Proceedings of the 2023 Winter Simulation Conference
C. G. Corlu, S. R. Hunter, H. Lam, B. S. Onggo, J. Shortle, and B. Biller, eds.

SQUASHING BUGS AND IMPROVING DESIGN: USING DATA FARMING TO SUPPORT VERIFICATION AND VALIDATION OF MILITARY AGENT-BASED SIMULATIONS

Susan K. Aros
Mary L. McDonald

Naval Postgraduate School
1 University Circle
Monterey, CA 93943, USA

ABSTRACT

Verification and validation of complex agent-based human behavior simulation models is a challenging endeavor, particularly since a dearth of real-world data makes it impossible to use most traditional validation methods. Data farming techniques have stepped up to the challenge, proving to be a valuable tool for verification and validation of complex models. In this paper we demonstrate how data farming and analysis aids in the verification and validation of complex models by presenting specific examples pertaining to WRENCH, an agent-based simulation model that represents complex interactions between security forces and civilians during civil security stability operations. We first provide an overview of data farming and its relevance for verification and validation of military agent-based simulation models, then give an overview of WRENCH, and finally demonstrate with examples how we have used data farming to aid in the verification and validation of WRENCH.

1 INTRODUCTION

Stability operations often involve interactions with civilians, some of whom may be hostile. Behavior of these civilians can be difficult to predict, particularly their response to various force activities designed to establish civil security. Research is needed to advance our understanding of civilian behavior and potential responses. Agent-based simulation modeling has been shown to be a well-suited methodology for modeling human behavior since it explicitly models the individual agency of people.

When developing agent-based simulation models that incorporate human behavior in potentially volatile situations, the primary challenge that arises is the complexity of humans, the variety of influencing factors on human behavior, the interplay among these factors in an individual, and the role that interpersonal interactions and relationships play in behavioral decision-making. Researchers at the Center for Modeling Human Behavior at the Naval Postgraduate School have developed a simulation model, the Workbench for refining Rules of Engagement against Crowd Hostiles (WRENCH), that models many of these complexities in a civilian population (Aros et al. 2021). WRENCH also models a security force that has the option to use a variety of intermediate force capabilities (IFCs) under different tactical rules of engagement (ROEs) during the civil security mission.

Verification and validation (V&V) of a model like WRENCH presents significant challenges. For many model constructs there is no real-world data to inform their design or parameterization. In addition, there is not enough real-word data to perform traditional model validation. Therefore, it is important to leverage any methods and tools that are available for V&V. Source materials from relevant disciplines must be consulted to ensure that the conceptual model aligns with what has been discovered by those disciplines, and in some cases, reasonable model parameter values or ranges can be distilled from those sources. Extensive code testing is an important part of verification activities, and face validation activities are helpful in validating model behavior as well. And for complex models such as WRENCH it is desirable to harness the power of large-scale experimentation. Indeed, Law (2022) stresses the role of conducting designed experiments to support V&V; one goal of this process being to perform sensitivity analyses to determine the most influential model factors under various conditions. Being able to separate the few critical factors from the many less-impactful factors, particularly for a model as highly-dimensional as WRENCH, helps focus further data collection and validation efforts. Data farming, a methodology for iterated design, conduct, and analysis of experiments, improves the efficacy of V&V efforts and accelerates model development and refinement.

In this paper we first give an overview of verification and validation issues and techniques for military simulation models. We then provide background information about WRENCH and the civil security scenario. Following that, we discuss data farming and analysis methods used with WRENCH and provide examples of analysis that led to squashing bugs and improving the design.

2 APPROACHES TO VERIFICATION AND VALIDATION OF MILITARY SIMULATIONS

The Department of Defense's authoritative instruction on verification, validation, and accreditation for models and simulations (US DoD 2018) defines verification as "the process of determining that a model or simulation implementation and its associated data accurately represent the developer's conceptual description and specifications." The same reference defines validation as "the process of determining the degree to which a model or simulation and its associated data are an accurate representation of the real world from the perspective of the intended uses of the model." Hartley (1997) summarizes issues particular to V&V of military simulations, owing to the unique nature of the domain, noting the difficulty in obtaining real-world referents to which simulation output data can be compared as well as the human element. Consequently, Hartley emphasizes the value of viewing V&V as a continuous process that continues until sufficient confidence is reached to use the model for the intended use.

So which methods can be used to conduct continuous V&V? There are many important works on the topic. Sargent (2020) describes several methods, including face validation, structured code review, and experimentation. Like Hartley, Sargent emphasizes that it is often too costly, too time consuming, or impossible to determine that a model is *absolutely* valid over the complete domain of its intended applicability. Therefore, it is imperative to conduct a series of tests until sufficient confidence is obtained that a model can be considered valid *for its intended application*. (Sargent 2020). He further states that a model may be valid for one set of experimental conditions and invalid in another, a concept that is included in his graphical paradigm of the model development process. Notably, this process includes experimentation and analysis as key components.

Law (2015) and Law (2022) provide an overview of techniques and approaches to model verification and validation, and similarly emphasizes the role of experimentation and the need to run simulation models over a variety of settings of the input parameters. Since we do not have the luxury of unlimited amount of time and resources to conduct V&V, knowing which model factors drive results under various conditions is very valuable as it allows the modeler to prioritize data verification and validation efforts on these inputs.

3 DATA FARMING FOR VERIFICATION AND VALIDATION ACTIVITIES

With the important role of experimentation for V&V in mind, we now turn to discussing data farming. In this section we provide a brief introduction to data farming and then discuss experimental designs and analysis methods for data farming.

3.1 What is Data Farming?

Data farming is an approach for iterated design, conduct, and analysis of computer experiments that is particularly well-suited to working with simulation models that have a very large number of inputs that could be tested (Sanchez et al. 2020). Data farming is used to address an overarching analysis goal such as verification, validation, stress testing, or exploration of results to answer specific research questions. Specific analytic objectives are set toward this end. These objectives then inform the selection of specific factors, factor ranges and levels, and the type of experimental design to be used. High performance computing systems are then leveraged to run the large-scale experiment, thus 'growing' the specific data required to meet the analytic objectives. (Sanchez et al. 2020). A variety of analysis and visualization methods are then employed to explore the resulting data and inform the analysis objectives. This process can then be iterated until the analysis objectives have been met and the goal achieved. The building blocks of data farming are a collaborative approach to rapid scenario prototyping, modeling platform development, design of experiments, high performance computing, and the analysis and visualization of the output (NATO 2014).

