ARMY RESEARCH LABORATORY

Literature Review on Systems of Systems (SoS): A Methodology With Preliminary Results

by Jeffrey A. Smith and Brian G. Ruth

ARL-TR-6734 **November 2013**

NOTICES

Disclaimers

Army Research Laboratory

White Sands Missile Range, NM 88002-5513

ARL-TR-6734 **November 2013**

Literature Review on Systems of Systems (SoS): A Methodology With Preliminary Results

Jeffrey A. Smith and Brian G. Ruth
Survivability/Lethality Analysis Directorate, ARL

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED (From - To)
November 2013	Final	January 2011–May 2013

4. TITLE AND SUBTITLE

Literature Review on Systems of Systems (SoS): A Methodology With Preliminary Results

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Jeffrey A. Smith, Brian G. Ruth

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

U.S. Army Research Laboratory
Information and Electronic Protection Division
Survivability/Lethality Analysis Directorate (ATTN: RDRL-SLE-G)
White Sands Missile Range, NM 88002-5513

8. PERFORMING ORGANIZATION REPORT NUMBER

ARL-TR-6734

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

U.S. Army Research Laboratory
2800 Powder Mill Road
Adelphi, MD 20783-1145

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The modeling and analysis of large and complex systems is a rich and diverse field of study, and arguably the modeling and analysis of Systems of Systems (SoS) falls squarely within this field. Complex systems range from air traffic control and traffic systems, to urban environments and industry supply/inventory systems, as well as the complex systems manifested in military operational environments. Each complex system presents unique modeling and analysis challenges, and it is quite unlikely that one can address these challenges by one modeling tool or environment. Our motivation in authoring this report owes itself in part to the well-intentioned meaning of researchers who suggest their tool, or environment, as the tool by which the Survivability/Lethality Analysis Directorate (SLAD) of the U.S. Army Research Laboratory (ARL) can meet its analytic needs for military systems of systems analysis (SoSA). Our goal is a rational basis by which, one can consider the recommendations of other researchers, without making what may be a considerable investment of time to study in depth the modeling tools and environments that they have developed. After reviewing the literature on SoS and SoSA—as well as literature more specific to the modeling and analysis—we developed a framework to measure the suitability of a particular modeling tool, or environment, to meet SLAD's need to conduct survivability, lethality, and vulnerability analyses (SLVA) of military SoS. This report documents our methodology and some initial applications.

15. SUBJECT TERMS

system of systems; SoS, SoSA, system of systems, system of systems analysis, SLV, SLVA, complex systems, systems analysis, military system of systems, military SoS, agent modeling, agent model, framework, methodology, assessment

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
					Jeffrey A. Smith
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPHONE NUMBER *(Include area code)*
Unclassified	Unclassified	Unclassified	UU	110	(575) 678-1332

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18

Contents

List of Figures

List of Tables

Acknowledgments

The authors would like to acknowledge Dr. Romeo del Rosario of the U.S. Army Research Laboratory (ARL) Sensors and Electron Devices Directorate (SEDD) for his innovative design of the quadrant model methodology utilized within this report.

1. Introduction

The mission of the Survivability/Lethality and Analysis Directorate (SLAD) within the U.S Army Research Laboratory (ARL) is to provide survivability, lethality, and vulnerability analyses (SLVA) and expert consultation to its customers. Among these important customers are the Army's independent evaluator Army Test and Evaluation Command (ATEC), program managers (PMs), and Army decision makers. Traditionally, SLVA focused on single-thread analyses that characterized the interaction between a single item of equipment and one or more threats as if that interaction took place in isolation from its military context. Although SLVA of individual items remains important, it is no longer sufficient to address the newer technical and business concerns of many SLAD customers—concerns inherently at the system of systems (SoS) level. Army and defense leadership is intent on fielding a network-enabled force and acquiring complex packages of military capabilities that will support the full range of Force Operating Capabilities.* Comprehensive analysis of these packages requires us to portray the results from subtle engineering interactions among different systems in the capability packages. Thus, we must consider the whole SoS in our analyses (*2*).

As SLAD develops its ability to conduct SLVA in the context of SoS (*3*), one question that often arises is—how does SLAD relate its work to that of others who are also addressing SoS issues, or developing methods to conduct SoS analyses (SoSA)? The intent of this report is to present a methodology with which one can survey the available literature, and to use this methodology to conduct a preliminary survey to assess the efficacy of this approach in answering this question. We structure this report into three principle sections. Section 2, starts from the available literature on SoS to develop criteria that will inform our survey efforts. In section 3, we develop a quadrant model that is particular to SLAD's SoS/SoSA needs, which we will use to obtain the preliminary results given in section 4. Finally, section 5 discusses some conclusions and presents a tentative path forward. An appendix is included for each of the various approaches we surveyed, along with a detailed breakdown of the scores for that approach and other relevant information.

2. Background

As a key component of this literature review, we will develop a framework with which to evaluate and score the literature that will populate our review. In section 2.1, we establish the

* Training and Doctrine Command (TRADOC) Pamphlet 525-66 (*1*) entitled, "Force Operating Capabilities" discusses the required capabilities in detail.

necessary background for our framework based on a survey of the academic literature. Here we draw on the general literature regarding SoS to garner insight into key characteristics.
In section 2.2, we identify aspects of SoS that are prototypical of military SoS and adopt a particular view of SoS/SoSA. Finally, in section 2.3, we discuss some implications that are relevant to SLAD's mission of conducting SLVA of military SoS.

2.1 Academic Background on SoS

We first define the characteristics of an idealized Army-appropriate design of a military SoS and associated SoSA. The first set of attributes associated with our definition is drawn from five principal characteristics that most authors (*4*,*5*) think an SoS should possess (figure 1). Boardman and Sauser (*6*) enumerated and described these characteristics as table 8.1 (*6*). In the paragraphs below, we use the terminology of Hitchens (*7*) and annotate via, "[…]" in places where we equate Hitchens with Boardman and Sauser. We also expand their terminology here and slightly expand upon their ideas so that our choices for coordinates in the quadrant model of section 3 become more obvious.

