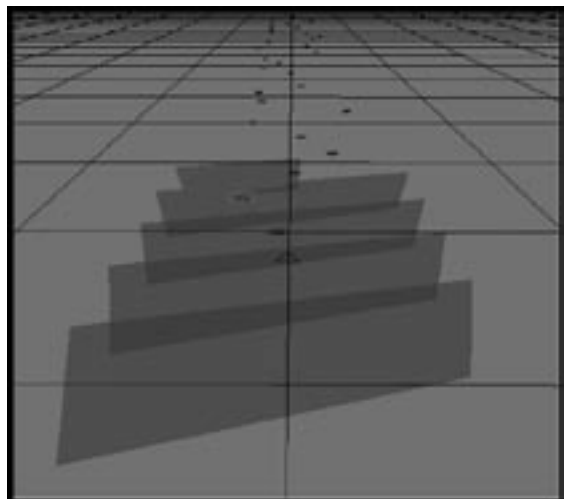


Figure 2. Three-D perspective

These figures were generated using the GUI that was developed for the LADAR model. The first figure shows four and a half templates on a 500 meter grid. The missile appears as a small triangular dot approximately one km below the upper template. The missile is flying from south to north. The second picture shows the templates on the grid from a perspective above and behind the missile. The missile location is defined by the triangle in Figure 2. In both figures, the small dots define the location of tactical targets placed along the flight path of the missile. Detected targets are highlighted with a circle. The sensor collects LADAR return data during multiple nods of the LADAR head, while the

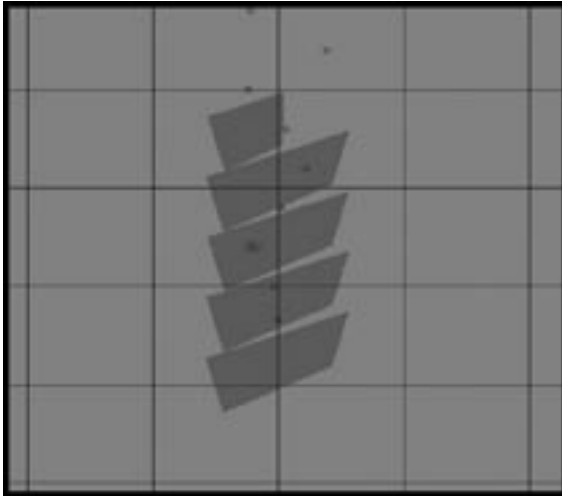


Figure 3. Incomplete coverage (example)

sensor is yawed in one direction. The yaw scan can be either right-to-left or left-to-right. The result of a single yaw scan is a template of LADAR returns from the terrain at a fixed range in front of the missile. High-resolution range and amplitude information is collected during the template definition. The LADAR data within the template is processed using Automatic Target Acquisition (ATA) Algorithms to locate target-like objects on the terrain. During the ATA template processing, the missile continues to fly downrange while the sensor head is yawed back to the starting position before collecting the next template. The data defining the template consists of laser return intensity and range. Therefore, a high fidelity map of the terrain and target is generated in the range template. The resolution of the LADAR data in the template is a function of missile speed and altitude, the sensor depression angle, and the LADAR parameters (nod rate, yaw rate, and PRF). If the missile is moving too fast, gaps will result in the terrain coverage, whereas, if it is moving too slow, there will be an overlap from template to template. Figure 3 illustrates the effect of flying at too great a speed for the specified LADAR parameters, resulting in gaps on the terrain between the templates. The downrange dimension of the template is a function of the sensor nod angle, the missile speed, and the sensor depression angle below the horizon. The cross range dimension of the template is a function of the yaw limits, the depression angle, and the missile altitude. The spatial resolution of the data in the template is a function of the LADAR PRF, and the range from the missile to the terrain which is controlled by the missile altitude and sensor depression angle. Summarizing, the LADAR runs at a constant PRF and the missile flies at a constant speed and altitude. The LADAR head nods down at one rate while data is collected, and at a faster rate while the head is returning. The sensor yaws at one rate while the template is being defined, and at a faster rate in the opposite direction to begin the next template. Every parameter must be set relative to all others to “paint” the terrain with the proper resolution,