Sanchez (2018) compares data farming to data mining in more detail, drawing a contrast between the "3 Vs" of data mining (volume, velocity, and variety) to the "3 Fs" of data farming (factors, features, and flexibility). Real world miners seek valuable nuggets of ore buried in the earth but have no control over what is there. Similarly, data miners seek to extract nuggets of information from the data they have available in existing datasets. Consequently, isolating cause and effect relationships may be difficult or impossible. On the other hand, real world farmers cultivate the environment to grow what they need and garner maximum yield. Similarly, data farmers intentionally choose how to vary the inputs of their simulation model, using sound techniques from the design of experiments (DOE) literature and knowledge of the model, to produce data that will lead to maximum information gain.

Sanchez et al. (2020) warns that if the "farming" plan for conducting model runs is not well-designed, several pitfalls can occur, which would severely limit the information that can be gained. Among these pitfalls are (1) basing analytic recommendations upon the output of only a few runs of a model; (2) the confounding of experiment variables (called factors), meaning that the effects of factors on the response cannot be untangled; and (3) failing to detect an interaction effect, where interaction means that the effect that one factor has on the response depends on the value of another (interacting) factor.

Other important considerations are size and scope of the planned experimentation. With a highly dimensional simulation model such as WRENCH, even a single V&V experiment to test a wide variety of parameters and ranges could easily exceed available computing resources, requiring months or even years to get the desired data. Thankfully, advances in the development of efficient experimental designs combined with high performance computing (HPC) systems typically used in data farming enable these wide-ranging experiments to be conducted at a scale and efficacy that otherwise would not have been possible.

3.2 Data Farming: Experimental Designs

We now turn to a brief overview of experimental designs, with a focus on those we have found most useful. The first step in designing an experiment is determining the set of model inputs and simulation parameters, called *experimental factors* in DOE terminology, that will be varied. For each factor, we next determine the range over which we wish to vary it, and the number of levels to sample from each.

The goal of the analysis influences the selection of a factor's minimum value and maximum value in an experiment, and therefore determines range. For example, if our goal is to perform a sensitivity analysis to determine which inputs induce significant variation in outputs, even when varied over small ranges, we may only wish to vary a factor plus or minus 10%, say, from its baseline value. A different goal is to perform a broad sweep analysis, intended to assess change in model outputs as an input is varied over all or a significant portion of its feasible range. We may wish to combine these goals, depending on the purpose of the analysis. For example, we might choose to vary some factors over small ranges, to test sensitivity or to account for reasonable natural variation while ranging other factors more broadly.

The number of levels sampled of each factor influences what types of insights are possible. For example, sampling a factor at only two levels will allow only linear effects to be fit, and we would miss the opportunity to detect curvature or estimate a quadratic or higher order polynomial response. The number and types of factors (e.g. categorical, discrete, continuous), the number of levels sampled for each, the analytical goals, and the computational budget are all considerations in the selection of an experimental design. An experimental design choice results in a run matrix, where every column represents a factor and every row represents a single design point (DP), which is a particular combination of factor values.

There are many different design choices available. Perhaps most well-known are gridded designs such as the full factorial, which tests every possible combination of the factors, and the fractional factorial, which tests carefully selected subsets of the full factorial, sampling at only two levels per factor, and trading some ability to independently estimate higher order interaction effects (resolution) with the number of design points required (efficiency). There are many good references for design of experiments such as Kleijnen (2017). However, the full factorial will become infeasible as the number of factors and levels grows and the fractional factorial limits to testing at only two levels. Another option is the space-filling design, which samples the interior of a design space in an efficient manner. We often use a first or second order nearly orthogonal Latin hypercube (NOLH), the second order being able to minimize correlations between all main effects, quadratic terms, and two-way interactions (Cioppa and Lucas 2007; MacCalman 2017). Since the NOLH works best when there are only continuous-valued factors, we also frequently use a nearly orthogonal and balanced (NOB) design (Vieira et al. 2013) that can handle a mix of continuous, discrete, and categorical factors, minimizing imbalance in categorical factor sampling while also minimizing pairwise correlations. These designs are both efficient and flexible, allowing a wide variety of response surface metamodels to be fit, including those with higher order interactions and polynomial terms. It may also be useful to combine different experimental designs, for example by crossing a small gridded design with a more efficient design over others. For a tutorial on designing and conducting large-scale simulation experiments, along with examples of different classes of designs useful for data farming studies, see Sanchez et al. (2020).

We note that NetLogo does have a built-in utility for experimentation, called BehaviorSpace, which makes it easy to specify full-factorial experimental designs. However, due to the number of runs required for a full-factorial design, the high-dimensionality of WRENCH, and the scope of the planned experimentation, it is infeasible to use BehaviorSpace. We therefore developed WRENCH Farmer, a custom add-on that allows large-scale, efficient experimentations to be run with any design provided in a distributed manner using HPC. We conclude this section by noting that data farming toward an analysis goal often includes iterative experiments, with each design inspired by learnings from the prior experiment.

3.3 Data Farming: Analysis Methods

In this section, we give a broad overview of types of analyses that can be useful for drawing insights from designed experiments. Good references on the topic of visualization and analysis of experiment data include Barton (2021), Barton (2015), Box et al. (2005), Law (2015), Sanchez and Lucas (2002) and Tufte (1983). Since WRENCH is an agent-based model, we note that Sanchez and Lucas (2002) list several features of agent-based models, such as nonlinearity, emergence, and the presence of many higher-order interactions, that make analysis of the data more complex than analyzing data from other types of simulation models.