1. *Autonomy*: The ability to make independent choices; the right to pursue reasons for being and fulfilling purposes through behaviors. This motivation arises via the indispensability of legacy systems to the functioning of an SoS. However, an SoS also possesses a higher purpose than any of its constituent systems, either taken independently or additively, an insight that is key to the military SoS we wish to study. Concerning the constituent systems and the SoS, autonomy mandates that they be both:

   - operationally independent,
   - managerially independent.

2. *Social Connectivity (Belonging)*: Constituent systems within the SoS are operationally bound together in secure, cooperative, and often complementary relationships. However, legacy systems may need to undergo (possibly radical) doctrinal changes in order to effectively serve within an SoS. With regards the relationship between the constituent systems and the SoS, belonging implies that the constituent systems:

   - share a mission context in a purposeful way,
   - understand their role in the purpose of the SoS.

3. *Material Connectivity (Connectivity)*: The ability of a system to link with other systems. Since legacy systems that contribute to an evolving SoS design are usually heterogeneous systems, they are unlikely to conform to *a priori* connectivity protocols. Given that the SoS places a huge reliance on effective connectivity within a dynamic theater of operations, guaranteeing intraSoS networked connectivity throughout a mission becomes critical. Materiel connectivity implies that the constituent systems are:

- interdependent and interoperable,
- distributed and networked.

Furthermore, there are many possible arrangements of the constituent systems; thus, there may be many ways to achieve the purpose of the SoS.

4. *Diversity*: An SoS displays noticeable heterogeneity among its constituent systems, for example, it demonstrates distinct or unlike capabilities and behaviors; consequently, the SoS can only achieve its higher purpose(s) by effectively leveraging the diversity of its constituent systems. For these constituent systems, diversity implies that:

- individually, they may all possess a unique identity;
- collectively, they will all comprise a heterogeneous whole.

5. *Emergence*: The behavior of the interacting systems in an SoS determines the collective behavior of the SoS; the term *emergence* characterizes these behaviors. While some of these emergent behaviors may be an intended consequence of design, others may be serendipitous or undesirable. Emergence requires a well-defined boundary, and while an SoS may possess dynamic boundaries, these boundaries are always clearly definable. Therefore, the SoS develops an emergent operational culture with enhanced agility and adaptability during the course of realizing its purpose. The term emergence also implies that the SoS:

- evolves,
- possess a collective intelligence,
- exhibits synergy among constituent systems,
- functions in a dynamic environment,
- adapts to the local and novel conditions.

As Boardman and Sauser, Hitchens, and others use these terms, they take these characteristics in a collective sense such that, an SoS will—to varying degrees—display each of these five characteristics. Furthermore, if one or more of these properties is either, not in a system, or only weakly observed within that system, these same authors would likely conclude that the system is not an SoS. In a final note, while there is no one "established" set of characteristics, our reading of the literature suggests that one can interpret other characteristics (*8, 9*) as the consequence of Boardman and Sauser's criteria interacting in a particular environment.

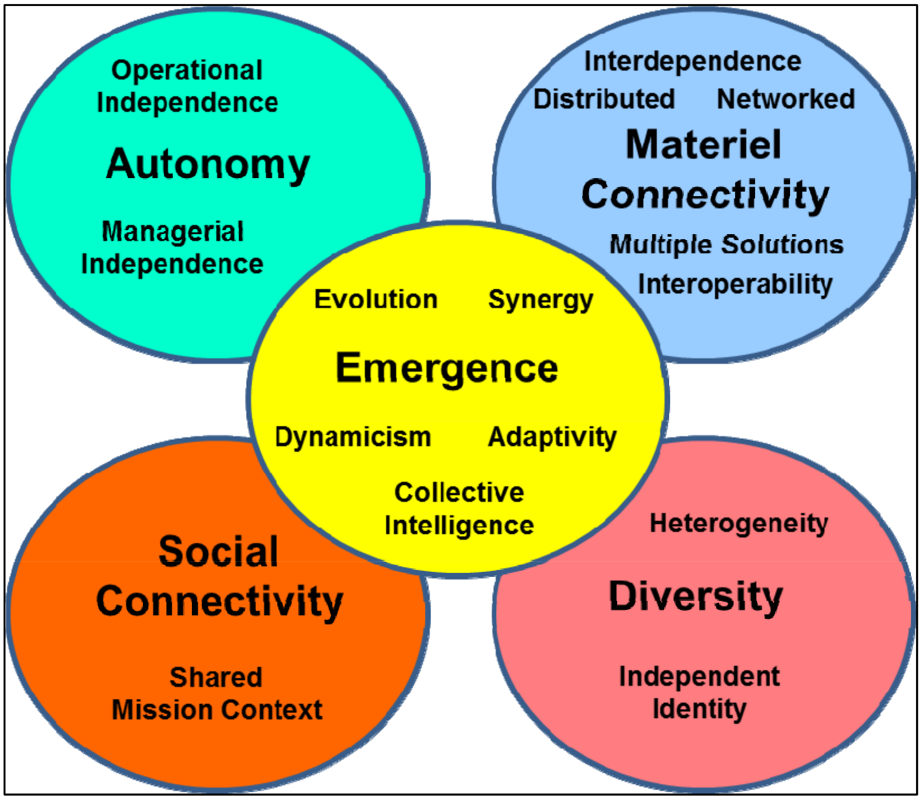

Figure 1. Venn diagram representation, as derived from Hitchins' generic reference form model (*7*), of Boardman and Sauser's principal characteristics that define an SoS.
Note: Here, a general SoS reflects the union of all sets of characteristics.

2.2 Military Background on SoS

Next, we review some of the military—particularly Army-centric—literature regarding SoS (*6, 10–12*) and use this literature to inform the discussion we began previously (*13, 14*). In the paragraphs that follow, we establish our view of what constitutes a military SoS. In previous literature (*13, 14*), we identified three concepts we believe are present in any military SoS:

- they are purposeful systems that undertake a defined mission,
- they operate and evolve in dynamic environments,
- they are sociotechnical systems.

The following paragraphs can be read from an all-encompassing point of view, we employ military SoS in a wide range of mission contexts (i.e., humanitarian to force-on-force conflict). However, SLAD's mission is assessing the survivability, lethality and vulnerability (SLV) of current and future forces; therefore, we focus our thinking and discussion on those missions that place Soldiers in peril (i.e., Soldiers facing hostile forces). This choice does not limit our argument in any way; in fact, it serves to make some of the points more easily understood. Therefore, by focusing on military SoS given a particular mission—i.e., to protect a village or enable the safe passage of noncombatants and supplies down a critical road— we can, where needed, draw exemplars from current operations that allow us to explain our ideas with some degree of concreteness. In the following paragraphs, we discuss each of these concepts from the associated perspectives of SoS and SoSA.