and with minimal gaps or overlaps. Typically, efficiency greater than 70% can be obtained (over 70% of the time, the sensor is collecting useful data while generating a template). The remainder of the time, the sensor is either nodding up or yawing back to the return point. The LADAR efficiency is high because a single laser pulse typically is split into a horizontal array of sub-pulses, which are transmitted with a fixed angular separation, as illustrated in Figure 4. The angular separation of the LADAR pulses and the range to the terrain defines the horizontal resolution of the template. The nod rate and the yaw rate must be set based on the laser PRF to provide an identical vertical and horizontal resolution. The nod rate must be set such that the vertical resolution matches the angular resolution of the sub-pulses.

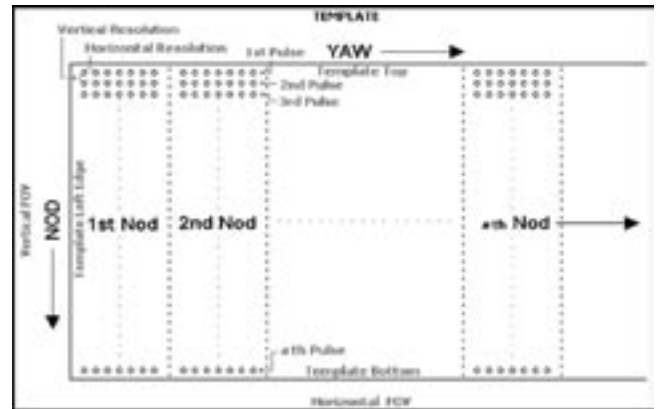


Figure 4. Template developments by nodding and yawing the LADAR boresight

4.1 Sensor Simulation Objectives and Methodology

The stand-alone model of the LAM concept addresses the overall LAM mission including the LADAR, the scan pattern, missile trajectory, and target descriptions. This model places targets on a flat plate terrain and flies the LAM on a specified trajectory while the LADAR scans and acquires targets. The detected targets are processed as a Target Report and the estimated target positions and time of detections are forwarded to a mission station where other assets, including the PAM, can engage them. This model was developed in a stand-alone mode to analyze several LAM LADAR concepts, and then portions of the model were inserted into IDEEAS to simulate the LAM missile, the LADAR sensor, and the LAM mission in a distributed force-on-force environment. In order to model the performance of the LADAR sensor it is necessary to specify all the parameters relative to the LADAR and the missile trajectory. In addition, it is necessary to define data to specify the targets and the environment. For a LADAR sensor, the target is defined by its physical size and shape, and its reflectivity. The environment is defined by the meteorological visibility. In addition, the environment

can include several adverse weather conditions including rain, snow and fog, as well as countermeasure smokes. When adverse weather or obscurant smokes are present, attenuating the laser pulse in the path from the sensor to the target simulates these effects. Figure 5 presents the overall flow of the simulation.

