Experiences with SIGNAL: Opportunities and challenges with distributed wargaming

Jonathan Whetzel
Nathan Fabian
Kiran Lakkaraju

Sandia National Laboratories
Albuquerque, New Mexico
87185 and Livermore,
California 94550

ABSTRACT

We describe the opportunities and challenges we faced when developing SIGNAL, an experimental wargame that was deployed as a distributed wargame.

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Issued by Sandia National Laboratories, operated for the United States Department of Energy by National Technology & Engineering Solutions of Sandia, LLC.

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

CONTENTS

1. INTRODUCTION

Experimental wargaming is a methodology developed to systematically, rigorously, and in a data driven manner, study problems of national security (Reddie et al. 2018; Lakkaraju et al. 2020). Wargaming is a powerful tool that has often been used to develop greater insight into complex national security problems by placing players in a simulated scenario (Perla and Curry 2011; Sabin 2014). Experimental sciences endeavor to study human decision making primarily through controlled experiments. Experimental wargames intend to combine the best of these two approaches by creating a controlled environment to study player behavior relevant to national security.

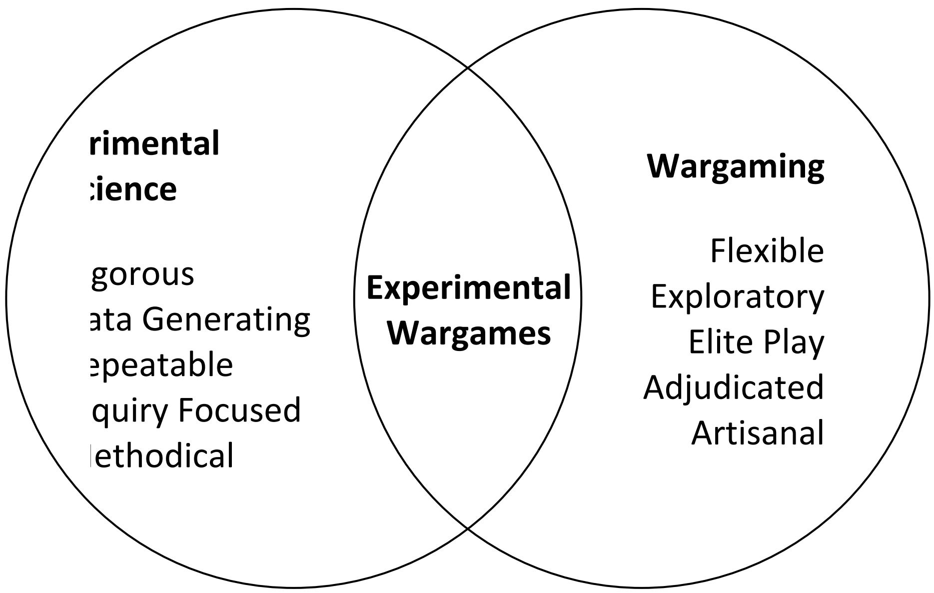

Figure 1: Experimental wargames lie at the intersection of experimental sciences and wargaming.

Experimental wargaming gives us options in addition to traditional approaches, such as empirical data analysis, to study emerging problems with cutting edge data analytic techniques. Conflict has grown immensely more complex with the emergence of numerous, powerful actors with global reach. Adding to the complexity are new domains of activity, particularly cyberspace. Modeling these conflicts from first principles will be difficult due to the numerous factors involved. Existing conflict data sets (such as the Dyadic Militarized Interstate Disputes (Maoz et al. 2019)) do not capture emerging technologies and their impact in new domains. Experimental wargames generate synthetic data that can help understand the issues present in these problems.

Experimental wargames follow a 9-step design and implementation process (Figure 2).

- **Identify Problem:** Understanding the overarching problem that is being considered.
- **Scope the Study:** Identifying specific scenarios that will be studied within a particular game.