At the very core of a military SoS, in the sense that we employ the term, exists two principle reasons for being, (1) a *mission*, or in simple terms "what they are to do," and (2) a *purpose*, or "why they are doing what they do." For any military entity, from the lowly private to the combatant command, purpose drives the choices they make as they conduct their mission. Thus, the commander of an infantry brigade combat team (IBCT) receives a purpose from his commander, and that inviolable purpose motivates the IBCT commander's mission-planning and subsequent actions, as they relate to peer and parent units. For example, that purpose might be to protect another unit as it clears a village and his mission may be to keep his forces between the unit he is protecting and his adversaries. Prototypically, the commander communicates his mental image of the purpose—decomposed in time—to his subordinates and peers as either intent, desired end state, mission, or a series of actions framed as a concept of operations. Practically, this communication is bidirectional in the sense that his subordinates must also understand that purpose to a significant degree. To undertake this communication, the commander makes many choices in order that he may successfully share his vision for a concept of operations; the Army's preferred method for this communication is "mission command" (*15*), which prescribes "why" an action takes place, but not the "how" of that action. General DePuy, a former Commanding General of the U.S. Army's Training and Doctrine Command (TRADOC), suggested one can think of the seminal act of creating a "concept of operations" as a decomposition of mission into subordinated concepts related by purpose (*16*). DePuy described this decomposition as *nested concepts and inter-related purposes*, wherein the purposes do not control, but merely constrain the subordinate commander's action (*15, 17*), in order to maintain a unity of effort throughout the SoS. To support this shared understanding of *commander intent* throughout the SoS, the *concept of operations* is a statement that directs the manner in which subordinate units cooperate to accomplish the mission, and it establishes the sequence of actions the force will use to achieve the end state. This serves to establish within the subordinate what Ackoff and Emery term *purpose-oriented relationships* (*18*) with the commander's unifying image, and informs them of their responsibility to attain the common goal.

From the above discussion, it should be obvious that military SoS, especially when SLV issues are concerned, function in a dynamic environment that evolves over time. When facing a similarly inclined adversary, it is the time-dependent, purposeful interplay between the missions of parent, peer, and subordinate units as they respond to uncertainty and change (internal and external), which ultimately determines success or failure in achieving the commander intent. Therefore, we are led to the observation that we cannot view a military SoS simply as a purely mechanical system such as a "network," or as a purely human system; we must adopt the viewpoint that a military SoS is fundamentally a *sociotechnical system* (*19, 20*).

Sociotechnical systems theory studies the interactions between the human and technical systems; asserting that it is the human/social component of the system that applies knowledge to satisfy local variations through the effective use of available technology. Thus, while the technological components of a system may have a small range of adaptation, it is the people in the system that are ultimately the source for a wide range of adaptations—and for a military SoS, this is the norm.

Figure 2 shows our graphical representation of the discussion in this section. In figure 2, the three nodes at the terminus of the arrows emanating from the "military SoS" represent the three existential characteristics that we view are essential elements of any discussion on military SoS, not to include the analysis thereof. In summary, we note that military SoS are constructs that:

- possess a collective decision-making process (DMP), which is modulated via the deployment of nested concepts and inter-related purposes (utilizing the concepts of commander intent, concept of operations, and purposeful systems);
- exist in a dynamic environment and whose actions contribute manifestly to the evolution of that environment (in turn demonstrate synchronous behaviors, multiple time scales, and nonstationary/nonergodic behaviors);
- apply concepts described with sociotechnical systems theory.

We also observe that the paradigm of:

- nested concepts and inter-related purposes both generates dynamic and evolving processes, and takes advantage of concepts from sociotechnical systems theory, while
- sociotechnical systems theory contribute to the generation of dynamic and evolving processes.

As this framework illustrates, the three existential concepts defining the nature and function of a military SoS are intercorrelated, constructive, and complementary by design.