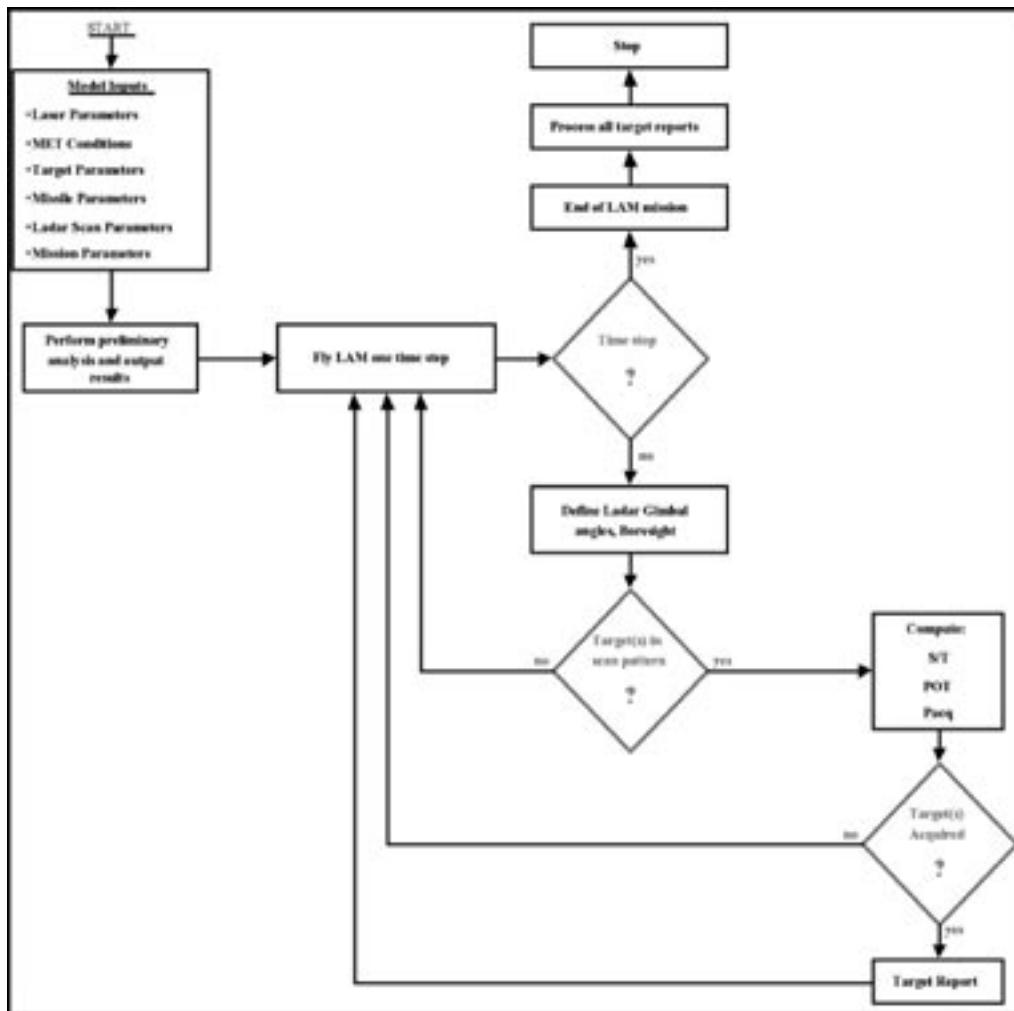


Figure 5. Simulation flow diagram

The objective of the simulation is to model the acquisition process for the LAM LADAR sensor. The simulation flies the LAM on an input trajectory and turns the LADAR on at specified times. The LADAR is not employed for target search unless the LAM is flying on a straight line in level flight at the proper altitude and speed. When the LADAR is operational, the sensor boresight is properly depressed and yawed through input limits to create the template on the terrain. When a target is scanned by the LADAR, the simulation performs computations to determine if the sensor acquires the target. The acquisition model computes Probability-of-Acquisition (Pacq) as a function of the following:

- LADAR parameters—Parameters for a typical LADAR concept include power, power per pulse, pulse width, energy per pulse, wavelength, number of sub-pulses, resolution, pulse repetition frequency, beam divergence, and receiver threshold.

- LAM trajectory and LADAR scan patterns—Typical missile trajectory parameters and sensor scan parameters include nominal altitude (above ground level), speed, depression angle, nod angle, nod frequency, yaw limits, yaw rates, and effective horizontal and vertical fields-of-view.
- MET Conditions and Obscuration—The atmospheric transmission is a function of the input visibility. The model also addresses obscuration due to battlefield smoke if it is present.
- Target type and orientation relative to the LADAR—The target orientation is defined by the target range, lookdown angle, and azimuth, as seen by the LADAR. The model also addresses target defilade.

Given that the target lies within the template, the simulation computes Pacq as follows:

- The signal-to-threshold of the target is computed based on

MET conditions, obscuration, range, and target reflectivity. These computations are presented in section 4.2 below.

- If S/T is greater than one, the number of laser returns on the target is computed as a function of the target type, orientation, range, and LADAR resolution. Laser returns (or pixels) on the target are computed by creating a pixel map of the target with the proper resolution and orientation. The POT is computed by summing the number of points in the pixel map that are on the target. Several example pixel maps are presented in section 4.3 below.