A key point in analyzing data from complex agent-based simulation models is that visual representations of the data are necessary and should precede and complement metamodeling (developing a “model of a model” that captures the relationship between inputs and outputs). A metamodel could be as simple as a regression equation or something much more complex. The metamodeling methods we most frequently use in analyzing WRENCH outputs are ordinary least squares regression, generalized linear models (e.g., logistic regression), classification and regression partition trees, bootstrap (random) forests, and Gaussian process modeling. In some cases, a metamodel may be used as a surrogate for the underlying simulation that generated the output. With time series data, we have explored the use of functional data analysis to treat the entire time series curve as the response, instead of fitting metamodels to one or more summary measure(s) and examination of cross-correlation. We are interested in evaluating dynamics over time as well as the end state for a specified simulated time-period. We also frequently use histograms and summary statistics, outlier box plots, bar and line plots, scatter plots, correlation matrices and plots, contour and surface plots, cluster analysis, Pareto optimal frontier for multiple objectives, heatmaps, and parallel coordinate plots. High-influence main effects or interactions discovered via metamodeling typically make good candidates for use in plots.

Our data analysis proceeds in stages, typically starting with a set of simpler analyses best suited for each type of data, We follow this by “deeper dives” and more complex analyses chosen based on the initial analysis and often guided by hypotheses. The hypothesis generation aspect is particularly important because the data is so highly dimensional. Subject matter expertise and focused inquiries are critical to help guide the search for relationships that are expected and for those that are counter-intuitive as these may indicate a verification or validation issue. Additionally, we are always mindful of the possibility of false positives, i.e. the conclusion that there is a statistically significant relationship when it has occurred simply by chance. Therefore, practical, as well as statistical, significance must be considered. Blindly and broadly searching for statistical relationships alone increases the rate of false positives.

Data farming methods are very helpful in confirming the expected behavior of the code and the model; in addition, they are also very effective in bringing into view potentially unexpected results and model behavior. Agent-based simulation models have the feature of producing emergent behavior, which creates an extra challenge when interpreting these unexpected results. Therefore, when using data farming for V&V of agent-based simulation models we want to first explore the data for any results that could indicate errors in the code, and conduct any necessary code review to determine if in fact an error is present in the code. Once code errors are ruled out, an effort must be made to determine if there could be a logic error in the design of the conceptual model which the code implements. Only once these have both been ruled out can we conclude that the surprising results of behavior may be valid emergent behavior.

4 WRENCH DESCRIPTION

WRENCH is an agent-based simulation model coded in NetLogo that models a security force carrying out a civil security mission and addressing any emerging civilian threats using IFCs and lethal weapons. WRENCH currently contains a well-developed urban area/compound security scenario, but can be adapted to model border-control and humanitarian aid distribution missions as well. WRENCH is a tactical-level simulation model covering a relatively small geographic area; it uses one-second time-steps with a typical run length measured in minutes and hours, rather than days. Here we provide a brief overview of WRENCH to provide context for the examples in this paper; for a more detailed overview see Aros et al. (2021).

WRENCH draws on literature from multiple disciplines to model complex drivers of individual and group behavior. Each individual civilian in the population is represented by a physical person agent and an identity agent. People agents move about an environment populated with a GIS dataset, interact with other physical agents, and if they are perceived as a threat they can be targeted, and possibly injured or killed, by the security force. Individual identities have state variables representing emotions (fear, anger), cognitive elements (belief about the legitimacy of the forces, hostile intent, memory), needs, and an overarching objective such as protest, attack, or stay-safe. An identity's variables are affected by combinations of other variables via formulas and conditional statements. Many direct-influence relationships are mediated and/or moderated by other of the identity's own state variables as well as perceptions of their environment, other agents in proximity, other members of the same group, and details about weapons impacts received or witnessed. Identity state variable drive their intentions, which in turn drive the associated person's behavior. Additionally, people agents modify their movement to avoid collisions, attempt to maintain desired personal space, and stay in relative proximity to their group leader (if their individual identity is in a group). Overall, many settings are available to the user to customize aspects of the population design.

Civilian individuals can group with each other, and these groups are modeled explicitly as a social identity group (SIG) identity agent. SIGs formed at the start of the simulation can be specified by the user (based on family relationships and/or a shared demographic identity characteristic), and SIGs can also form, change, and dissolve dynamically over time for compelling reasons such as fear. Member identities influence, and are influenced by, the group to which they belong. Identities of any level can join together to create a next-higher-level SIG, thereby allowing large groups to grow while still maintaining the individual identities and the relationships within the smaller groups that joined together.

In the area/compound security scenario, the security forces patrol the area in vehicles on the road network and secure the compound with a gate guard at each compound gate. If a person invades the compound, or if a gate guard perceives an immediate threat, the compound squad begins to defend the gates. At that point, the patrolling squads are dispatched to gates to assist the with defense of the compound.

The security force can have a user-designated set of one or more types of IFCs issued to them and must operate within user-specified ROEs. IFC details are modeled according to product specifications and the immediate effects are informed by these specs, literature, and a study of video evidence. IFCs are also categorized by severity and type of effect: there are three levels of pain IFCs which cause injury levels varying from lasting pain to immobilization, and three levels of psych IFCs that provide varying levels of intimidation and discomfort but have no lasting physical effects. Once security force personnel are in defense mode, they decide who to target and which weapons to use based on the perceived threat of the individual, user-specified ROEs (a detailed rule-set), common rules such as not shooting people in the back, and other settings that moderate escalation of force decisions whenever a direct weapon-severity to threat-level match is not available.

WRENCH is designed to be run interactively, where the user can set some aspects of the population and force design, and change several of these settings during a run to see immediate effects of that decision. This is facilitated by NetLogo's graphical user interface (GUI) as well as two additional GUIs we developed to support ROE design and IFC selections. WRENCH can also be run constructively, and in this mode, many more aspects of the population and force design can be specified, and a great deal of data can be output which enables a deeper exploration of effects and interactions. To date WRENCH has been primarily used in the constructive mode with experimentation but is also run in the interactive mode for some aspects of ongoing V&V activities. The interactive mode also has the potential to be used in training.