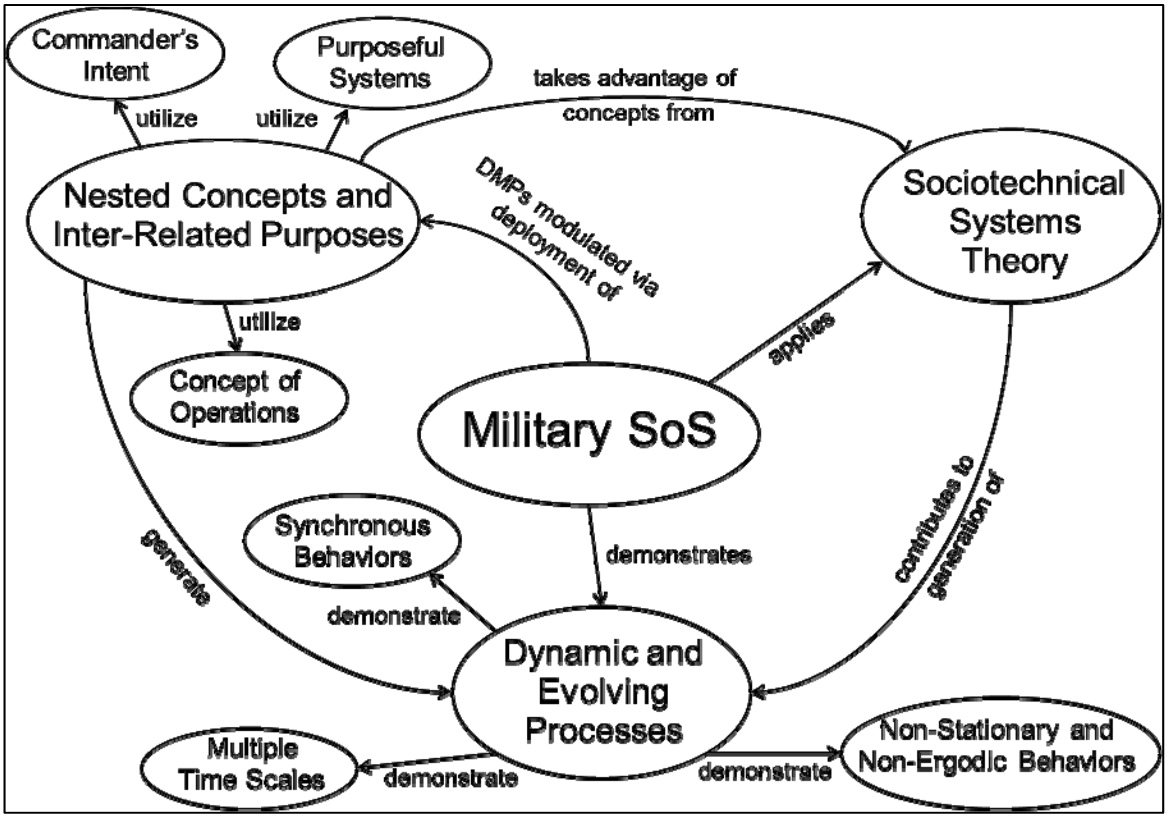

Figure 2. A framework that inter-relates the existential characteristics of military SoS.

2.3 Analyzing the Military SoS

In section 2.1, we surveyed the broad expanse of academic literature on SoS to arrive at some ubiquitous and essential characteristics of SoS, and section 2.2 established our view of military SoS and constructed a framework (figure 2) with which to view the inter-relationships among the ideas discussed in sections 2.1 and 2.2. However, we must also come to grips with analysis of these SoS, and most importantly, that of the military SoS. To do so, we will ultimately extend the idea of figure 2 to include some of the major analytic approaches one can employ.

From the lens of sociotechnical systems theory, analysis of military SoS recognizes that the behavior of a military SoS results from the purposeful coordination of the parent, peer, and subordinate unit missions in order to realize commander intent. Immediately, this suggests that the analytic problem is one not only of a "horizontal," or breadth sense scope, but also of a vertical scope through the command hierarchy. The number of constituent systems that must act together to complete a collective SoS task is referred to as the *scale* of the task. *Variety* is a measure of the distinct actions that the SoS can collectively take to complete a task. *Multiscale analysis of complex systems* builds on the twin recognitions that scale and variety/complexity are necessary for the effective performance of SoS (*21*), a notion that is especially true of military sociotechnical systems. However, at whatever level units exist, units within a military SoS must respond to uncertainties within their environment to survive. This suggests that any multiscale analysis technique applied to these unfolding sociotechnical processes must also account for their natural nonstationary and nonergodic properties, as well as their adaptations to address uncertainties. Furthermore, because the military SoS of interest are not static, but rather, dynamic evolving systems in nature, any analysis must have at least elements that respect the time-series nature of the system representation.

In classical analytical approaches, one undertakes analysis via decomposition using established orders, such as an organizational structure. Here, the intent is to explain the whole via inferences drawn at first from the explanation of the defined parts as implied by the established order. Alternatively, one could undertake the analysis via a *synthetic approach* that seeks to deduce the functionality of a component via its relation to the whole. While both approaches have their use in the analysis of military SoS, they will miss key aspects of the SoS that are important for SLV analyses. Military SoS are created with the intention that its emergent behavior realizes an intended mission, and in this sense they become more than the sum of its parts; yet, if these SoS are improperly taken apart for analysis they may lose this important property.

In figure 3, we map these analytic approaches onto the framework of figure 2 to show their inter-relationships. This particular analytic view uses DePuy's notion of nested concepts and inter-related purposes, in conjunction with commander intent, and concept of operations—in order to decompose the SoS for analysis. We then properly integrate the results into a holistic analysis of the SoS. Within DePuy's notion, the role purpose takes is to provide a constant element by which we can gauge action. In mission command, the purpose of a given unit is inviolable, which means that the purpose of a unit can only be changed by an order—by that unit's commander; thus, creating a measurable time-series event. We note that the tasks chosen to accomplish a mission can change considerably during that mission; however, purpose is the one fundamental constant, both vertically and horizontally, and thus forms the basis by which one can understand the time-varying evolutionary nature of the system.

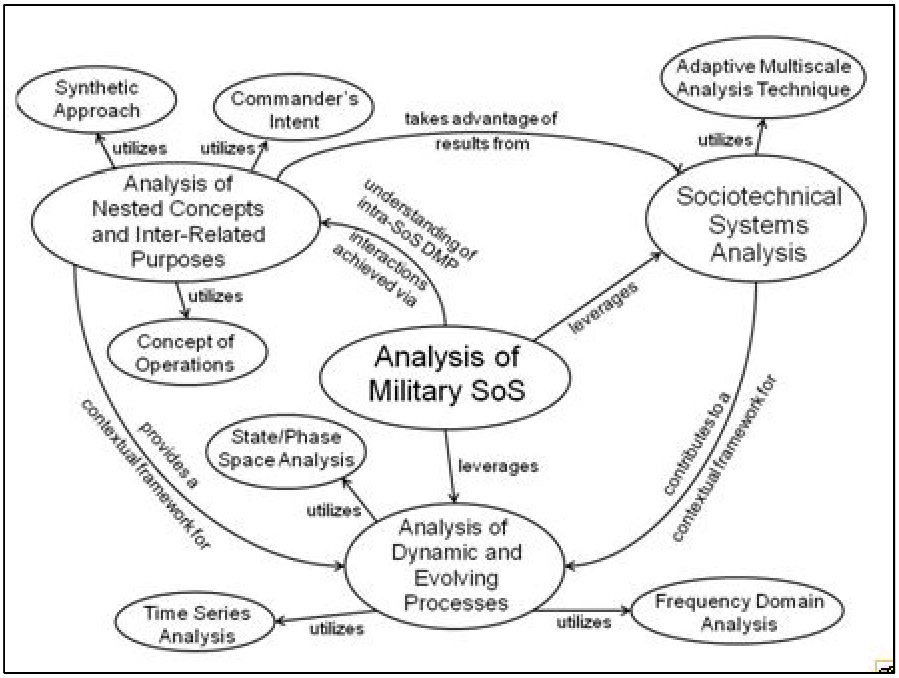

Figure 3. Extending the framework of military SoS to include analysis.