- Pacq is computed based on an empirical curve of Pacq vs. POT. This curve was computed assuming Pacq is a normal function of line pairs, and is approximately 75% when there are 8.3 line pairs on the target. A line pair is made up of two lines, so this requirement translates to approximately 275 pixels on target when Pacq is 75%. Pacq as a function of line pairs and POT are presented in section 4.4 below.

- A uniform random number is drawn and compared to Pacq to determine if the target is acquired.
- If the target is acquired, it is placed in a Target Report and processed by the mission planner to determine if and when the PAM will engage it. The Target Report includes target inertial location and acquisition time. An example Target Report is presented below in section 4.5.

4.2 Computation of Signal-to-Threshold (S/T)

The Signal-to-Threshold for laser return is computed as follows:

$$S/T = \frac{E_{des} (T_{rs}) (T_{rm}) \rho}{\pi \left(\frac{R_m}{1000} \right)^2 Thr_s}$$

Where:

E_{des}	Laser pulse energy (joules)
ρ	Target reflectivity (%)
T_{hrs}	Threshold energy at sensor dome for acquisition (joules/cm ²)
T_{rs}	Transmission due to smoke and obscuration (%)
T_{rm}	Transmission due to clear air (%)
R_m	Range from sensor to target (m)

Transmission due to smoke and obscuration (T_{rs}) is a function of a specified smoke type (defining the wavelength dependent extinction coefficient), and a specified concentration-path-length [5].

The atmospheric transmission is a function of the specified visual range as follows:

$\alpha_v = 3.912 / vis$ visual extinction coefficient

$\alpha_{1.06} = 0.73 \alpha_v^{1.16}$ extinction coefficient at 1.06 micron

$T_{rm} = e^{-\alpha_{1.06} R_m}$ atmospheric transmission (%)

Vis visual range (km)

When S/T < 1, Pacq = 0.0

When S/T > 1, Pacq = f(POT)

Where POT defines the number of pixels on the target. POT is a function of the following parameters;

1. Target length, width and height (i.e., target type)
2. Target orientation relative to the sensor (azimuth and declination angle)
3. LADAR resolution
4. Range from the sensor to the target

4.3 Computation of Pixels on Target (POT)

This approach of associating POT to LADAR ATR performance is recognized in the work of Perona, et al. [6], where POT was used as a criterion for selection of correlation filters for ATR. Application of the approach to simulate LAM sensor performance gives a relative performance measure for evaluating system trades such as missile flight profile, sweep rate, and lookdown angle. However, this approach is not sufficiently complex to produce true ATR performance predictions. The work of Sims [7] describes more comprehensive approaches to ATR performance measurement based upon signal to clutter and knowledge base signature distortion.

Given that the measures of interest in the NLOS-LS experimentation are relative comparisons, the core of the LAM model is the computation of target acquisition based on S/T and POT. The stand-alone model uses the Target Signature Model (TSM) from the Laser Designator Weapon System Simulation to define the target shape, as a function of the target type and aspect. The TSM was modified to create a pixel map of the target as seen at the proper range, azimuth, and depression angle. TSM was employed as a Dynamic Link Library and the pixel resolution, target type, and aspect angles were inputs. The output consists of the number of pixels resolved on the target by the LADAR at the specified range and aspect. A file that includes numerous tactical targets defines the target type. Any target scanned by the LADAR will have a specific presented area as a function of the target type and orientation. The number of pixels resolved on the target is a function of the range from the LADAR to the target, the target size, and the resolution of the LADAR template. A typical LADAR might have a resolution of 0.4 milliradians and acquire targets at approximately 1 km. Laser returns from this LADAR would be separated by 0.4 meters at 1 km range. Therefore, a target board that is 4m by 4m would be resolved with 100 POT. There are two methods for computing POT in the LADAR model. One way is to compute the number of pixels resolved in the horizontal and vertical dimension based on the target dimensions, the range to the target, and the resolution of the LADAR. This is the method currently employed in IDEEAS. The second method is to employ the TSM code and compute a pixel map of the target

