The Expected Results Method for Data Verification

Paul Monday
Lockheed Martin
Radcliff, KY
Paul.W.Monday@lmco.com

ABSTRACT

The credibility of US Army analytical experiments using distributed simulation depends on the quality of the simulation, the pedigree of the input data, and the appropriateness of the simulation system to the problem. The second of these factors is best met by using classified performance data from the Army Materiel Systems Analysis Activity (AMSAA) for essential battlefield behaviors, like sensors, weapon fire, and damage assessment.

Until recently, using classified data has been a time-consuming and expensive endeavor: it requires significant technical expertise to load, and it is difficult to verify that it works correctly. Fortunately, new capabilities, tools, and processes are available that greatly reduce these costs. This paper will discuss these developments, a new method to verify that all of the components are configured and operate properly, and the application to recent Army Capabilities Integration Center (ARCIC) experiments.

Three recent developments have focused improving the process to load the data. OneSAF has redesigned their input data file formats and structures so that they correspond exactly with the Standard File Format (SFF) defined by AMSAA, ARCIC developed a library of supporting configurations that correlate directly to the AMSAA nomenclature, and the Entity Validation Tool was designed to quickly execute the essential models with a test-jig approach to identify problems with the loaded data.

The missing part of the process is provided by the new Expected Results Method. Instead of the usual subjective assessment of quality, e.g., “It looks about right to me”, this new approach compares the performance of a combat model with authoritative expectations to quickly verify that the model, data, and simulation are all working correctly.

Integrated together, these developments now make it possible to use AMSAA classified performance data with minimal time and maximum assurance that the experiment's analytical results will be of the highest quality possible.

ABOUT THE AUTHOR

Paul Monday is the Lead Engineer for Lockheed Martin’s BLCSE Engineering Support Team. He has been working with OneSAF, OneSAF TestBed, ModSAF, JSAF, virtual simulators, data analysis, and other simulation systems for over 25 years, starting with the SIMNET-D program at Ft Knox. He received a Master of Science degree in Geophysics from Stanford University, and is a Certified Modeling and Simulation Professional (CMSP).

The Expected Results Method for Data Verification

Paul Monday
Lockheed Martin
Radcliff, KY
Paul.W.Monday@lmco.com

INTRODUCTION

As long as mankind has been fighting wars, simulation has been used to analyze military systems and concepts. Computer-based simulations date to the days when a computer bug was an actual insect. Equally longstanding is the question of whether simulations can replicate reality. There are five key factors that contribute to the credibility of computer simulation (Muessig, 2001):

1. Capability – Does the simulation model the key elements of the battlefield to support the subsequent analysis?
2. Software Accuracy – Does the software emulate the key models (physical, behavioral, environmental)?
3. Data Accuracy – Is the input data appropriate for the models and relevant to the scenario?
4. Results Accuracy – Do the results of the simulation correlate with the real world?
5. Usability – Can the simulation be employed in an error-free manner?

While Battle Lab Collaborative Simulation Environment (BLCSE) simulation experiments encompass a wide range of key Capabilities, the most critical are the sensor models that determine when one entity acquires another, the weapon accuracy models that determine when an entity is struck, and the vulnerability models that assess the effects of the weapon fire. For Army simulations, the gold standard of combat models are those that use algorithms from the Army Materiel Systems Analysis Activity (AMSAA). AMSAA defines algorithms that are applicable to entity-level simulations, and provides data for that incorporates the specifics of a particular scenario, including time period, availability of equipment, and likely employment. AMSAA verifies that their data, when used in their algorithms, will produce credible results for a particular scenario. That is, AMSAA data and algorithms support steps 2, 3 & 4 from the list above.

There are several tasks for using classified AMSAA data (Monday, 2009). Once the data is provided, it must be loaded into the simulation. Each AMSAA file is formatted as required to the appropriate input area, so the simulation can then read and make use of the data, assuming that it has already been configured to understand the data's contents. To enable this, each of the names that AMSAA uses for entities, weapons, munitions, etc. must be defined in the simulation, and information related to each name must be provided. Further, each entity must be configured in the simulation with the appropriate sensors, weapons, and munitions. Finally, the analyst must verify that the simulation handles the new data correctly, in accordance with scenario requirements (Sargent, 2013).

Performing these tasks has been time consuming and has required specialized knowledge. However, the situation has improved dramatically in recent years due to the introduction of new technologies. First, One Semi-Automated Forces (OneSAF) has adopted the same format for its data files that AMSAA uses to provide the data. This means that reformatting AMSAA data is not needed. Second, ARCIC has developed a library of auxiliary data that supports the classified data. Third, ARCIC has created a tool to ensure that the data is consistent and complete. Finally, ARCIC has developed an automated method for verifying that the data operates in the simulation exactly the way that AMSAA expects.

BACKGROUND

Battle Lab Collaborative Simulation Environment

The BLCSE is a distributed and collaborative M&S environment that enables Concept and Capabilities Development across the Army and TRADOC. It consists of a persistent and secure network enabling collaboration and interoperability across several Army and TRADOC organizations; a collection of constructive and virtual models and simulations with supporting functional interoperability, event management, and data collection and analysis tools; an accessible repository that provides certified scenarios, data, standards, and procedures; and video teleconferencing, white board capability, and voice over internet protocol (VOIP) communications. The BLCSE currently connects over 20 sites via the Defense Research and Engineering Network (DREN). The BLCSE federation is a suite of multiple and integrated simulations currently connected via the procedures of the High Level Architecture (HLA) standard Institute of Electrical and Electronics Engineers (IEEE)-1516; and Enumerations and protocol data units (PDUs) under procedures of the Distributed Interactive Simulation (DIS) standard IEEE-1278.