- **Identify Research Question, IV and DV's**: Developing a testable, falsifiable hypothesis and articulating what will be varied (IVs) and what will be measured (DVs).
- **Design Prototype Game**: Develop a prototype of the experimental wargame. This requires deciding on the game mechanics, scenarios, number of players, etc.
- **Prototype and Test**: Implement the prototype, often as a board game, to test. Insights from the testing process may suggest a change in the game design (shown as the feedback loop to "Design prototype game" in Figure 2), and in some cases where it is realized that we cannot capture the domain in a way to facilitate study, may force us to rethink the entire scope of the study (feedback loop to "Scope the Study")
- **Implement experimental wargame:** Implement the experimental wargame in a way to gather data.
- **Collect Data**: Collect data from numerous subjects.
- **Analyze Data:** Use traditional data analysis methods, such as regression, to identify what changed in the DV based on the IVs
- **Draw Conclusions**: Ties the results back to the research question.

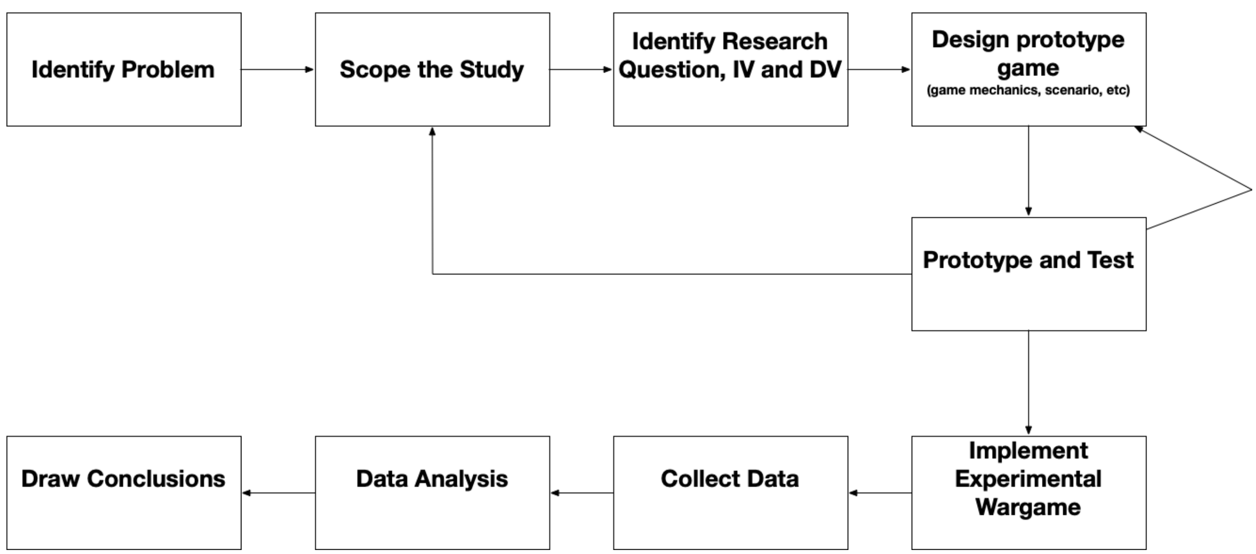

Figure 2: 9 step process for experimental wargame creation.

In order to draw reasonable conclusions in which one has confidence, there needs to be a significant amount of data. Standard methods in the experimental sciences, such as power analysis, provide you with estimates of the number of samples needed to have reasonable confidence in your results. The number of samples increases as the effect size is smaller. Effect size estimation is based on the IV and DVs chosen.

Experimental wargames, thus, have the problem of requiring many samples, which has naturally led to these types of games being constructed as distributed, online games. Distributed (where players are dispersed across the world) relaxes the requirement for geographical co-location, allowing for easier play. Online games, where players can play using their computers, facilitates data collection and connects people from across the globe. Experimental wargames and distributed wargames are tightly linked concepts.

In the following we will describe our experiences with designing and deploying our first experimental wargame, SIGNAL. In particular we will focus on the technical issues present in a distributed wargame. In Section 2 we provide a brief summary of the SIGNAL wargame. Section 3 describes the technical architecture of SIGNAL and Section 4 describes the opportunities and challenges we faced when developing SIGNAL.

2. THE SIGNAL EXPERIMENTAL WARGAME

SIGNAL[1] is a three-sided experimental wargame designed to study conflict escalation in a nuclear context. Players take on the role of a country's leader, in charge of military, diplomatic and economic decision making for the country. SIGNAL was implemented as a board game and an online game. Figure 3 shows the map for the online version of SIGNAL.