as a function of the target type, the target range and LADAR resolution. This is the method employed in the stand-alone LAM model. The Target type is specified via input and the target dimensions are obtained from the IDEEAS target file. The TSM code renders a pixel map of the specified target based on target range, and the resolution of the LADAR. Several examples of this process are presented in Figures 6 through 9. Figure 6 presents a pixel map of a front aspect of a T-72 tank at a 1 km range using a LADAR with a resolution of 0.4 milliradians. In this figure ones represent pixels, which are on the target, and zeros represent pixels that are off the target. At a 1 km range, one pixel will subtend an area 0.4 meters by 0.4 meters on the target. As seen in this figure, the target is 6 pixels (2.4 meters) high and 9 pixels (3.6 meters) wide, which approximately matches the dimensions of the T-72 target. A total of 50 pixels define the target. Figure 7 presents the side view of the same target. The target height is the same but the width is now 18 pixels (or 7.2 meters), and there are 88 pixels on the target. Figure 8 presents the same target at a 45-degree aspect and a 13-degree depression. The pixel map in this case contains a POT count of 108. Finally, Figure 9 presents the target at a 500 meter range. There are 426 pixels on target at this range. Notice that this is about a four-fold increase in POT at 1000 meters. If the LADAR resolution were doubled (to 2 mrad), the POT count would be 426 at 1000 meters.

For the purposes of 1stApp, and given the low level of feature resolution in the database that was used, reduction of POT by targets partially hiding behind clutter was not considered in the experiment. The ELE terrain format of the IDEEAS simulation, which is a gridded terrain, was not of sufficient fidelity to adequately simulate realistic target defilade. Current efforts to convert IDEEAS to a polygonal terrain format will make this a possibility in future work. However, it should be noted that in most distributed simulations, occlusion or partial obscuration of targets is only calculated when hiding behind terrain polygons, and not for hiding behind other simulated objects such as friendly, enemy, or civilian vehicles. This is a potential area for future work in target acquisition modeling.

For current analytical purposes, individual cases of partial obscuration are calculated using the POT method as described, but done so in terms of degree of defilade (partial, hull, full). These traditional defilade terms lose their meaning with airborne sensors such as LAM, and do not represent explicit and dynamic obscuration for particular target detections and engagements. Consequently, this function was not used during 1stApp.

4.4 Computation of Probability-of-Acquisition

Historically, Probability-of-Acquisition (Pacq) is based upon the number of line pairs needed for a human to resolve the various acquisition levels (see Johnson [8]). The line pairs and critical target dimension employed as the Johnson

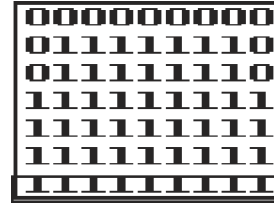


Figure 6. T-72 at 1 km, front view: 9 rows, 7 columns, 50 POT

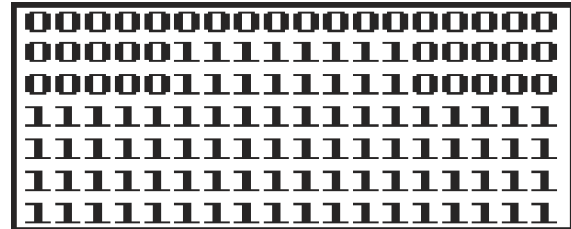


Figure 7. T-72 at 1 km, side view: 7 rows, 18 Columns, 88 POT

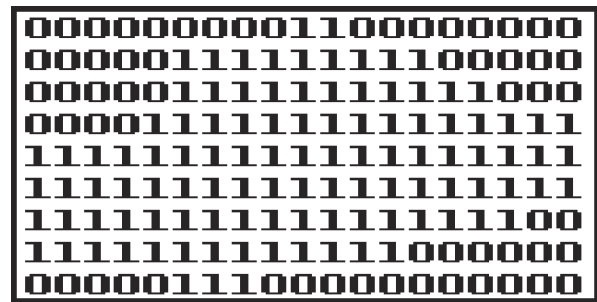


Figure 8. T-72 at 1 km, 45 deg aspect, 13 deg depression: 9 rows, 19 columns, 108 pixels

Criteria were extended by Alongi, et al. [9], to missile seeker performance with a methodology validated by the Army Materiel Systems Analysis Activity. Work in this area since Johnson has continued to support the principle of target acquisition as a function of physical dimensions of target presented area.