The BLCSE is used primarily by ARCIC and the TRADOC Capability Development and Integration Directorates and Battle Labs to support distributed experiments (events). Army experimentation is defined as immersing Soldiers and Leaders within live, virtual and constructive (LVC) environments that explore concepts and assess capability needs and solutions across Doctrine, Organization, Training, Materiel, Leadership & Education, Personnel, and Facilities (DOTMLPF) in order to learn and mitigate risk for current and future forces.

One Semi-Automated Forces

One Semi-Automated Forces (OneSAF) is a composable, entity-level Computer Generated Forces (CGF) simulation designed for brigade and below, combat and non-combat operations (Logsdon & Wittman, 2007). Being a SAF model, it provides intelligent, doctrinally correct behaviors to increase the span of control for workstation operators. It was built to represent the modular and future force and to represent entities, units, and behaviors across the spectrum of military operations in the contemporary operating environment. OneSAF is unique in its ability to simulate unit behaviors from fire team to company level for all units.

Other Key Simulations

Other entity-producing simulation federates in the BLCSE include Fire Simulation XXI (indirect fire), Advanced Tactical Combat Model (rotary wing, air defense), Extended Air Defense Simulation (air defense, air surveillance), Air Warfare Simulation (fixed wing), Joint Deployment Logistics Model (logistics), and Comprehensive Minefield Simulation (mines).

Damage Effects Server

The BLCSE federation uses a central Damage Effects Server (DES) that computes combat effects for all munition detonations in the exercise. The DES is based on OneSAF, and it uses OneSAF data files and vulnerability models. This approach greatly simplifies the configuration and troubleshooting of vulnerability data for a widely-distributed BLCSE simulation system. It also means that all of the data and vulnerability models in the DES must function correctly.

STANDARD FILE FORMAT

AMSAA provides data in Standard File Formats (SFF). Typically colon-separated, the format defines the column names and values for each data type. For example, the ART_HE_AL file (Figure 1) contains the lethal area for each combination of munition, target, angle of fall, and target's state. Strict adherence by AMSAA to the SFF standard, and support by OneSAF for SFF input files (Barnett, 2009) means that the user must merely copy AMSAA files to the simulation's data area. It is now possible to import a complete data release in a single day.

```
Date:Munition:Target:Target Exposure:Target Posture:Target State:
Kill:Target Environment:Target Terrain:Fuze Type:Angle of Fall:
Lethal Area:Classification:Certification Level:Data Pedigree:Row Id
8/15/2005:M107:M1A1D:FE:U/A:U/A:F:OPEN:FLAT:PD:20:80:U:A::68101
8/15/2005:M107:M1A1D:FE:U/A:U/A:F:OPEN:FLAT:PD:35:80:U:A::68102
8/15/2005:M107:M1A1D:FE:U/A:U/A:F:OPEN:FLAT:PD:56:80:U:A::68103
8/15/2005:M107:M1A1D:FE:U/A:U/A:K:OPEN:FLAT:PD:20:10:U:A::68104
8/15/2005:M107:M1A1D:FE:U/A:U/A:K:OPEN:FLAT:PD:35:10:U:A::68105
8/15/2005:M107:M1A1D:FE:U/A:U/A:K:OPEN:FLAT:PD:56:10:U:A::68106
8/15/2005:M107:M1A1D:FE:U/A:U/A:M:OPEN:FLAT:PD:20:80:U:A::68107
8/15/2005:M107:M1A1D:FE:U/A:U/A:M:OPEN:FLAT:PD:35:70:U:A::68108
8/15/2005:M107:M1A1D:FE:U/A:U/A:M:OPEN:FLAT:PD:56:65:U:A::68109
8/15/2005:M107:M1A1D:FE:U/A:U/A:M/F:OPEN:FLAT:PD:20:95:U:A::68110
8/15/2005:M107:M1A1D:FE:U/A:U/A:M/F:OPEN:FLAT:PD:35:80:U:A::68111
8/15/2005:M107:M1A1D:FE:U/A:U/A:M/F:OPEN:FLAT:PD:56:100:U:A::68112
8/15/2005:M107:BRDM2:FE:U/A:U/A:F:OPEN:FLAT:PD:20:81:U:A::1123
8/15/2005:M107:BRDM2:FE:U/A:U/A:F:OPEN:FLAT:PD:35:80:U:A::1124
8/15/2005:M107:BRDM2:FE:U/A:U/A:F:OPEN:FLAT:PD:56:97:U:A::1125
8/15/2005:M107:BRDM2:FE:U/A:U/A:K:OPEN:FLAT:PD:20:39:U:A::1126
```

Figure 1. Lethal area vulnerability data in SFF format

LIBRARY OF SUPPORT DATA

While the typical AMSAA data release includes a wide variety of data, it is not nearly complete. Support information is essential for a simulation like OneSAF to ingest the data and to use it. Creating the support information for OneSAF is actually the biggest job by far when preparing a classified data load for the first time. By placing this into a reusable library, ARCIC has laid the foundation for rapid data input.