5 WRENCH DATA AND EXPERIMENTS

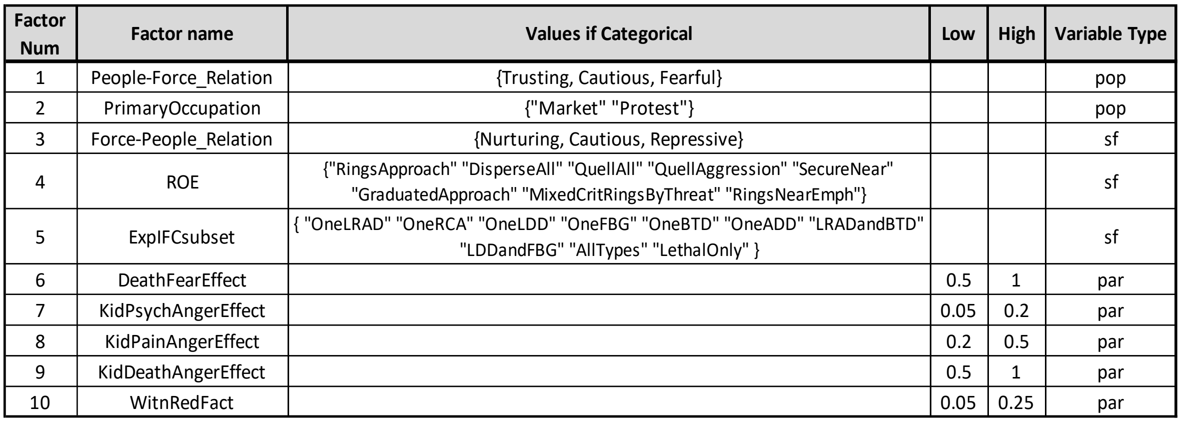
WRENCH outputs, for each simulation run, raw data consisting of several data files that capture a wide variety of information for each time step. The output includes information about each identity's emotions, cognitive elements, needs, overarching objective, and group membership if in a group. Identifying labels for identities, which indicate whether it is an individual identity or a SIG identity, and if a SIG, what type it is and whether it was formed initially or dynamically is captured. Also captured are many tracking metrics such as the number of people that invaded the compound, the number of times weapons of each severity level were used by the security force, injury levels in the population, and any deaths that have occurred from either lethal force or due to accumulated injuries from high-severity IFCs.

Many possibilities exist for creating aggregations of the raw time series data. The choice depends on the purpose of the analysis or focused query. At a minimum, we automatically produce an "end of run" data set where each column of data represents either the value on the last tick of a cumulative count metric or a summary statistic of a metric that can rise and fall over time. Depending on the focus of the analysis, we may also code additional variables that help easily track when unusual conditions have occurred, one example being if the compound was penetrated by many intruders, but no force of any kind was ever used by the security force. When aggregating, summary statistics we typically use (where the statistic is calculated over the ticks for each run), are initial value, final value, minimum, average, median, maximum, range, and delta (final value minus initial value).

Another type of data aggregation occurs when we calculate statistics (most typically mean, median, or specified quantile) over the replications within each design point. We typically use such data for fitting metamodels. Since the data is so highly dimensional, we may additionally aggregate data as needed to perform a specific focused query. For example, we might use delta or range metrics to identify which factors are driving high-magnitude change, then focus on these as dimensions of a plot. Another example is to use statistical summary measures as "indicators" to identify DPs for which there was particularly small or large magnitude change, and then pull those DPs aside for side-by-side graphical analysis. Similarly, for a given DP(s), we may use indicators to select seeds that would be interesting to take a deeper dive into, for example, to examine change at the individual agent or group level. The number of possibilities for the analysis given the number of inputs and number and types of outputs make analysis a challenging, but always interesting, endeavor!

With regard to data farming experiments, we have performed over a dozen large-scale experiments over the course of WRENCH development, using several of the designs mentioned in section 4.2. As an example, we will discuss one of these experiments. The description of factors is shown in Table 1. For this experiment, we varied ten factors, five of which are categorical, and the others are discrete or continuous. The variable type given in the last column denotes whether the input being varied is a population characteristic (pop), security force choice (sf), model parameter (par), or environmental variable (env). Given the mix of factor types, we use a NOB design with 100 DPs. Since WRENCH is a stochastic model each DP is replicated 30 times with each replication using a different random number seed. This design efficiently samples the space and allows us to fit a wide variety of metamodels to the output.

Table 1: Description of factors for one of the experiments.



Factor Num	Factor name	Values if Categorical	Low	High	Variable Type
1	People-Force_Relation	{Trusting, Cautious, Fearful}			pop
2	PrimaryOccupation	{"Market" "Protest"}			pop
3	Force-People_Relation	{Nurturing, Cautious, Repressive}			sf
4	ROE	{"RingsApproach" "DisperseAll" "QuellAll" "QuellAggression" "SecureNear" "GraduatedApproach" "MixedCritRingsByThreat" "RingsNearEmph"}			sf
5	ExpIFCsubset	{ "OneLRAD" "OneRCA" "OneLDD" "OneFBG" "OneBTD" "OneADD" "LRADandBTD" "LDDandFBG" "AllTypes" "LethalOnly" }			sf
6	DeathFearEffect		0.5	1	par
7	KidPsychAngerEffect		0.05	0.2	par
8	KidPainAngerEffect		0.2	0.5	par
9	KidDeathAngerEffect		0.5	1	par
10	WitnRedFact		0.05	0.25	par