Players have access to action cards that allow them to execute various capabilities, including military actions (conventional, nuclear and cyber), and economic (build towns/cities). Players can interact with other players to make trades as well as negotiate treaties.

Data was collected about player behavior within the game, including what actions players took, when they took the actions, and their trades and communications with other players. This data, along with demographic data collected through a player survey, formed the backbone of the data set that was used to study conflict escalation patterns.

[1] Access signal here: https://www.signalvideogame.com/

Figure 3: SIGNAL-Online Map

3. TECHNICAL ARCHITECTURE OF SIGNAL

There are three technologies that form the foundation for SIGNAL.

- *Unity Game Engine* for the core game development,
- *Nakama* for the real-time network communication between game instances,
- *Amazon Web Services* (AWS) to host the game on the cloud.

In this section we will discuss these technologies, why we chose them, and how we've used them to build and deploy SIGNAL.

3.1. Unity

The Unity Game Engine is a popular choice for both entertainment and serious game development. An attractive feature for our use-case is a straight-forward and easy-to-use 2D development pipeline. A board game is well-suited to using a 2D top-down representation and SIGNAL's digital design benefited from this. Unity also relies entirely on a C# mid-level scripting language. This is a good choice for rapid and adaptable development as the design-build-test iteration is much shorter than if we required an extensive build process. Early in the development of SIGNAL as both the board game and e-game were developed simultaneously, the ability for the e-game team to quickly adapt to game rule changes with a reasonably efficient scripting layer was critical to keeping these scope changes manageable.

Finally, Unity can deploy to a very large number of platforms, and importantly for us the web via its WebGL compiler. Our intent from very early on was to make the game available on the web to reach the broadest possible audience using this ubiquitous deployment platform and requiring no app download or install to play the game. While it would have been possible to write the game entirely in native HTML5 and WebGL directly, we believe that borrowing much of that foundation from the Unity core was a big aid in reducing our overall resource cost. Additionally, because we built in Unity if at any point in the process, we found an insurmountable hurdle with web development such as resource limitation on the build or browser security requirements, we could potentially fall back to a more ordinary installable PC build of the game. This would have required compromising the goal of no download/install but was nice to have available if we discovered such insurmountable issues with the web deployment.

3.2. Nakama

Our choice in using Nakama was also driven primarily by our web-centered deployment. Nakama is one of a few game networking middleware that can target both a more normal TCP/IP socket communication as well as what are known as Websockets, a real-time socket-like networking protocol for web applications. Nakama hid the complexity of the underlying network transport layer from our development process and allowed us to focus solely on the game content necessary to keep the games synchronized for the three players. Again, this allowed us to flexibly rely on the web as a deployment vehicle but have confidence that we could fall back to a normal PC app if it became untenable.

Additionally, we relied heavily on Nakama's User Accounts and Matchmaking service to handle randomly assigning games to active players. This service works by having the game instance running in the player's browser request a match from the service and once it has enough requests come in for a game, it assigns a match. In our case as players launched the game from the website, the matchmaking would be engaged and once it found three such players it would notify all three and our code would kick in to start the game. There were some challenges in implementation of the matchmaking process, especially when one player was waiting for a long time for a game, but upon stabilizing this process it became a critical component in our ability to make SIGNAL work for the broad reach we sought.

3.3. Amazon Web Services (AWS)

AWS in the end served several purposes for our deployment. Early on, we used an Elastic Instance, a virtual server provided by Amazon in AWS parlance. We used this service in the beginning to support our development process. This server is where we ran the Nakama networking relay server and matchmaker for testing very early on in the process to ensure our networking and matchmaking worked. As we started to move toward deployment, AWS served as the host for the website that contained the game itself. This was served primarily as static content, but we could rely on Amazon's high availability infrastructure to deliver that content to browsers anywhere in the world very quickly.