There are two primary types of support data in the library: Enumerations and Compositions. Enumerations are used in OneSAF to internally represent discrete values like entity type, munition type, and posture. They mostly correspond to AMSAA's Standard Nomenclature Database (SND). The SND values are used in the data to describe entities (M1A2, BMP2), munitions (M392A2, M80), mounts (turret, pintle), etc. Each SND value that is used in the data must be defined as an enumeration in OneSAF before it can successfully load the data.

Besides the values for entities and weapons in the Data Request, many other SND values must be supported because of AMSAA's internal surrogation process. They use surrogation when the systems are very similar, or when data for the requested system isn't available. The result of this approach is that a data request for 200 entities might include those values, plus another 500 SND values for the surrogated systems.

The other primary type of support data in the library are the composition files for entities, sensors, and weapons. The entity composition defines the items which comprise the entity, like the specific sensors, weapons, basic load, sensor controllers, mobility agents, etc. While PM-OneSAF supplies baseline entity and component compositions, these cannot be used for classified work since they do not incorporate the correct components. In fact, the complete definition of an entity's components can become classified, even before any AMSAA data is loaded. Further, the names of some systems and weapons are themselves classified, so we maintain the Library as classified data.

ENTITY VALIDATION TOOL

Loading the data is easy, determining that it is being handled properly by the simulation is harder. The Entity Validation (EV) Tool is the basic tool to ensure that OneSAF reads the data properly, and identifies gaps and other data integrity issues.

The EV Tool is a component of OneSAF that displays information for selected entities which includes general descriptive data, performance analysis of selected combat models, and checks of data validity. It uses regular OneSAF models to create entities, sensors, weapons, etc., but in an automated manner. This test-jig approach means that each analysis considers all parts of the OneSAF system, not just the data itself. It is typically used to check the entities in a particular scenario, and can complete its analyses in just a few minutes. The rapid execution time enables a productive check-fix-recheck cycle.

Composition Parameters

Understanding the various parameters that are configured for a particular entity is very difficult in OneSAF because of the way that parameters are distributed across many files. Each entity is composed with multiple component composition files (sensors, weapons), most performance characteristics are located in model-specific files, and few users can make much sense of the xml data. The EV Tool brings all of this data into a single easy-to-understand HTML file (Figure 2) for each entity. It includes size, weight, basic load, and a list of the entity's components.

IFV_M2A3_OIF_Bradley_Infantry.xml

Composition

- Name: /PAIR/compositions/entity/mr/BLCSE/Blue/IFV_M2A3_OIF_Bradley_Infantry.xml
- Country: US
- EntityType: IFVBradleyM2A3_entity
- EntityClass net.onesaf.core.services.odm.entity.GroundVehicle

BasicLoad

Name	Quantity	Qty/Container	Wt/Container
atgmTOW2B_munition	3.0	1	40.0
25MM/APDS_munition	330.0	30	0.8666667
25MM/HEI-T_munition	300.0	30	0.8666667
M59_munition	4400.0	800	0.045
chemicalProtectiveSuit	3.0	1	0.453
fuelJP8Bulk	0.665	1	814.81
grenadeScreeningSmokeL8A1_munition	6.0	4	1.5
grenadeScreeningSmokeL8A3_munition	6.0	4	1.5

Physical Component: CdrI2:M35M36NightSight

- Description: component/physical/sensor/mr/BLCSE/NightSight_M35M36_Sensor.xml
- Fidelity: MEDIUM
- MaxSensorRange: 11359.405517578125
- Scanning
- FOV: wide 7.200000000000001 x 7.200000000000001
- Magnification: 7.0
- MaxRange: 11359.405517578125
- Name: M35M36NightSight_sensor
- Sensor: M35M36NightSight_sensor
- Type: NONE
- FOVWidth: 0.12566370614359174
- FOVMaxRange: 0.762939453125
- MaxRangeLifeform (ICFullyLoaded_entity): 0.762939453125
- MaxRangeGroundVehicle (M1A1_entity): 0.762939453125
- MaxRangeAircraft (AH-64D_entity): 0.762939453125
- MaxEffectiveRange

Condition	IC	GV	AV
Day	647	4763	10622
Night	368	3075	7958

Figure 2 Entity summary description

Sensor Analysis

The performance of a sensor is impossible for a mortal to ascertain from the raw AMSAA performance data, which includes such items as contrast, spatial frequency, and characteristic dimension. The EV Tool empirically calculates the performance of each sensor versus each target at selected ranges and illuminations by creating a sensor and multiple targets and then exercising the model many times. The results (Figure 3) can be examined for obvious outliers (like a Direct-View-Optic not sensing in daylight), and to understand the effects on the battlefield.