6 DATA FARMING FOR VERIFICATION AND VALIDATION OF WRENCH

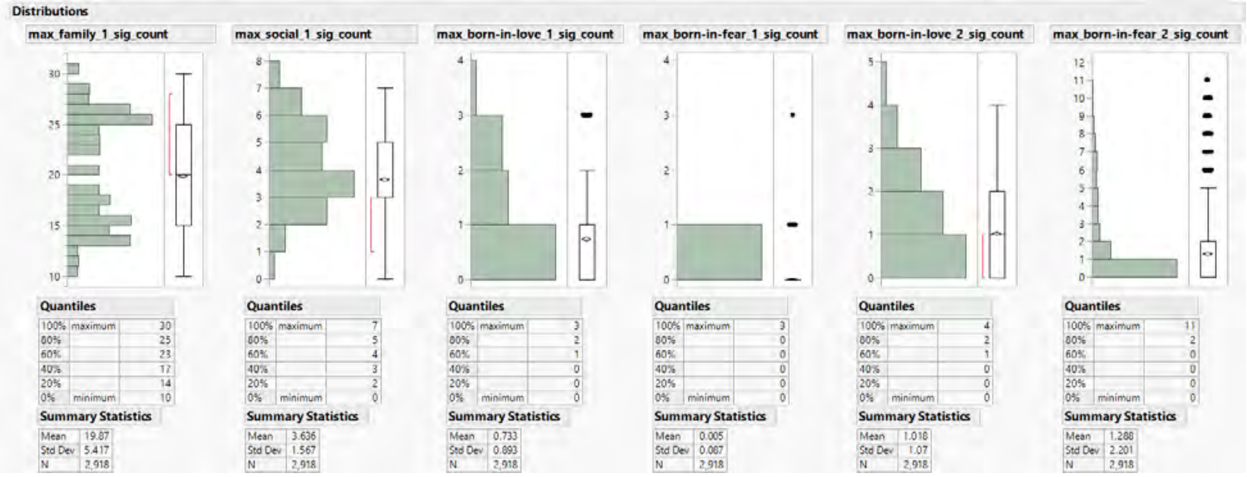
We turn now to describing several examples of how large-scale experimentation and analysis helped to verify model functionality, expose issues, and identify where additional complexity was warranted. In the past three years of model development, over a dozen experiments, totaling hundreds of thousands of model runs, have been performed. The use of efficient experimental designs has enabled the testing of thousands of combinations of inputs that would have otherwise gone untested. The examples we present here demonstrate how the high volume of runs and different analyses of varying levels of complexity can be helpful. They also demonstrate the use of different levels of aggregation, from no aggregation (exploring time-series data of individual agents within a single replication) to high-level aggregation.

6.1 Discovering Rare Terminating Errors

Early experimentation with WRENCH revealed that some combinations of inputs caused an error that wasn't seen during regular code testing activities using WRENCH's interactive mode with animation and error checking routines. During an experiment, certain errors would terminate the run and place an entry in the error log, helping greatly in locating the cause in the code. After several iterations of experiments it was believed that all bugs causing terminating errors had been caught and fixed. However, a subsequent larger experiment exploring a broader set of factors and ranges, with 55,330 total runs, illuminated two elusive errors. One of these errors occurred in just 5 runs (0.00903% of runs), and another occurred in only 3 runs (0.00542% of runs). Without the large-scale data farming experiment testing a wide range of factor combinations, these rare errors may never have been caught. This also demonstrates the value of data farming for V&V before we even delve into the results.

6.2 Discovering a Hidden Error

As development has continued, additional outputs have been coded to measure SIG formation and dynamics. The first experiment using the new outputs revealed surprisingly low level-1 born in fear (BIF) SIG formation, as shown in the Figure 1 histograms and summary statistics. For these runs, there were 134 instantiated people (level-0 "me" identities). The count metrics displayed here represent the max (over the run) of the number of SIGs formed at each level of the hierarchy. Individuals may band together to form a level-1 SIG, level-1 SIGs may band together to form a level-2 SIG, and so on. The histograms revealed that there were quite a bit fewer level-1 BIF SIGs being formed than level-2 BIF SIGs. Normally we would expect there to be fewer level-2 than level-1SIGs, but since the level-1 family and social SIGs are not shown it was possible that there was a reasonable explanation for this; however, if that was the reason then we'd expect a similar relative difference for the level-1 and level-2 born in love (BIL) SIGs which was not revealed in the histograms.



Quantiles

		max_family_1_sig_count	max_social_1_sig_count	max_born-in-love_1_sig_count	max_born-in-fear_1_sig_count	max_born-in-love_2_sig_count	max_born-in-fear_2_sig_count
100%	maximum	30	7	3	3	4	11
80%		25	5	2	0	2	2
60%		23	4	1	0	1	0
40%		17	3	0	0	0	0
20%		14	2	0	0	0	0
0%	minimum	10	0	0	0	0	0

Summary Statistics

	max_family_1_sig_count	max_social_1_sig_count	max_born-in-love_1_sig_count	max_born-in-fear_1_sig_count	max_born-in-love_2_sig_count	max_born-in-fear_2_sig_count
Mean	19.87	3.636	0.733	0.005	1.018	1.288
Std Dev	5.417	1.567	0.893	0.087	1.07	2.201
N	2,918	2,918	2,918	2,918	2,918	2,918

Figure 1: Formation of different types of SIGs in WRENCH.

With no clear explanation for the low level-1 BIF formation, a code review was undertaken. In this review, a hidden coding error was detected. When an identity looks for another to join with due to fear, a distance constraint is enforced, and this distance constraint is scaled such that higher level (larger group) identities can look further for another SIG to join with. However, using the identity level of the seeking identity as the variable to scale the distance constraint led to an inadvertent 'multiply-by-zero' issue for level-0 identities to form a BIF, meaning they had to be essentially co-located, which was rare. This formula was adjusted to avoid multiplying by zero, and a subsequent experiment showed that this resolved the issue. Without doing a side-by-side comparison of results across metrics, this error may never have been caught. This demonstrates the fruit from even a simple analysis.

6.3 Exploring Complex Relationships

The WRENCH model contains a dynamic network of interactions and moderating relationships that governs how the personal attributes of each individual change over time in response to external stimuli and internal attributes, and these changes influence the behavioral decision-making of each individual. For example, an individual's hostility is one of the primary influences on their behavioral decisions, and their hostility level is primarily driven by their own fear and anger as well as their beliefs about the legitimacy of the forces. In addition, use of weapons by the security force is the primary, but not sole driver, of changes in an individual's fear, anger, and legitimacy belief levels, and the strength and direction of these effects depend not only on the types of weapon but also the impacted person's hostility (actual hostility for impact on self or perceived hostility for impact on a witness), whether the impacted person is a child or a colleague (member of the same SIG), and whether the impacted person is killed by the impact.