The cloud database DynamoDB served as our logging backend. This required that one client instance from each running game would echo all messages to this DB as they happened in game. This was a very light-weight implementation through a REST API writing the messages in JSON format. Our aim was to make this process have minimal impact on the players' games and rely on our ability to post-process the database to extract useful data later. Each message included useful meta-data like the Game ID (a globally unique ID for each game provided by the Nakama matchmaker), the current round, phase, and player rotation. Internally, each message from each game was stored in essentially a huge, interlaced list of such messages. During post-processing we used the same REST API to pull and filter messages to determine relevant information about a given game, such as whether the game "went nuclear" and how early in the rounds.

One final important service provided by AWS that became critical to avoiding errors in our deployment was the use of the load balancing front-end. This layer was primarily used for its ability to proxy our content rather than necessarily balancing the load. By proxying our content we were able to keep all of the complexity surrounding the client's need to talk to the matchmaker, the websockets, the logger, etc. behind a presentation that looked like a single SSL connection to a single source. This helped us to mitigate several early challenges we faced with users' firewalls or browser security issues such as Cross-Origin Resource Sharing (CORS) errors. These are important security measures that should remain functional while still allowing users to play the game. Although there are still likely some edge cases we failed to find, proxying our content through the load balancer solved a large number of very apparent issues.

4. OPPORTUNITIES AND CHALLENGES IN DEVELOPING A DISTRIBUTED WARGAME

Creating the online version of SIGNAL posed a unique set of challenges unlike those experienced with traditional wargames. During the Project on Nuclear Gaming[2], we constructed a board game variant of SIGNAL and conducted trials familiar to the wargaming community: gathering subject matter experts together at a single location, where participants would receive detailed instruction on the wargame and spend considerable time focused on the task at hand. With experimental wargaming, we needed to reach a wider audience while still providing an engaging, carefully designed product where player behaviors remain true to form for the wargaming community. This section details challenges we encountered both in conceiving and implementing SIGNAL into a distributed, experimental wargame.

[2] https://pong.berkeley.edu/

4.1. Computer/Network Differences

Distributed wargames require mediation through computer-based tools, each of which have their own set of minimum computer hardware and network requirements. As with the video game entertainment industry, one must understand the audience in terms of how they will access the game, and how the variations in computer resources between participants can impact the overall game experience between players. A participant with a poor network connection or older computer hardware could potentially cause latency issues that disrupt the wargame experience. Participants may also run different computer operating systems or have Information Technology (IT) policies where they are unable to install external software on their computer. These issues together impact both the design and decisions in how participants access the distributed wargame.

The audience for the online version of SIGNAL offered a unique challenge by needing the game as widely accessible as possible. To solve this challenge, SIGNAL online was developed as a web browser based game, with considerations made on what web-browsers participants may have installed and which ones could support the SIGNAL software. As a web-based application, participants could readily access SIGNAL without needing to install additional software.

As a networked application, SIGNAL required methods for gracefully handling instances where participants unexpectedly leave the wargame due to computer-related issues (e.g., network disconnection or interruption, computer crash). Due to experimental design choices, SIGNAL does not include video or audio channels where participants can notice if another has left the game. SIGNAL borrowed design patterns from commercial online video games to both notify participants if one unexpectedly left the wargame and give them choices on how to proceed. SIGNAL pauses the wargame once it determines a participant is no longer connected to the wargame session and provides players with the option to either wait for the participant to reconnect or remove them from the game, replacing that participant with a computer-controlled player.

4.2. Asynchronous Play

Distributed wargaming offers a tremendous advantage over traditional wargaming events through enabling participants to engage in asynchronous play: the ability for wargame participants to play outside of an organized event. This reduces the burdens of participant schedules and financial expenses normally incurred bringing participants together in-person. Though appealing on the surface, in-person events provide some intangible benefits not appreciated when moved to an online forum. In-person events require more investment from participants in both their time and attention, along with providing a more natural connection with other participants that increases the likelihood they will fully immerse themselves in the task at-hand. These natural connections also extend to engaging with the adjudicators and moderators, enabling participants to quickly receive clarifications on questions that arise as they play.