Firer	Target	Sensor	Day	Night
M1A2	T72S	ThermalSight	ID(50-2000) Rec(2000-2500) Det(2500-5000)	ID(50-1500) Rec(1500-2000) Det(2000-5000)
M1A2	T72S	CdrLdr_VisionBlock	ID(50-1500) Rec(1500)	ID(50-200) Rec(200)
M1A2	T72S	Driver_II		ID(50-200)
M1A2	T72S	Driver_VisionBlock	ID(50-1500) Rec(1500)	ID(50-200) Rec(200)
M1A2	T72S	DaySight	ID(50-3000)	ID(50-1500) Rec(1500-2000) Det(2000-2500)
M1A2	IC-AK74	ThermalSight	ID(50-1000) Det(1000)	ID(50-1000) Rec(1000)
M1A2	IC-AK74	CdrLdr_VisionBlock	ID(50-200)	
M1A2	IC-AK74	Driver_II		ID(50-100)
M1A2	IC-AK74	Driver_VisionBlock	ID(50-200)	
M1A2	IC-AK74	DaySight	ID(50-1500)	ID(50-200)
BMP2	M1A2	NightSight_M35M36		ID(50-1500) Rec(1500-2000) Det(2000)
BMP2	M1A2	CdrLdr_VisionBlock	ID(50-1500) Rec(1500)	ID(50-200) Rec(200)
BMP2	M1A2	Driver_II		ID(50-200)
BMP2	M1A2	Driver_VisionBlock	ID(50-1500) Rec(1500)	ID(50-200) Rec(200)
BMP2	M1A2	Missile_Helo_I2		ID(50-2000)
BMP2	M1A2	DaySight_APC_Missile	ID(50-4000)	ID(50-1500) Det(1500-2000)

Figure 3 Analysis of sensor performance

Weapon Analysis

The accuracy of a weapon is just as difficult to estimate using the AMSAA bias-and-dispersion data. The EV Tool empirically determines the probability of hit by creating a weapon and firing it many times at a NATO standard target. The table (Figure 4) can be evaluated by SMEs to ensure that the data is reasonable.

Firer	Mount	Weapon	Munition	Range	FR?	LoadTime	LayTime	Ph	RMS	ShotToHit	Result
M88A2	M2HB	pintle	M20	50	T	0	1	86	0.78	1	Hit
M88A2	M2HB	pintle	M20	100	T	0	1	62	1.62	1	Hit
M88A2	M2HB	pintle	M20	200	T	0	1	57	5.75	1	Hit
M88A2	M2HB	pintle	M20	1000	T	0	1	41	129.39	2	Miss
M88A2	M2HB	pintle	M20	1500	T	0	1	28	228.71	3	Miss
M88A2	M2HB	pintle	M20	2000	T	0	1	35	277.02	2	Miss
M88A2	M2HB	pintle	M20	2500	T	0	1	27	279.64	3	Miss
M1135	M240	pintle	M59	50	T	0	1	98	0.71	1	Hit
M1135	M240	pintle	M59	100	T	0	1	82	0.82	1	Hit
M1135	M240	pintle	M59	200	T	0	1	54	3.58	1	Hit
M1135	M240	pintle	M59	1000	T	0	1	46	54.96	2	Miss

Figure 4 Analysis of weapon accuracy

Vulnerability Analysis

The vulnerability analysis empirically determines the probability of kill by firing at a target and then executing the OneSAF vulnerability models to assess the damage. This process is repeated many times and the results are averaged. The analysis data (Figure 5) can help the users to understand the expected performance of the simulation. The table shows Pk by level (K-Catastrophic, I-Incapacitated, MF-Mobility&Firepower, F-Firepower, M-Mobility, N-NoKill).

Firer	Weapon	Mount	Munition	Target	Range	I2TRange	K	I	MF	F	M	N
M1A2/SEP	M256	turret	M829A1	T-72A	100	0	39	0	16	12	12	21
M1A2/SEP	M256	turret	M829A1	T-72A	1000	0	45	0	17	11	4	23
M1A2/SEP	M256	turret	M829A1	T-72A	2000	0	34	0	12	14	10	30
M1A2/SEP	M256	turret	M829A1	T-72A	3000	0	40	0	6	13	12	29
M109A1	M284	turret	M549A1	IC	50	0	0	100	0	0	0	0
M109A1	M284	turret	M549A1	IC	50	1	0	100	0	0	0	0
M109A1	M284	turret	M549A1	IC	50	5	0	100	0	0	0	0
M109A1	M284	turret	M549A1	IC	50	10	0	98	0	0	0	2
M109A1	M284	turret	M549A1	IC	50	50	0	8	0	0	0	92
M109A1	M284	turret	M549A1	IC	50	100	0	0	0	0	0	100

Figure 5 Analysis of vulnerability

Rules of Engagement Analysis

This analysis displays which munition that OneSAF will use against a particular target by range. The rules are based on AMSAA data. The table (Figure 6) can be used to troubleshoot problems in the simulation.

Firer	Target	Munition
M2A3	T72S	25MM/APDS(50-100) TOW2B(100-4000)
M2A3	IC-AK74	M59(50-1500) 25MM/HEI-T(1500-3000)
Stryker	T72S	M20(50-200)
Stryker	IC-AK74	M20(50-2500)
M1A2SEP	T72S	M59(50-1000) M829A1(1000-5000)
M1A2SEP	IC-AK74	M20(50-2500)
AH64	T72S	30mmHEIM799(50-2500)
AH64	IC-AK74	30mmHEIM799(50-2500)
T72S	M1A2SEP	125mmAPFSDS(50-3000)
T72S	IC-M4	762mmNFI(50-1500) 12.7MM/API-T(1500-2500)
BMP2	M1A2SEP	30MM/APDS(50-100) AT-5(100-5000)
BMP2	IC-M4	762mmNFI(50-1500) 30MM/HEI(1500-3000)

Figure 6 Analysis of rules of engagement

Data Validity Analysis

The data validity analysis verifies a variety of requirements. For example, each target must have contrast values in order to be acquired by a sensor. These checks (Figure 7) ensure that each entity will perform basic battlefield actions, like sensing, shooting, moving, and being killed.