As the complexity of a model increases it becomes more difficult to assess the aggregate effects of complex influences and relationships between variables. Data farming helps with the exploration of these effects and relationships. As an example, we constructed a scatterplot to visualize the relationship between type of IFC used by the forces and the resulting total effects of the combination of influences the legitimacy beliefs and hostility levels of the civilians (Figure 2). Each dot in the plot represents an average across all replications in a design point of the experiment, with the population's average change in legitimacy belief on the vertical axis and average change in hostility on the horizontal axis. The color of the dot represents which type(s) of IFCs were available for the forces to use (in addition to lethal), with orange and red indicating higher-level pain IFCs and blue and green indicating lower-severity psych IFCs. We would expect to see that the use of lower severity IFCs results in decreased hostility increased legitimacy beliefs, which is what we found. And, as expected, we also see that higher severity weapons led to a decrease in legitimacy beliefs and an increase in hostility. This logical result increases our confidence in the validity of the model design. In our V&V activities we've generated a wide variety of other charts and metamodels to similarly explore these complex model dynamics.

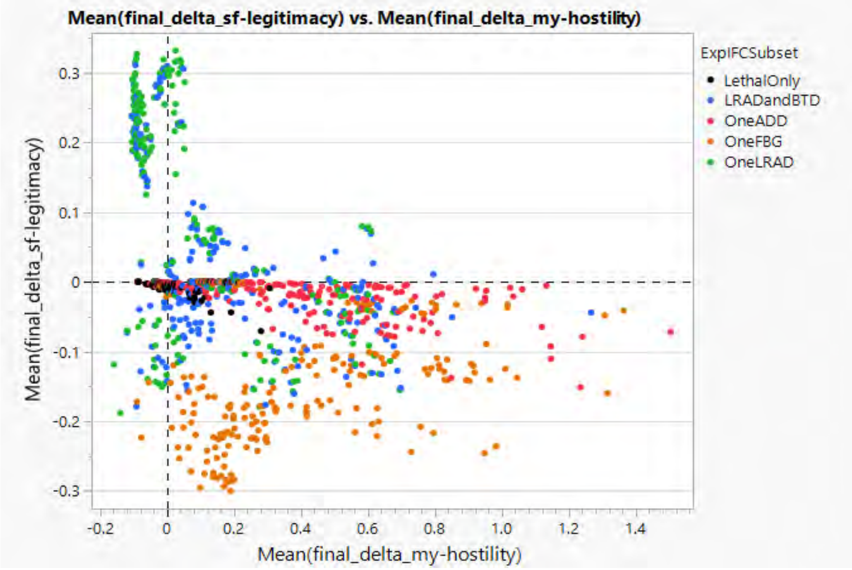


Figure 2: Change in agent hostility and legitimacy beliefs by type of IFC.

6.4 Exploring Individual Agents Over Time

Analyzing aggregated data, as demonstrated in the preceding examples, is very helpful but does have its limitations since it can mask important information. Our early analyses using time series output were focused on exploring aspects of the population on average. As experimentation and analysis have continued, though, we have been able to begin diving deeper and performing more micro-level analyses focusing on individual agents. After using other data farming analysis methods to identify runs for which the micro-level analysis might be most useful, we then plotted the time-series data for individual agents. Through this exploration we discovered that there were unreasonably rapid changes for some identities on some metrics that the average metrics had masked. Figure 3 shows a time-series plot tracking the legitimacy beliefs of a few sample identities. The figure key provides the unique NetLogo *who* number for the identity; in this case *who* numbers <= 129 represent individual identities, and the others represent SIG-level identities.

These results prompted a review of the relevant code, and with some focused testing, the source of the issue was discovered: a specific type of data structure in the code was not getting cleared after use. This structure tracked the perceived legitimacy, as interpreted by the individual, of each weapon impact they received or witnessed since the last iteration of the cognitive processes that determined their revised legitimacy belief, and was supposed to be cleared as soon as the legitimacy belief was changed accordingly. The result was a compounding effect of every impact received or witnessed while the formulas were calibrated for a single effect of each on legitimacy beliefs, thus changing legitimacy rapidly once any impact happened to initiate a change in legitimacy beliefs. For SIG identities, this effect still exists but is muted because legitimacy beliefs of a SIG are affected by all members rather than by direct effect, and groups change more slowly. Note that the lines for SIG identities 277 and 281 terminate mid-chart because the SIG disbanded where its line terminates.

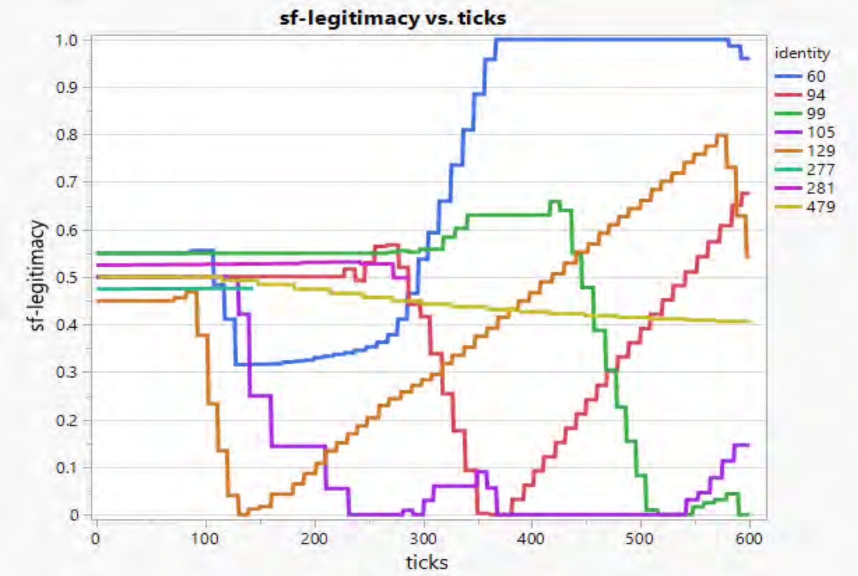


Figure 3: Individual security force legitimacy belief over time.