Pacq is a function of the performance of the ATA employed by the LAM. If S/T is greater than one, we can assume a sufficient signal-to-noise (S/N). Therefore, the significant parameter driving system performance is POT. If POT is very low, there is not enough range information in the target image to actually distinguish one target from another, although potential targets can be detected with high probability (particularly on relatively flat terrain). However, as POT increases, more range information is available defining the actual size and shape of the target. Therefore, as POT becomes very large, Pacq will approach one. Based on a review of data from existing LADAR programs, we assume as a rule-of-thumb that when POT is approximately 275, Pacq will be approximately 75%. Probability-of-Acquisition

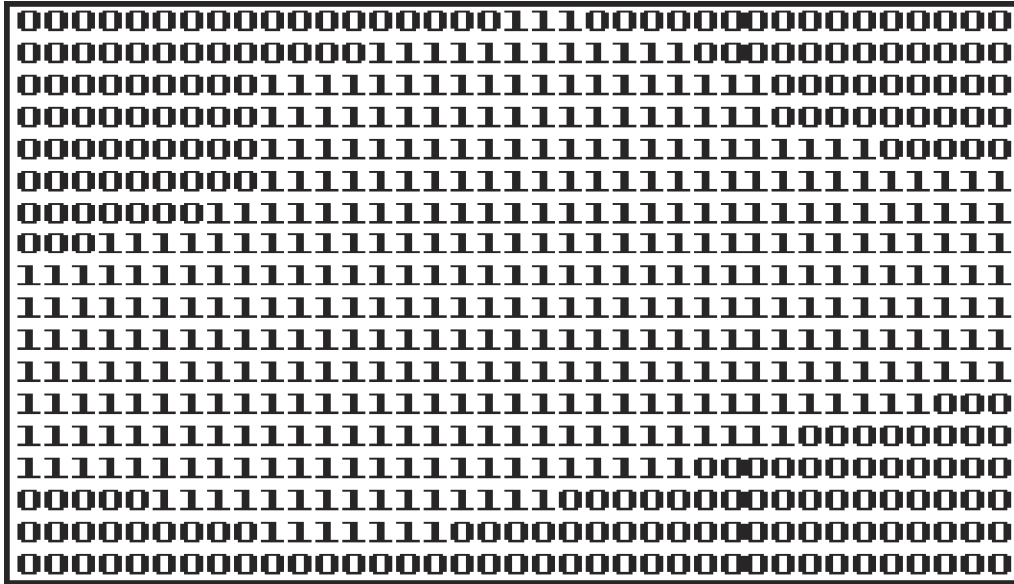


Figure 9. T-72 at 0.5 km, 45 deg aspect, 13 deg depression: 18 rows, 37 columns, 426 pixels

in an imaging system has traditionally been assumed to be normally distributed based on line pairs across the target critical dimension. Line-pairs can be related to pixels in an image because one row of pixels is required to form one line in an image. Therefore, two rows of pixels will form a line pair in either the horizontal or vertical direction. So if an imaging system (with high S/N) resolves four line pairs on a target that is approximately square shaped, it will resolve 8 lines and 64 pixels. Following this logic, 275 pixels equates to approximately 8.3 line pairs. Figure 10 presents a normal probability curve as a function of line pairs, where the 50% probability point requires approximately 6.5 line pairs. The 75% probability point requires approximately 8.3 line pairs. This curve was generated as a function of pixels and plotted in Figure 11. This probability curve is currently employed in the LADAR model. Notice that since we assume a normal distribution as a function of line pairs, the resulting Pacq vs. POT curve is not normal, but appears more lognormal.