Message	Composition
No connector source for 191 in FOX_NBC_Recon	FOX_NBC_Recon
Duplicate connector 19 in BunkerHill	BunkerHill
No data in ccttMobilityData for MaxxPro	MaxxPro
No data in TGTACQ_CONTRAST.dat for AH-64D (R_HELO) at Day:WINTER:SKY:OVERCAST:1	ApacheD
No fuel (NABulkPOLSupplyType) as specified in POL_Consumable.xls for AN/TPQ-50	AN/TPQ-50
Min/max effectRange in Class_V_Supplies for M1064:M934 should be 100.0-8200.0	M1064A3
No data in EntityType_to_CombatPower.xls M1064	M1064A3

Figure 7 Summary of data errors

EXPECTED RESULTS METHOD

Loading the data and verifying that the simulation operates in a reasonable way is as far as most people can go. The Expected Results Method takes the next step to verify that the model and data are operating correctly, by comparing the statistical results of a simulation model with an authoritative standard.

The method is based on the concept of Standard Situations. The situation includes all key criteria, like firer, munition, target, range, exposure, etc. Expected Results are calculated from raw AMSAA data for each Standard Situation. This is the authoritative standard. Then the simulation model is exercised for the same Standard Situations, and the results are statistically analyzed to produce the model's Actual Results. Finally, for each situation, the Expected Results are compared with the Actual Results. If they differ significantly, then the simulation model has a defect. That defect could be a flaw in the model's algorithms, in the model's handling of the data, or in the model's implementation of the situation.

Once the method is set up, it can be executed quickly with minimal manual input. The simulation developer can verify that the data is loaded properly, that the simulation is reading the data properly, and that the simulation models perform correctly, and he can re-verify his system as often as necessary.

The method has three parts: 1) calculating the authoritative values (Expected Results), 2) exercising the simulation (Actual Results), and 3) comparing the two sets of results. Steps 1 & 2 each produce spreadsheets with data for each combination of a pair and a situation. Step 3 compares one spreadsheet with the other.

Calculation of Expected Results

AMSAA is the standard source of classified performance data. Since they don't provide the Expected Results (ER), we use our own programs to calculate the ER data from the raw AMSAA data. These programs are not part of any simulation system, and have been calibrated using selected examples from AMSAA.

Raw AMSAA Data

AMSAA organizes its data into standard directories, like ARMOR, INFANTRY, ARTY_CONV. The ER programs use the data in that same structure.

Pairing List

The pairing list is provided by AMSAA, and describes each combination of firer/weapon/mount/munition and target. The ER programs use the pairing list to regulate which entities or combinations of entities will be analyzed.

Basic Data File

The Basic Data file defines the situations for the subsequent calculations. It specifies which variables will be matrixed to generate all of the Standard Situations. Parameters include:

- Firer to target ranges
- Impact to target ranges (for indirect-fire rounds)
- Aspect angles
- Target exposures
- Target postures (for individual combatants)
- Acquisition levels
- Light levels
- Clutters
- Target backgrounds
- Seasons

ER – Sensor

The ER-Sensor program reads raw AMSAA data, a pairing list, and a control file and then executes the Acquire-TTP sensor model against each of the pairs for each Standard Situation. The program calculates probability of acquisition given infinite time (p-infinity).

Observer	Sensor	Target	LLevel	Range	Exposure	Posture	Clutter	Background	Weather	Season	AcqLevel	P-Inf
M2A1	VBlock	2S12	10000	50	FE	STATIONARY	LOW	SAND	CLEAR	FALL	DETECTION	1
M2A1	VBlock	2S12	10000	100	FE	STATIONARY	LOW	SAND	CLEAR	FALL	DETECTION	1
M2A1	VBlock	2S12	10000	200	FE	STATIONARY	LOW	SAND	CLEAR	FALL	DETECTION	1
M2A1	VBlock	2S12	10000	1000	FE	STATIONARY	LOW	SAND	CLEAR	FALL	DETECTION	0.998986
M2A1	VBlock	2S12	10000	1500	FE	STATIONARY	LOW	SAND	CLEAR	FALL	DETECTION	0.967874
M2A1	VBlock	2S12	10000	2000	FE	STATIONARY	LOW	SAND	CLEAR	FALL	DETECTION	0.830642
M2A1	VBlock	2S12	10000	2500	FE	STATIONARY	LOW	SAND	CLEAR	FALL	DETECTION	0.599927
M2A1	VBlock	2S12	10000	3000	FE	STATIONARY	LOW	SAND	CLEAR	FALL	DETECTION	0.383924
M2A1	VBlock	2S12	10000	4000	FE	STATIONARY	LOW	SAND	CLEAR	FALL	DETECTION	0.145696
M2A1	VBlock	2S12	10000	5000	FE	STATIONARY	LOW	SAND	CLEAR	FALL	DETECTION	0.06021
M2A1	VBlock	2S12	10000	7500	FE	STATIONARY	LOW	SAND	CLEAR	FALL	DETECTION	0.010051
M2A1	VBlock	2S12	10000	9000	FE	STATIONARY	LOW	SAND	CLEAR	FALL	DETECTION	0.004153
M2A1	VBlock	2S12	10000	50	FE	STATIONARY	LOW	SAND	CLEAR	FALL	CLASSIFICATION	1
M2A1	VBlock	2S12	10000	100	FE	STATIONARY	LOW	SAND	CLEAR	FALL	CLASSIFICATION	1
M2A1	VBlock	2S12	10000	200	FE	STATIONARY	LOW	SAND	CLEAR	FALL	CLASSIFICATION	1
M2A1	VBlock	2S12	10000	1000	FE	STATIONARY	LOW	SAND	CLEAR	FALL	CLASSIFICATION	0.998986
M2A1	VBlock	2S12	10000	1500	FE	STATIONARY	LOW	SAND	CLEAR	FALL	CLASSIFICATION	0.967874
M2A1	VBlock	2S12	10000	2000	FE	STATIONARY	LOW	SAND	CLEAR	FALL	CLASSIFICATION	0.830642
M2A1	VBlock	2S12	10000	2500	FE	STATIONARY	LOW	SAND	CLEAR	FALL	CLASSIFICATION	0.599927
M2A1	VBlock	2S12	10000	3000	FE	STATIONARY	LOW	SAND	CLEAR	FALL	CLASSIFICATION	0.383924
M2A1	VBlock	2S12	10000	4000	FE	STATIONARY	LOW	SAND	CLEAR	FALL	CLASSIFICATION	0.145696