3. Methodology

Now that we have defined distinct, yet related, SoS and SoSA frameworks to serve as technical guideposts in preparation for our literature review, we next describe a method with which to evaluate the reviewed literature relative to SoS and SoSA design standards as represented in the ontologies.

We base our methodology on a modified version of the supply categorization quadrant model as first proposed and used by del Rosario et al. (*22*), in their comprehensive assessment of technical requirements associated with achieving a logistics-oriented prediction and pre-emption capability (*23*). In our implementation of their model, we rank the ability to model SoS using a given tool (simulation, model, etc.), and the ability to conduct SoSA employing that model in terms of:

1. The overall similarity of an evaluated SoS or SoSA design to the reference standards reflected by the SoS and SoSA framework conveyed in figures 2 and 3, respectively.
2. The overall level of technology maturation of an evaluated SoS or SoSA design relative to an estimated developmental threshold level, wherein a design would be mature enough for potential integration into the ARL SoS modeling and analysis business process.

3. The overall accessibility (for the purposes of conducting ARL business) of an evaluated SoS or SoSA design as a function of both technology cost and general availability to ARL.

Note that, from this point on, our use of the term "SoS" will be specifically in reference to a model (either software- or test-data-based) of a physical SoS, while "SoSA" will refer to the tools, techniques, and methodologies (TTM) associated with analysis of data generated by the SoS model relative to an operational context.

To assess the degree of similarity between a given model, tool, or simulation capability and the frameworks we developed in sections 2.2 and 2.3, we shall use the criteria of table 1. Here, each criterion is scored as a 1, 2, or 3; where 1 is taken to mean a low similarity on those criteria, 2 a medium degree, and 3 a high degree. We will weight each criterion with an integer value between 1 and 10, inclusive. The similarity score itself is computed as the weighted sum of the scores, normalized by the weight, or

$$Similarity = \frac{1}{\sum_{i=1}^{N(\text{attr})} w_i^{\text{attr}}} \sum_{i=1}^{N(\text{attr})} w_i^{\text{attr}} * s_i^{\text{attr}} \quad (1)$$

To assess the degree of maturity of a given model, tool, or simulation capability, we shall identify the major components of that model, tool, or capability and score them as a degree of maturity of 1, 2, or 3. Here we take 1 to mean a low degree of maturity of that particular component, 2 an interim—or moderate—degree of maturity, and 3 a high degree of maturation, or of direct benefit to ARL. The maturity score itself is computed as the weighted sum of the scores, normalized by the weight, or

$$Maturity = \frac{1}{\sum_{i=1}^{N(\text{comp})} w_i^{\text{comp}}} \sum_{i=1}^{N(\text{comp})} w_i^{\text{comp}} * s_i^{\text{comp}} \quad (2)$$

In most all cases, for the maturation weight we will use a value of 10 to indicate that all components used to evaluate maturity are equally desirable.

Table 1. The criteria, weights, and scoring ranges used to develop the similarity score for rating a tool's ability to model an SoS, and any related tools ability to conduct SoSA.

(a) System of systems				
	Attribute			
	Description	Number	Weight	Score
Nested Concepts and Inter-Related Purposes →	Commander Intent	1	10	1,2,3
	Concept of Operations	2	10	1,2,3
	Purposeful Systems	3	7	1,2,3
Dynamic and Evolving Processes →	Synchronous Behaviors	4	10	1,2,3
	Multiple Time Scales	5	7	1,2,3
	Nonstationary and Nonergodic Processes	6	5	1,2,3
	Sociotechnical Systems Theory	7	7	1,2,3
(b) System of systems analysis				
Analysis of Nested Concepts and Inter-Related Purposes →	Commander Intent	1	7	1,2,3
	Concept of Operations	2	7	1,2,3
	Synthetic Approach	3	10	1,2,3
Analysis of Dynamic and Evolving Processes →	State-Space Analysis	4	5	1,2,3
	Time-Series Analysis	5	10	1,2,3
	Frequency-Domain Analysis	6	5	1,2,3
Sociotechnical Systems Analysis →	Adaptive Multiscale Techniques	7	7	1,2,3

To score accessibility (for the purposes of conducting ARL business) of an evaluated SoS or SoSA design, we shall use one of three distinct colors:

- *Green* indicates the SoS/SoSA technology is (1) currently owned by the government and easily accessible to ARL, or (2) is an open-source product.
- *Yellow* indicates, (1) the technology is government-owned, but ARL access to this technology might be dependent upon some formal condition (e.g., memorandum of understanding between ARL and the technology developer), or (2) accessibility status of the technology is currently unknown.
- *Red* indicates the technology is either (1) privately developed by industry with considerable costs associated with government use and/or access, or (2) the technology is currently unavailable to ARL.

Therefore, this evaluation approach allows an SoS, or SoSA design of interest, to be represented by the descriptive triad (Similarity, Maturity, Accessibility), that can in turn be plotted on a quadrant chart.

In our usage of this quadrant model, we score SoS and associated SoSA designs for a technology to arrive at a particular scoring triad. The values in this triad determine that design pair's position on a quadrant chart of the form presented in figure 4, wherein we report the SoS and associated SoSA designs separately. In figure 4, we have notionally plotted assessments of two different tools to model a particular SoS of interest and the tools available with which to conduct a modeling analysis. It is easy to see the comparative value of this approach for assessing the viability of a range of tools with which one might model an SoS of interest—as well as the associated tools. In the remainder of this report, we apply this technique to tools in literature.