All of these calculations assume a low clutter environment. In order to truly model the effects of clutter on ATR performance, a more robust algorithmic representation would be required of a specific ATR, which could then be compared against a confusion matrix of targets, false targets, and partially obscured targets. Given the current immaturity of LADAR ATR, test data is not yet available to accurately portray the effects of clutter on target acquisition.

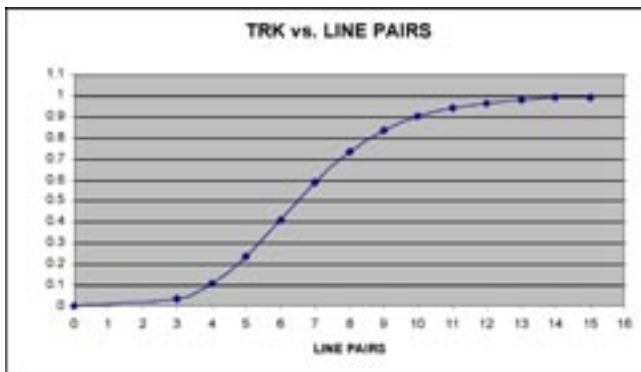


Figure 10. TRK=Probability-of-Acquisition as a function of line pairs

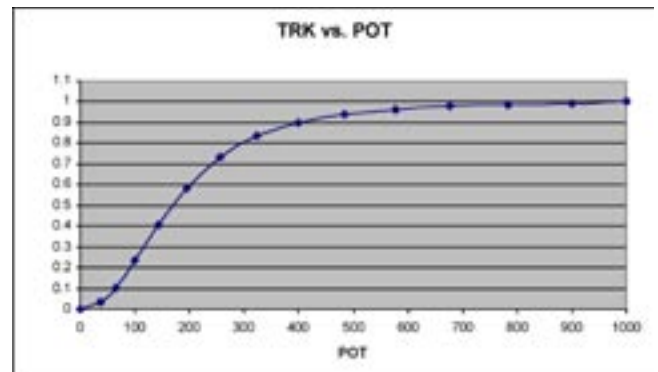


Figure 11. TRK=Probability-of-Acquisition as a function of Pixels on Target

4.5 Model Output

Table 1 presents an example of some output values that are generated by the model before the LAM flight is executed.

Variable	Value	Units	Definition
HFOV	28.0	Deg	Total yaw angle
VFOV	3.4	Deg	Nod angle
EPRF	156100.0	hrtz	Effective PRF (laser PRF times number of sub-pulses)
PPV	148.35	—	Number of pulse returns in the vertical dimension
PPH	1221.7	—	Number of pulse returns in horizontal dimension
PPTMP	181247.4	—	Total pulse returns in one template (PPVxPPH)
TFLTMP	1.56	sec	Scan time to define one template
TSTEP	0.4647	sec	Time interval between end of one template and start of next
NSUB	7	—	Number of sub-pulses per laser pulse
NNOD	174.53	—	Number of nods per template
TNOD	8.9624682E-03	sec	Time interval for one nod (reciprocal of nod frequency)
SCNEFF	0.7423	—	LADAR efficiency while generating a template
SCNEFF1	0.5723	—	LADAR efficiency including TSTEP
SCAN EFFICIENCY	—	—	SCNEFF1<1
MAXIMUM RANGE SOT	2.7193	km	Range where S/T = 1
MAXIMUM RANGE	1.1483	km	Range to top of template
MID RANGE TO GROUND	1.000	km	Range to middle of template
MIN RANGE TO GROUND	0.8867	km	Range to bottom of template
GROUND RANGE MID of TEMPLATE	0.7419	km	Ground range to template center
BEAM DIA AT RMAXDEP	0.2140	meters	Pulse diameter at max range
SPOT SEPARATION AT RMAXDEP	0.4593	meters	Pulse separation at max range
DEPTH	268.37	meters	Template depth downrange
DIS	202.80	meters	Distance missile moves between templates
DEL	65.47	meters	Overlap between templates
LADAR RANGE TO TOP OF TEMPLATE	—	—	Comment
BEAM SMALLER THAN RESOLUTION	—	—	Comment

Table 1. Model pre-flight parameters

The output values presented in Table 1 are computed once at the beginning of the simulation run to verify that the inputs defining the LADAR parameters, the missile parameters, and the MET conditions define a reasonable LAM configuration. The model will terminate if the following conditions are not met:

1. $S/T < 1$ at the maximum range (to the top of the template)
2. Beam diameter at max range $>$ pulse separation at max range.
3. Scan efficiency is $> 100\%$

The model will run even if DEL is a negative value but there will be gaps in the terrain coverage. In the above example, the LADAR parameters scan parameters and missile trajectory will combine to generate a scan pattern on the ground as depicted in Figure 1.