Figure 8 Expected sensor results

ER – Vulnerability

The ER-Vulnerability program is similar, but it executes several lethality models to calculate probability of kill for catastrophic, mobility and firepower, mobility only, etc.

The supported models include:

- Direct fire – ground vehicle
- Direct fire – individual combatant
- Direct fire – individual combatant, fragmentary munition
- Direct fire – ATGM munition
- Indirect fire – HE munition
- Indirect fire – HE munition, fixed wing target
- Indirect fire – ICM munition
- Indirect fire – ICM munition, fixed wing target
- Indirect fire – smart munition
- Conventional mine – ground vehicle
- Conventional mine – individual combatant
- Smart mine – ground vehicle

The Expected Results table (Figure 9) shows authoritative values for Pk by level (K-Catastrophic, I-Incapacitated, MF-Mobility&Firepower, F-Firepower, M-Mobility, N-NoKill).

Firer	Weapon	Mount	Munition	Target	F2TRge	I2TRge	N	M	F	MF	I	K	ER
M2A1	M242	turret	M792	2S12	50	0	0.6	0.1	0	0.2	0	0.1	N
M2A1	M242	turret	M792	2S12	100	0	0.6	0.1	0	0.2	0	0.1	N
M2A1	M242	turret	M792	2S12	200	0	0.6	0.1	0	0.2	0	0.1	N
M2A1	M242	turret	M792	2S12	1000	0	0.6	0.1	0	0.2	0	0.1	N
M2A1	M242	turret	M792	2S12	1500	0	0.6	0.1	0	0.2	0	0.1	N
M2A1	M242	turret	M792	2S12	2000	0	0.6	0.1	0	0.2	0	0.1	N
M2A1	TOW	turret	TOW2B	2S12	200	0	0.1	0.1	0.35	0.15	0	0.3	S
M2A1	TOW	turret	TOW2B	2S12	1000	0	0.1	0.1	0.35	0.15	0	0.3	S
M2A1	TOW	turret	TOW2B	2S12	2000	0	0.1	0.1	0.35	0.15	0	0.3	S
M2A1	TOW	turret	TOW2B	2S12	3000	0	0.1	0.1	0.35	0.15	0	0.3	S
IC-B	M4	prone	M193	IC-R	50	0	0.42	0	0	0	0.58	0	I
IC-B	M4	prone	M193	IC-R	100	0	0.44	0	0	0	0.56	0	I
IC-B	M4	prone	M193	IC-R	200	0	0.44	0	0	0	0.56	0	I
IC-B	M4	prone	M193	IC-R	1000	0	0.6	0	0	0	0.4	0	N
2S12	2S12	baseplate	OF843B	Stryker	50	0	0.2	0	0	0	0	0.8	K
2S12	2S12	baseplate	OF843B	Stryker	50	1	0.395	0.077	0.059	0.464	0	0.005	S
2S12	2S12	baseplate	OF843B	Stryker	50	5	0.999	0	0	0	0	0	NN
2S12	2S12	baseplate	OF843B	Stryker	50	10	1	0	0	0	0	0	NN
2S12	2S12	baseplate	OF843B	Stryker	50	50	1	0	0	0	0	0	NN
2S12	2S12	baseplate	OF843B	Stryker	50	100	1	0	0	0	0	0	NN

Figure 9 Expected vulnerability results

Calculation of Actual Results

The calculation of Actual Results must be performed separately for each simulation. While this can be accomplished by manually placing firers and targets on the battlefield to match each Standard Situation, it is far more efficient to develop an automated test-jig. For OneSAF, the Entity Validation Tool provides this role. Its controller can be configured with the same firers, munitions, and targets as specified in the pairing list. It also provides options for the various options that comprise the Standard Situations. Executing the tool typically takes only a few minutes to generate a spreadsheet with the appropriate layout.