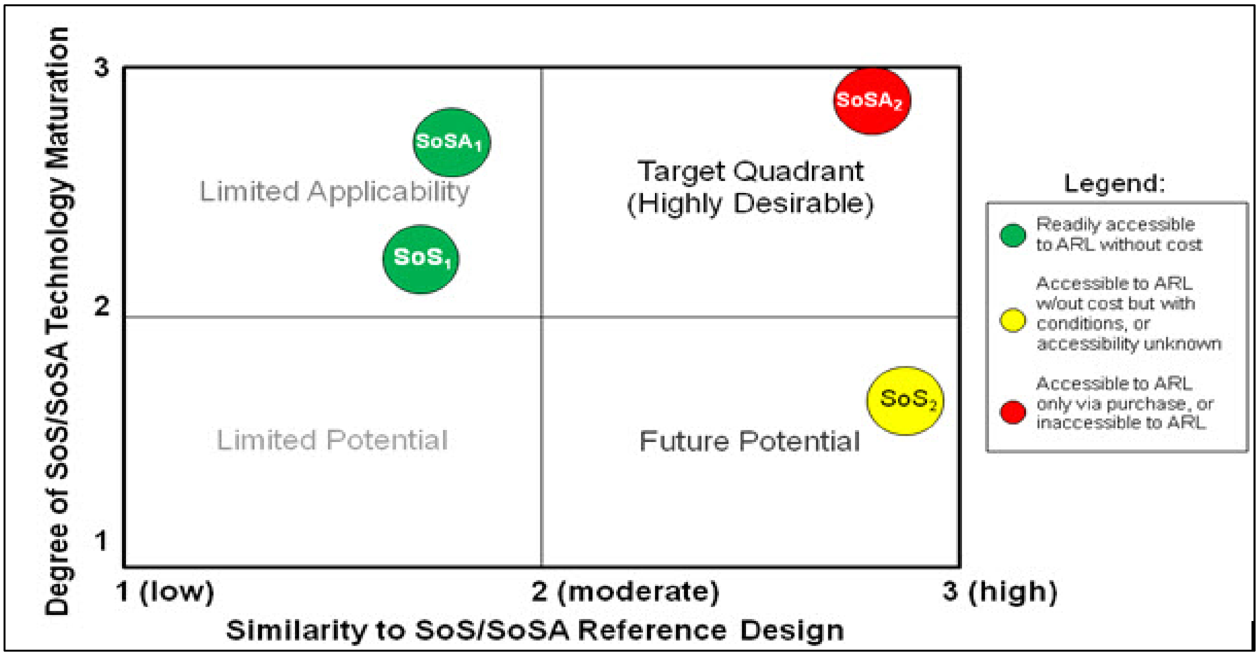

Figure 4. Quadrant chart addressing SoS and associated SoSA designs for two notional SoS/SoSA technologies.

4. Results

Using the methodology we developed in section 3, we surveyed the literature to identify a broad spectrum of technology; our goal was one of breadth as opposed to a specific aim to find the "one best" technology. Although we are interested in modeling military SoS, we chose not to consider exclusively those technologies most closely aligned with military modeling and simulation; thus, we also chose some technologies that were either more mathematically based, or devoted to urban planning, for example. We arrived at a list of 10 different modeling tools—or technologies—as well as any defined analysis packages associated with their respective modeling tools. While we discuss each of these tools in depth in a corresponding appendix, here we present a brief introduction to each tool set, noting that the letter corresponds to the appropriate appendix.

A. The CityDev model was developed by Semboloni to study the dynamics of simulated urban development within a decentralized SoS consisting of developers, industrial firms, commercial firms, service providers (public and private sector), and households of consumers/laborers (*24–27*).

B. As part of the ARL Advanced Decision Architectures Collaborative Technology Alliance (ADA CTA), Chandrasekaran and Josephson of the Ohio State University's Laboratory for Artificial Intelligence Research (LAIR) developed a multicriterion decision technology called the, *Seeker-Filter-Viewer* architecture (S-F-V) (*28*).

C. The *Political Science-Identity* (PS-I) agent-based computer simulation platform was originally developed by Lustick and various colleagues at the University of Pennsylvania to operationally simulate, refine, and test competing versions of political constructivist identity theory (*29*).

D. The U.S. Air Force has been developing the *Wargame Construction Toolset* (Warcon) that would empower Air Force instructors to create small-scale instructional wargames that embody modern military doctrine and war-fighting principles, including Network-Centric Warfare (*30*).

E. The *Measures of Tipping Points, Robustness, and Path Dependence* (MTPRPD) methodology is a novel approach created by Bramson (University of Michigan) to address the analysis of complex system dynamics in terms of "tipping points," system robustness, and system state-space path dependence (*31–33*).

F. An easy to use, agent-based modeling environment called *Pythagoras*, focused on the simulation and analysis of both human factors in military combat and noncombat situations (*34*).

G. The *Peace Support Operations Model* (PSOM)—a faction-to-faction, time-stepped, cellular geography, semi-agent-based model—that was initially designed to represent a range of civil and military aspects of Peace Support Operations (*35*).

H. The *Enhanced ISAAC Neural Simulation Toolkit* (EINSTein) is an adaptive agent-based model of land-based military combat that evolved out of a more far-reaching project to develop a new fundamental theory of warfare based upon Complexity Theory (*36–40*).

I. *Map Aware Non-Uniform Automata* (MANA) is an agent-based model developed by the Operations Analysis team at the Defence Technology Agency (DTA) in New Zealand (*41–46*).

J. The *Beijing National Defense University (BNDU) Agent-Based Model (ABM)* is a proposed new method of demonstrating SoS weapon and combat equipment simulation based on the theory of complex adaptive systems (CAS), to meet the needs of realistic information warfare simulation and analysis (*47, in Chinese. Translated April 28, 2011*).

We assessed each of the above tools for their ability to model an SoS consistent with our needs to model military SoS, and we have summarized the results of that assessment in table 2. We also assessed any ancillary tools that were available to these modeling environments for their ability to conduct SoSA. Table 3 summarizes these results.

It is clear from inspecting the data in tables 2 and 3, our methodology produces a reasonable spectrum of results. Furthermore, as each of these modeling tools were developed for purposes other than modeling a dynamic military SoS, the relatively midrange similarity scores seem appropriate. It is not surprising that the overall maturity scores tended towards the higher ranges, given that each of these modeling environments has been discussed in the open literature, and therefore should be expected to be reasonably mature.

Table 2. Summary of the modeling tools and techniques we evaluated to model an SoS.

Appendix	Name	Similarity	Maturity	Accessibility
A	CityDev	1.68	3.00	Red
B	S-F-V	2.32	2.67	Green
C	PS-I	1.98	2.82	Green
D	Warcon	1.89	1.75	Yellow
E	MTPRPD	Not feasible to model an SoS		
F	Pythagoras	1.88	2.67	Yellow
G	PSOM	1.82	2.25	Yellow
H	EINSTein	1.27	3.00	Yellow
I	MANA	1.88	3.00	Red
J	BNDU	2.23	3.00	Red

Table 3. For each modeling environment assessed for modeling an SoS, our assessment of their ability to conduct an SoSA using their tools.