6.5 Assessing Emergent Behavior

One of the most challenging aspects of V&V for agent-based simulation models is determining whether emergent behavior is valid or invalid. In our experimentation and analysis we tracked and output the amount of time that a security force vehicle on patrol was delayed due to a person blocking the road. The analysis revealed that many runs had an amount of delay that seemed excessive, but a careful examination of the code did not reveal any errors. When running WRENCH with animation during other verification activities, we noticed that occasionally when a vehicle was stopped by a person in the road the person remained there and preventing that patrolling squad from aiding in the defense of the compound. We then developed the hypothesis that a higher number of protesters, a longer time with no intruder into the compound, and a cautious or nurturing force stance towards the crowd would contribute to a larger crowd gathering and "milling about" near the compound, thus blocking one or more of the roads near the compound and increasing delay. To test this hypothesis, we constructed a partition tree to identity characteristics of scenarios that were associated with high total delay values, allowing not only experimental factors but also other hypothesized contributors to enter as explanatory variables. The fitted tree, shown in Figure 4, confirms that (reading down the right-most branch) the largest delays occurred when there were at least 70 end-of-run protestors, no intrusion or time of first invader greater than 32 seconds, and a security force that adopted a cautious or nurturing stance. Under these conditions (right-most terminal node), the average delay was approximately 982 seconds, or over 16 minutes. The confirmation of this hypothesis supports the interpretation of these high delays as valid in situations where a patrol force does not attempt to clear people from the road. (Multiple real world events also support the validity of this emergent model behavior.)

This is an example of an emergent behavior that was not anticipated and was initially considered suspect, but then supported as being valid by further results analysis. The other benefit of this analysis for us is the determination that adding model complexity to enable security force vehicles on patrol to engage people blocking the road is justified for scenarios representing patrolling squads who do this in reality, since the results showed that these delays significantly affected the results. When modeling complex systems, and especially human behavior, it is important to be diligent in not adding complexity for complexity's sake, but rather to "justify your complexity" (as emphasized by Lee Schruben, Titan of Simulation) to help keep modeling projects on-time and within budget without sacrificing validity. Data farming can help determine when complexity is justified.

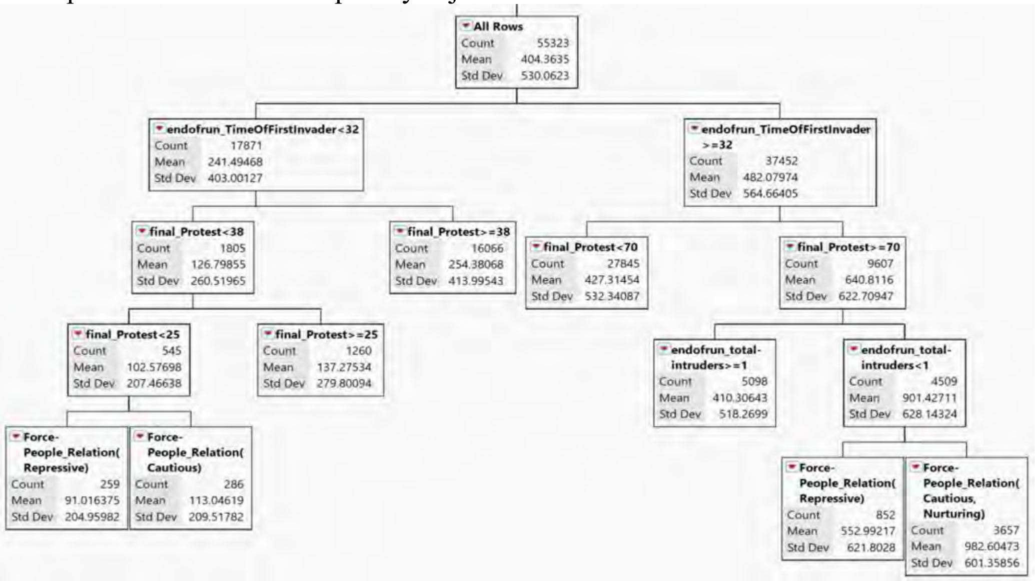


Figure 4: Partition tree of total delay.

7 SUMMARY AND FUTURE DIRECTIONS

Data farming is a powerful tool to probe the functionality of complex models relatively quickly and efficiently. Our use of data farming has led to valuable, credible analyses that have enabled model improvements at a pace and efficacy that was otherwise not possible. With WRENCH we have thus far primarily used data farming in an exploratory manner to support verification and validation, but as improvements are made and our confidence grows in the validity of our model, we are moving toward using data farming to address important research questions about civil support activities.

WRENCH has great potential to provide significant insights that can aid in decision-making about which intermediate force capabilities may be best suited to different civil support situations, and what approach and choices by the security force may be most effective. With its interactive capabilities, WRENCH also has the potential to be used interactively to support training. And finally, information gained from WRENCH could be used to enhance the realism of live training exercises and inform the development of wargames for civil support missions.

ACKNOWLEDGMENTS

The development and ongoing improvement of WRENCH was made possible by support from the Joint Intermediate Force Capabilities Office. WRENCH data farming was made possible by Stephen Upton, a Faculty Associate for Research in the Simulation Experiments and Efficient Designs (SEED) Center at the Naval Postgraduate School. Mr. Upton developed WRENCH-Farmer and WRENCH-Miner used to conduct and post-process large-scale WRENCH experiments.

REFERENCES

Aros, S. K., A. M. Baylouny, D. E. Gibbons, and M. L. McDonald. 2021. "Toward Better Management of Potentially Hostile Crowds". In *Proceedings of the 2021 Winter Simulation Conference*, edited by S. Kim, B. Feng, K. Smith, S. Masoud, Z. Zheng, C. Szabo and M. Loper, 1-12. Piscataway, New Jersey: Institute of Electrical and Electronic Engineers, Inc.