The Actual Results table (Figure 10) shows the simulation's values for Pk by level.

Firer	Weapon	Mount	Munition	Target	F2TRge	I2TRge	N	M	F	MF	I	K	ER
M2A1	M242	turret	M792	2S12	50	0	0.596	0.105	0	0.198	0	0.101	N
M2A1	M242	turret	M792	2S12	100	0	0.583	0.103	0	0.211	0	0.103	N
M2A1	M242	turret	M792	2S12	200	0	0.598	0.102	0	0.196	0	0.105	N
M2A1	M242	turret	M792	2S12	1000	0	0.598	0.098	0	0.204	0	0.1	N
M2A1	M242	turret	M792	2S12	1500	0	0.607	0.1	0	0.197	0	0.096	N
M2A1	M242	turret	M792	2S12	2000	0	0.61	0.101	0	0.188	0	0.101	N
M2A1	TOW	turret	TOW2B	2S12	200	0	0.098	0.097	0.361	0.145	0	0.299	S
M2A1	TOW	turret	TOW2B	2S12	1000	0	0.098	0.103	0.345	0.152	0	0.303	S
M2A1	TOW	turret	TOW2B	2S12	2000	0	0.101	0.101	0.352	0.149	0	0.298	S
M2A1	TOW	turret	TOW2B	2S12	3000	0	0.1	0.101	0.347	0.154	0	0.298	S
IC-B	M4	prone	M193	IC-R	50	0	0.405	0	0	0	0.595	0	I
IC-B	M4	prone	M193	IC-R	100	0	0.446	0	0	0	0.554	0	I
IC-B	M4	prone	M193	IC-R	200	0	0.456	0	0	0	0.544	0	I
IC-B	M4	prone	M193	IC-R	1000	0	1	0	0	0	0	0	N
2S12	2S12	baseplate	OF843B	Stryker	50	0	0.209	0	0	0	0	0.791	K
2S12	2S12	baseplate	OF843B	Stryker	50	1	0.381	0.081	0.06	0.472	0	0.005	S
2S12	2S12	baseplate	OF843B	Stryker	50	5	1	0	0	0	0	0	NN
2S12	2S12	baseplate	OF843B	Stryker	50	10	1	0	0	0	0	0	NN
2S12	2S12	baseplate	OF843B	Stryker	50	50	1	0	0	0	0	0	NN
2S12	2S12	baseplate	OF843B	Stryker	50	100	1	0	0	0	0	0	NN

Figure 10 Actual vulnerability results

Comparison of Expected and Actual Results

The comparison of the spreadsheets is pretty simple. Find the corresponding rows from the expected and actual data, and then determine whether the Pks are "close enough". If the fractional difference between two corresponding Pks is greater than 0.1, then the program flags that situation for further examination. In the example, the simulation reported no effects for the M193->IC-R at 1000 meters, while the correct answer was 40% chance of incapacitation.

The Comparison table (Figure 11) shows the differences between the authoritative and the simulation's values for Pk by level. "Significant" differences are flagged by color.

Firer	Weapon	Mount	Munition	Target	F2TRge	I2TRge	N	M	F	MF	I	K	Reason
M2A1	M242	turret	M792	2S12	50	0	0.007	0.048	0	0.01	0	0.01	OK
M2A1	M242	turret	M792	2S12	100	0	0.028	0.029	0	0.052	0	0.029	OK
M2A1	M242	turret	M792	2S12	200	0	0.003	0.02	0	0.02	0	0.048	OK
M2A1	M242	turret	M792	2S12	1000	0	0.003	0.02	0	0.02	0	0	OK
M2A1	M242	turret	M792	2S12	1500	0	0.012	0	0	0.015	0	0.04	OK
M2A1	M242	turret	M792	2S12	2000	0	0.016	0.01	0	0.06	0	0.01	OK
M2A1	TOW	turret	TOW2B	2S12	200	0	0.02	0.03	0.03	0.033	0	0.003	OK
M2A1	TOW	turret	TOW2B	2S12	1000	0	0.02	0.029	0.014	0.013	0	0.01	OK
M2A1	TOW	turret	TOW2B	2S12	2000	0	0.01	0.01	0.006	0.007	0	0.007	OK
M2A1	TOW	turret	TOW2B	2S12	3000	0	0	0.01	0.009	0.026	0	0.007	OK
IC-B	M4	prone	M193	IC-R	50	0	0.036	0	0	0	0.025	0	OK
IC-B	M4	prone	M193	IC-R	100	0	0.013	0	0	0	0.011	0	OK
IC-B	M4	prone	M193	IC-R	200	0	0.035	0	0	0	0.029	0	OK
IC-B	M4	prone	M193	IC-R	1000	0	0.4	0	0	0	1	0	N differs
2S12	2S12	baseplate	OF843B	Stryker	50	0	0.043	0	0	0	0	0.011	OK
2S12	2S12	baseplate	OF843B	Stryker	50	1	0.035	0.049	0.017	0.017	0	0	OK
2S12	2S12	baseplate	OF843B	Stryker	50	5	0.001	0	0	0	0	0	OK
2S12	2S12	baseplate	OF843B	Stryker	50	10	0	0	0	0	0	0	OK
2S12	2S12	baseplate	OF843B	Stryker	50	50	0	0	0	0	0	0	OK
2S12	2S12	baseplate	OF843B	Stryker	50	100	0	0	0	0	0	0	OK

Figure 11 Comparison of Expected and Actual results

Calibration of analysis

The program has been calibrated by comparing selected results that were produced using AMSAA's internal analysis programs. A selection of pairs and situations was chosen to exercise all important aspects of the ER-Vulnerability program. For each combination, AMSAA SMEs calculated the Pks according to the requirements of the model in question. Then the pairs and situations were processed with the ER-Vulnerability program and the results were compared. The cause of each difference was identified and fixed. This doesn't mean that the ER-Vulnerability has been accredited by AMSAA, only that it has been shown to exactly duplicate AMSAA calculations for the chosen pairs.