The online version of SIGNAL handled these issues with asynchronous play through both its design and introductory materials. Through these considerations, SIGNAL reflected more of a commercial video game than other wargame designs. SIGNAL had a fixed ruleset with limitations on potential actions and outcomes, removing the need for adjudicators to interpret participant choices. SIGNAL utilized time-sensitive actions, adding suspense and pressure within a participant's decision-making to keep them focused. SIGNAL used visual and audio aesthetics reflective of popular video games (e.g., Civilization) to entice participants to stay engaged and give them analogies that facilitated quicker learning of the user interface. SIGNAL used other commercial video game practices such as online tutorial videos posted to YouTube and an interactive, single-player in-game tutorial to help participants familiarize themselves with the game before playing with other participants.

With asynchronous play, the wargame may need to avoid using external audio or video communication channels (e.g., Zoom, Discord). Instead of removing participant conversations, a key advantage with in-person events, SIGNAL incorporated an in-game chat functionality where participants could communicate. Design choices in SIGNAL such as resource trading motivated participants to take advantage of the in-chat functionality. SIGNAL also provided the option for participants to chat with everyone simultaneously (public messages) or elect to chat with another participant privately. This enabled SIGNAL participants to engage in closed-door negotiations and acts of deception, providing a richer set of behaviors not easily achievable using a similar game within an in-person event.

4.3. Building a Player Population

Distributed wargames can reach a wider audience beyond in-person events, creating a unique ability necessary in transforming wargames into experimental platforms. By enabling more potential participants to the wargame, one can gather data from multiple wargame sessions to observe how varying conditions may influence participant decision-making. This enables wargame designers to examine research questions with more statistical rigor than provided by traditional wargames.

Although a distributed wargame can be as easily accessible as a publicly available website, attracting participants is not automatically guaranteed. In SIGNAL, three participants were required for the wargame session to begin. Since participants could join at any time, one could assume a participant could enter the SIGNAL website and find other participants also ready to play. Yet, SIGNAL received many complaints from participants who waited for several minutes without joining a session due to lack of participants on the website at the time they wanted to play. Though the SIGNAL audience base totaled in the thousands, it was too small to ensure that even three participants across the globe would be logged into the SIGNAL website simultaneously.

Boosting SIGNAL website activity relied on borrowing tactics used in the commercial video game space to attract a player base. Play campaigns, organized through social media outlets and mailing lists, gave dates and times for potential participants to join. Some SIGNAL team members would serve “play duty”, receiving an automated message to login to the SIGNAL website if at least one participant was online searching for others to play. SIGNAL play sessions were informally organized (e.g., Albuquerque game developer meetups, campus events at University of California, Berkeley and the University of New Mexico) – requiring far less preparation than traditional wargaming events due to the SIGNAL design choices outlined in the previous section. Though most wargaming sessions recorded for SIGNAL research studies came through financially compensated participants, approximately one-third of the player population were unpaid members of the public who gained exposure to SIGNAL through these outreach activities.

REFERENCES

Lakkaraju, Kiran, Jason Reinhardt, Joshua Letchford, Jonathan Whetzel, Bethany L. Goldblum, and Andrew W. Reddie. 2020. “Experimental Wargames to Address the Complexity: Scarcity Gap.” In *Proceedings of the 2020 Spring Simulation Conference*, 1–12. SpringSim ’20. Fairfax, Virginia: Society for Computer Simulation International.

Maoz, Zeev, Paul L. Johnson, Jasper Kaplan, Fiona Ogunkoya, and Aaron P. Shreve. 2019. “The Dyadic Militarized Interstate Disputes (MIDs) Dataset Version 3.0: Logic, Characteristics, and Comparisons to Alternative Datasets.” *Journal of Conflict Resolution* 63 (3): 811–35. https://doi.org/10.1177/0022002718784158.

Perla, P., and J. Curry. 2011. *Peter Perla’s the Art of Wargaming a Guide for Professionals and Hobbyists*. Lulu.com. https://books.google.com/books?id=t1xgLwEACAAJ.

Reddie, Andrew W., Bethany L. Goldblum, Kiran Lakkaraju, Jason Reinhardt, Michael Nacht, and Laura Epifanovskaya. 2018. “Next-Generation Wargames.” *Science* 362 (6421): 1362–64. https://doi.org/10.1126/science.aav2135.

Sabin, Philip. 2014. *Simulating War: Studying Conflict through Simulation Games*. Reprint edition. London Oxford New York New Delhi Sydney: Bloomsbury Academic.