Appendix	Name	Similarity	Maturity	Accessibility
A	CityDev	1.69	3.00	Red
B	S-F-V	2.80	3.00	Red
C	PS-I	2.31	2.60	Green
D	Warcon	1.00	1.00	Yellow
E	MTPRPD	2.06	1.00	Green
F	Pythagoras	1.39	3.00	Yellow
G	PSOM	1.67	2.00	Yellow
H	EINSTein	2.08	2.50	Yellow
I	MANA	1.39	3.00	Red
J	BNDU	No data with which to assess		

5. Conclusions and Summary

Large and complex systems modeling and analysis is a rich and diverse field of study, and arguably the modeling and analysis of SoS falls squarely within this field. Complex systems range from air traffic control and traffic systems, to urban environments and industry supply/inventory systems, as well as how the military considers complex systems. Each complex system presents unique modeling and analysis challenges, and it is quite unlikely that one can address these challenges by one modeling tool or environment. Our motivation in authoring this report owes itself in part to the well-intentioned meaning of researchers who suggest their tool, or environment, as the tool by which SLAD can meet its SLVA needs in SoSA. Our desire is to provide a rational basis by which, one can consider the recommendations of other researchers, without making what may be a considerable investment of time to study in depth the modeling tools and environments that they have developed. After reviewing the literature on SoS and SoSA—as well as literature more specific to the modeling and analysis—we developed a framework to measure the suitability of a particular modeling tool, or environment, to meet SLAD's need to conduct SLVA of military SoS.

To evaluate our framework, we reviewed 10 divergent modeling tools—or environments—and arrived at the ratings provided in tables 2 and 3. In our choice of these environments to review, our aim was not one of depth, but rather the desire to assess the basic application of our methodology to some candidate environments that might escape scrutiny under more focused study. Our intent was to assess the ability of our approach to provide a reasonable range of results, while at the same time being easily applied to published literature. A secondary benefit of this broad-spectrum approach allows the community-at-large to scrutinize our methodology with openly available data, rather than using obscure environments with data that may not be available, or that may be unfamiliar to the modeling community. We recognize that there is a downside to this methodology; researchers will target their particular tool, or modeling environment, at a class of problem or analytic question that is likely distinctly dissimilar to those one will find in an SoS SLVA. However limiting this downside may be, we reasonably expect to rule out from further consideration those modeling environments with little or no potential to support an SoS SLVA; for our ends this is sufficient and allows us to focus our resources on more profitable environments.

As desired, our methodology produced a reasonable spectrum of results. The relatively midrange similarity scores produced indicated—as expected—that each of the assessed environments were developed for purposes other than modeling a dynamic military SoS. Furthermore, we were not surprised that that the overall maturity scores tended towards the higher ranges, given that each of these modeling environments has been discussed in the open literature and therefore should be expected to be reasonably mature. Our initial review indicated one candidate worthy of further inspection: Lustick's PS-I modeling environment (*29*), which reflects its focus capturing the sociodynamics exemplified by how agents respond to knowledge in political environments. Though Lustick's environment warrants being looked into further, and may provide insight for some applications, we do not expect it to be useful for SoS SLVA without a considerable degree of investment.

We have developed a methodology by which one can quickly screen a modeling environment for its applicability to the modeling and analysis of a military SoS. Noting that researchers tend to develop tools for specialized purposes—not that of an SLVA of a military SoS—we recognize that use of this tool is a filter; those environments that score well, should be looked at more deeply, and those that do not can be removed from consideration. Finally, we applied this methodology to a wide range of environments so that the modeling community may review our approach and provide suggestions for improvement.

6. References

1. U.S. Army *Force Operating Capabilities*; TRADOC Pamphlet 525-66; Headquarters, Training and Doctrine Command: Ft. Monroe, VA, March 7, 2008.
2. Starks, M. W.; Flores, R. *New Foundations for Survivability/Lethality/Vulnerability Analysis (SLVA)*; ARL-TN-216; U.S. Army Research Laboratory: White Sands Missile Range, NM, June 2004.
3. Smith, J. A.; Starks, M. W. *Toward a Survivability/Lethality Analysis Directorate (SLAD) Methodology for Conducting System of Systems Analysis (SoSA)*; ARL-TR-5555; U.S. Army Research Laboratory: White Sands Missile Range, NM, June 2011.
4. U.S. Army Research Laboratory *Workshop on System of Systems Analysis. Sponsored by the Army Research Office*; Las Cruces, NM, March 17, 2009.
5. National Science Foundation *System of Systems Colloquium - Exploring the State-of-the-Art of System of Systems: Theory, Applications, and Vital Issues. Sponsored by the National Science Foundation, Washington DC*, Chicago, IL, July 16, 2007.
6. Boardman, J.; Sauser, B. *Systems Thinking: Coping With 21st Century Problems;* Taylor & Francis: Boca Raton, 2008.
7. Hitchins, D. K. *Systems Engineering: A 21st Century Systems Methodology;* John Wiley: Chichester, West Sussex, England, 2007.
8. Maier, M. W. Architecting Principles for Systems-of-Systems. In the *Systems Engineering: Practices and Tools: Proceedings of the Sixth Annual International Symposium of the International Council on Systems Engineering*, vol. 1; International Council on Systems Engineering: Boston, MA, 1996, pp. 567–574.
9. Maier, M. W. Architecting Principles for Systems-of-Systems. *Sys. Eng.* **1998**, *1* (4), 267–284.
10. Alberts, D. S.; Garstka, J.; Stein, F. P. *Network Centric Warfare: Developing and Leveraging Information Superiority;* 2nd (revised) ed.; National Defense University Press: Washington, DC, 1999.
11. Dahmann, J. S. Systems Engineering for Department of Defense Systems of Systems. In *System of Systems Engineering: Innovations for the 21st Century*, Jamshidi, M., ed.; Wiley: New York, 2008, pp. 218–231.
12. U.S. Department of Defense. Defense Acquisition Guidebook. *https://dag.dau.mil/Pages/Default.aspx*, 2011 (accessed September 6, 2011).

13. Smith, J. A.; Harikumar, J.; Ruth, B. G. Another Perspective on System of Systems: A Step Towards an Analytical Framework. In the *Proceedings of the 4th Annual IEEE International Systems Conference.(SysCon 2010).*, IEEE: 2010, pp. 394–399.