Applying the ER Method

The Expected Results Method can be used to evaluate any simulation if the actual results of that simulation can be produced for the Standard Situations. This can be accomplished by manually placing firers and targets at the appropriate ranges (for vulnerability), causing the firer to shoot the prescribed munition many times, and analyzing the results to generate the required statistics. Or a test-jig can be built into the simulation to exercise the appropriate internal models in an automated way. In either case, once the Actual Results spreadsheet is generated, the ER's comparison program can be executed.

Benefits of the ER Method

The ER Method can speed the verification testing from months to days, while improving the completeness of the tests from a small fraction of possible pairings and situations to nearly 100%. More importantly, it can be applied by average users who lack in-depth understanding of the combat models, so it offers the path for wider usage of AMSAA classified data to analytical experiments.

APPLICATION TO ARCIC EXPERIMENTS

The Expected Results Method was applied in a limited way to the Unified Challenge 15 ARCIC Exercise, which occurred in June, 2015. It helped uncover several significant defects in OneSAF vulnerability models. For example, the OneSAF Expected Casualty model was hard-coded to expect certain discrete range values, and this expectation was met with the unclassified data shipped with OneSAF. Unfortunately, the classified data had different range values. The effect of this disparity was that the calculated Pks were accurate for some ranges and were wrong for others. Comparison with the authoritative results provided by the ER Method illustrated the problem.

This application of the ER Method for finding defects in the simulation models will be continued through 2015. Our experience so far has uncovered many previously-unknown problems which cause the simulation to fail in selected circumstances. For example, OneSAF does not handle the target status correctly (launcher raised/lowered), but this failure only affects certain target types. The ER Method's ability to analyze a wide variety of entities, munitions, and situations means that it can quickly identity fringe defects.

We expect to use the ER Method as an integral part of the process for preparing for the FY16 Exercise. After the AMSAA data is loaded, the ER Method will be used to ensure that the data operates correctly for all pairs and situations. Then, the ER Method will be applied to each subsequent software release to verify that the simulation's integrity has not been inadvertently compromised. Given that the data set will probably have about 150 distinct entity types for both Blue and Red, checking 20,000 firer-target pairings in an automated way will be a huge accomplishment.

Meanwhile, the usual Data Verification testing that is performed as part of every ARCIC exercise will be streamlined. Instead of attempting to check all pairs by manually placing firers and targets, the manpower-intensive process will be used to spot-check the ER Method itself. We believe that the outcome of this application of the ER Method to the FY16 Exercise will be reduced costs and timelines, and much higher quality.

CONCLUSION

Using computer simulation for analysis depends on the credibility of the simulation and its execution. A variety of factors contribute to this credibility, and the most important revolve around the data and models that support the key battlefield models like sensors, weapon fire, and vulnerability. AMSAA provides algorithms and data for entity-level simulations that have wide acceptance in the Army analytical community, but there are several hurdles to actually using AMSAA data. Classified data must be loaded into the simulation and the simulation must be configured to accept and fully use the new data. The simulation must be tested to ensure that the data is handled properly. These formerly were almost insurmountable hurdles.

Soon, using AMSAA data will be no more difficult than designing a scenario, building a terrain database, or integrating multiple simulations. It won't be a trivial task, but loading and verifying the data will be straightforward and reliable, and will require minimal technical expertise. Recent technologies for this improvement include: standard data formats from AMSAA, support for these formats by the simulation, and tools to ensure that the simulation can use the data. The new innovation described in this paper is the Expected Results Method, which automates the verification of the simulation's actual performance for all of the thousands of firer-target and observer-target pairings, multiplied by conditions like range, exposure, dispersion, etc.

Application of these concepts to ARCIC Experiments has already shown dramatic reductions in time to load and check the data, and equally dramatic increases in the quality of the results of the simulation. As the tools and processes mature, we expect that these trends will continue and the promise of credible battlefield simulation will be realized.

ACKNOWLEDGEMENTS

Thanks to Scott Pridgeon and the Joint Data Center at AMSAA for help in calibrating the Expected Results programs.

REFERENCES

Muessig, Paul, "An Integrated Approach to Evaluating Simulation Credibility", Aug, 2001
Monday, Paul, "Configuring OneSAF with Classified Data", Huntsville Simulation Conference, 2009
Sargent, RG, "Verification and Validation of Simulation Models", Journal of Simulation, 2013
Logsdon, John, Wittman, Dr Rob, "Standardization, Transformation, and OneSAF", RTO-MP-MSG-056 20 – 1, 2007
Barnett, Andrew, "Using Semi-Automated Forces OneSAF as a Tool to Support Analysis", AMSAA, March 31, 2009