Barton, R. R. 2015. “Tutorial: Simulation Metamodeling”. In *Proceedings of the 2015 Winter Simulation Conference*, edited by L. Yilmaz, W. K. V. Chan, I. Moon, T. M. K. Roeder, C. Macal, and M. D. Rossetti, 835–849. Piscataway, New Jersey: Institute of Electrical and Electronic Engineers, Inc.

Barton, R. R. 2021. “Tutorial: Graphical Methods for the Design and Analysis of Experiments”. In *Proceedings of the 2021 Winter Simulation Conference*, edited by S. Kim, B. Feng, K. Smith, S. Masoud, Z. Zheng, C. Szabo and M. Loper, 1765–1779. Piscataway, New Jersey: Institute of Electrical and Electronic Engineers, Inc.

Box, G. E. P., W. G. Hunter, and J. S. Hunter. 2005. *Statistics for Experimenters: An Introduction to Design, Data Analysis and Model Building*. 2nd ed. New York: Wiley.

Cioppa, T. M., and T.W. Lucas. 2007. “Efficient Nearly Orthogonal and Space-Filling Latin Hypercubes”. *Technometrics* 49(1):45–55.

Hartley, D. S. 1997. “Verification & Validation In Military Simulations”. In *Proceedings of the 1997 Winter Simulation Conference*, edited by S. Andradóttir, K. J. Healy, D. H. Withers, and B. L. Nelson, 925–932. Piscataway, NJ: Institute of Electrical and Electronics Engineers, Inc.

Kleijnen, J. P. C. 2017. “Design and Analysis of Simulation Experiments: A Tutorial”. In *Advances in Modeling and Simulation*, edited by A. Tolk, J. Fowler, G. Shao, and E. Y¨ucesan, 135–158. Cham, Switzerland: Springer International Publishing AG.

Law, A. M. 2015. *Simulation Modeling and Analysis*. 5th ed. New York, NY: McGraw-Hill.

Law, A. M. 2022. “How to Build Valid and Credible Simulation Models”. In *Proceedings of the 2022 Winter Simulation Conference*, edited by B. Feng, G. Pedrielli, Y. Peng, S. Shashaani, E. Song, C.G. Corlu, L.H. Lee, E.P. Chew, T. Roeder, and P. Lendermann, 1283–1295. Piscataway, NJ: Institute of Electrical and Electronics Engineers, Inc.

MacCalman, A., H. Vieira, and T. Lucas. 2017. “Second Order Nearly Orthogonal Latin Hypercubes for Exploring Complex Stochastic Simulations”. *Journal of Simulation* 11(2):137-150.

NATO 2014. Data Farming in Support of NATO. Technical Report TR-MSG-088, NATO Science & Technology Organization.

Sanchez, S. M. 2018. “Data Farming: Better Data, Not Just Big Data.”. In *Proceedings of the 2018 Winter Simulation Conference*, edited by M. Rabe, A. A. Juan, N. Mustafee, A. Skoogh, S. Jain, and B. Johansson. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Sanchez, S. M., and T. W. Lucas. 2002. “Exploring the World of Agent-Based Simulation: Simple Models, Complex Analyses”. In *Proceedings of the 2002 Winter Simulation Conference*, edited by E. Yucesan, C. Chen, J. L. Snowdon, and J. Charnes, 116–126. Piscataway, New Jersey: Institute of Electrical and Electronic Engineers, Inc.

Sanchez, S. M., P. J. Sanchez, and H. Wan. 2020. “Work Smarter, Not Harder: A Tutorial on Designing and Conducting Simulation Experiments”. In *Proceedings of the 2020 Winter Simulation Conference*, edited by K.-H. G. Bae, B. Feng, S. Kim, S. Lazarova-Molnar, Z. Zheng, T. Roeder, and R. Thiesing, 1128–1142. Piscataway, NJ: Institute of Electrical and Electronics Engineers, Inc.

Sargent, R. G. 2020. “Verification and Validation of Simulation Models: An Advanced Tutorial”. In *Proceedings of the 2020 Winter Simulation Conference*, edited by K.-H. Bae, B. Feng, S. Kim, S. Lazarova-Molnar, Z. Zheng, T. Roeder, and R. Thiesing, 16-29. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Tufte, E. R. 1983. *The Visual Display of Quantitative Information*, Graphics Press, Cheshire, CT: Graphics Press.

United States Department of Defense. 2018. *DoD Modeling and Simulation (M&S) Verification, Validation, and Accreditation (VV&A)*. [DoDI 5000.61 with Change 1].

Vieira, H., S. M. Sanchez, K. H. K. Kienitz, and M. C. N. Belderrain. 2013. “Efficient, Nearly Orthogonal-and-Balanced, Mixed Designs: An Effective Way to Conduct Trade-off Analyses via Simulation”. *Journal of Simulation* 7(4):264–275.

AUTHOR BIOGRAPHIES

SUSAN K. AROS is a Research Assistant Professor in the Naval Postgraduate School’s Department of Defense Management and Director of NPS’s Center for Modeling Human Behavior. She has a PhD in Information, Risk, and Operations Management, an MEng in Operations Research and Industrial Engineering, an MA in Spiritual Formation and Soul Care, and a BA in Psychology. Her published research spans the areas of human behavior modeling, peace support operations, disaster response operations, inter-organizational communication, production planning and scheduling, and RFID use in remanufacturing, and uses various methodologies including agent-based simulation, discrete-event simulation, and mixed integer programming. Her email address is skaros@nps.edu.

MARY L. MCDONALD is a Faculty Associate for Research in the Naval Postgraduate School’s Simulation Experiments & Efficient Designs Center for Data Farming. She received her B.A. in Mathematics from Northwestern University and M.S. in Applied Mathematics from the Naval Postgraduate School. She has taught courses in probability and statistics and simulaiton analysis, and has lectured for courses in combat modeling. She has over 25 years of experience in applying simulation modeling, design of experiments, evolutionary algorithms, and high-dimensional data analysis to Department of Defense problems. Her email is mlmcdona@nps.edu.