Several other output files are created during a simulation execution, including the following:

1. The input file is mirrored to an output file.
2. The extinction coefficients for all the sensor models is output so the user can verify that the MET conditions and battlefield obscuration is input correctly.
3. The LAM Target Report is output following final processing to “Deconflict” the targets.
4. Several PAM related output files are output. These files will be documented in a separate report.

Summarizing, the Netfires/LAM simulation can be (and has been) used to perform trade studies between the LAM trajectory, the LADAR parameters, and the Scan parameters. LAM capability to search the terrain at a given rate is a function of the missile speed and the template size. However, the LAM capability to acquire targets when they lie within the template is a function of the MET conditions, the laser power, the target reflectivity, and the POT (where POT is a function of the LADAR resolution, target range, target size, and aspect).

5. Conclusions

Representation of NLOS-LS in distributed virtual environments presents several new challenges which must be met to conduct credible analysis of future combat. The first of these challenges is the modeling of loitering munitions,

particularly using LADAR imagery with ATR. The second is the challenge of fully integrated networked fires, with all the complexities of using NLOS-LS in a system-of-systems context.

To address these challenges, AMRDEC has developed and employed a functional methodology for representing the LADAR seeker in the Loitering Attack Missile in the context of networked fires executed in a distributed virtual environment. The use of physics-based simulation tools in the 1stApp supports credible representation of complex system behavior, and establishes a baseline for future simulation efforts within the RDECOM.

Future work in this area should involve more explicit representation of tracker and ATR functions against photorealistic imagery, and against targets in realistic clutter—either through tactical or prototype software or hardware in the loop or through the development and implementation of more rigorous ATR abstractions. This level of fidelity will be required to conduct technology trade-offs of target acquisition performance, effects of enemy countermeasures, battlefield obscuration, and adverse weather.

6. References

- [1] Lorenzo, M., et al. 2002. Network Challenges and Observations Noted During an HLA Distributed Simulation Exercise. Simulation Interoperability Workshop (SIW), Paper 02S-SIW-059.
- [2] Bentley, T., et al. 2002. HLA Lessons Learned from the RDEC Federation Calibration Experiment. SIW, Paper 02S-SIW-042.
- [3] Birmingham, J., et al. 2003. The Architectural Design, Implementation and Observations of a Network, Built over the DREN, in Support of the Distributed HLA Simulation Federation of the RDE 1st Application. SIW, Paper 03F-SIW-029.
- [4] Weber, R., et al. 2003. The Many Lessons from the RDE Command 1st Application—But Did We Learn Anything? SIW, Paper 03F-SIW-005.
- [5] Military Handbook. 1986. Quantitative Description of Obscuration Factors for Electro-Optical and Millimeter Wave Systems. Metric DOD-HDBK-178(ER).
- [6] Perona, M. T., A. Mahalanobis, K. N. Zachery. 2000. System Level Evaluation of LADAR ATR using Correlation Filters. Automatic Target Recognition X, *Proceedings of SPIE*, Volume 4050.
- [7] Sims, S. R. F. 2000. Putting ATR Performance on an Equal Basis—The Measurement of Knowledge Base Distortion and Relevant Clutter. Automatic Target Recognition X, *Proceedings of SPIE*, Volume 4050.
- [8] Johnson, J. 1958. Analysis of Image Forming Systems. *Proceedings of the Image Intensifier Symposium*.
- [9] Alongi, R. E., W. Greenleaf. 1992. Battlefield Environment Weapon System Simulation Team Report Target Search Algorithms. U.S. Army TR RD-GC-92-20.