14. Smith, J. A.; Harikumar, J.; Ruth, B. G. *An Army-Centric System of Systems Analysis (SoSA) Definition*; ARL-TR-5446; U.S. Army Research Laboratory: White Sands Missile Range, NM, February 2011.

15. U.S. Army *Mission Command: Command and Control of Army Forces*; Field Manual 6–0; Headquarters, Department of the Army: August 11, 2003.

16. DePuy, W. E. Concepts of Operation: The Heart of Command, the Tool of Doctrine. *Army* **1988**, *38* (8), 26–40.

17. Prosser, B. D. *Nested Concepts: Implementing Commander's Vision and Securing Unity of Effort*; Monograph U.S. Army Command and General Staff College, School of Advanced Military Studies: Ft. Leavenworth, KS, May 22, 1997.

18. Ackoff, R. L.; Emery, F. E. *On Purposeful Systems: An Interdisciplinary Analysis of Individual and Social Behavior As a System of Purposeful Events;* Aldine Transaction: New Brunswick, NJ, 2006.

19. Cherns, A. The Principles of Sociotechnical Design. *Hum. Relat.* **1976**, *29* (8), 783–792.

20. Cherns, A. Principles of Sociotechnical Design Revisited. *Hum. Relat.* **1987**, *40* (3), 153–162.

21. Bar-Yam, Y. Multiscale Variety in Complex Systems. *Complexity* **2004**, *9* (4), 37–45.

22. Hugos, M. H. *Essentials of Supply Chain Management;* John Wiley & Sons: Hoboken, NJ, 2003.

23. del Rosario, R. D.; Tom, K. F.; Mitchell, G.; Ruth, B. G.; Grazaitis, P. J.; Ferryman, T.; Scharett, D. E.; Toomey, C. J., Jr. *Underlying Technologies for Military Logistics Prediction and Preemption Capability*; ARL-SR-0166; U.S. Army Research Laboratory: Adelphi, MD, February 2009.

24. Semboloni, F.; Assfalg, J.; Armeni, S.; Gianassi, R.; Marsoni, F. CityDev, An Interactive Multi-Agents Urban Model on the Web. *Computers, Environment and Urban Systems* **2004**, *28* (1-2), 45–64.

25. Semboloni, F. The Management of Urban Complexity Through a Multi-Agent Participatory Simulation. *DisP* **2007**, *170*, 57–70.

26. Semboloni, F. The CityDev Project: An Interactive Multi-Agent Urban Model on the Web. In *Complex Artificial Environments: Simulation, Cognition and VR in the Study and Planning of Cities*, Portugali, J., ed., Springer-Verlag: Berlin, 2006, pp. 155–163.

27. Semboloni, F. The Multi-Agent Simulation of the Economic and Spatial Dynamics of a Poli-Nucleated Urban Area. In *The Dynamics of Complex Urban Systems: An Interdisciplinary Approach*, Albeverio, S.; Andrey, D.; Giordano, P.; Vancheri, A., eds., Springer-Verlag: Berlin, 2008, pp. 409–427.

28. Josephson, J. R.; Chandrasekaran, B.; Carroll, M.; Iyer, N.; Wasacz, B.; Rizzoni, G.; Li, Q.; Erb, D. A. An Architecture for Exploring Large Design Spaces. In the *Proceedings of the Fifteenth National Conference on Artificial Intelligence and the Tenth Annual Conference on Innovative Applications of Artificial Intelligence*, American Association for Artificial Intelligence: Menlo Park, CA, 1998, pp. 143–150.

29. Lustick, I. PS-I: A User-Friendly Agent-Based Modeling Platform for Testing Theories of Political Identity and Political Stability. *JASSS* **2002**, *5* (3), 7.

30. Fu, D.; Houlette, R.; Bascara, O. An Authoring Toolkit for Simulation Entities. In the *Proceedings of the 2001 Interservice/Industry Training, Simulation & Education Conference (I/ITSEC 2001)*, 2001.

31. Bramson, A. L. Measures of Tipping Points, Robustness, and Path Dependence. In the *Proceedings of Swarmfest 2009*, 2009.

32. Bramson, A. L. Formal Measures of Dynamical Properties: Tipping Points. In *Complex Adaptive Systems and the Threshold Effect: Views from the Natural and Social Sciences: Papers from the AAAI Fall Symposium (FS-09-03)*, Hadzikadic, M. and Carmichael, T., eds., AAAI Press: Menlo Park, CA, 2009, pp. 2–11.

33. Bramson, A. L. Formal Measures of Dynamical Properties: Robustness and Sustainability. In *Complex Adaptive Systems—Resilience, Robustness, and Evolvability: Papers from the AAAI Fall Symposium (FS-10-03)*, Hadzikadic, M. and Carmichael, T., eds., AAAI Press: Menlo Park, CA, 2010, pp. 16–25.

34. Cioppa, T. M.; Lucas, T. W.; Sanchez, S. M. Military Applications of Agent-Based Simulations. In the *Proceedings of the 2004 Winter Simulation Conference*, Ingalls, R. G.; Rossetti, M. D.; Smith, J. S.; Peters, B. A., eds., 2004, pp. 171–180.

35. Body, H.; Marston, C. The Peace Support Operations Model: Origins, Development, Philosophy and Use. *Journal of Defense Modeling and Simulation* **2011**, *8* (2), 69–77.

36. Ilachinski, A. An Artifical-Life Approach to War. In the *Proceedings of the Smithsonian Lecture Series*, 2000.

37. Ilachinski, A. *Artificial War: Mulitagent-Based Simulation of Combat;* World Scientific Publishing Co. Pte. Ltd.: Singapore, 2004.

38. Ilachinski, A. *Enhanced ISAAC Neural Simulation Toolkit (EINSTein): Beta-Test User's Guide*; CIM 610.10; Center for Naval Analyses: Alexandria, VA, November 1999.

39. Ilachinski, A. Exploring Self-Organized Emergence in an Agent-Based Synthetic Warfare Lab. *Kybernetes* **2003**, *32* (1/2), 38–76.

40. Ilachinski, A. Towards a Science of Experimental Complexity: An Artificial-Life Approach to Modeling Warfare. In the *Proceedings of the 5'Th Experimental Chaso Conference*, 2000.

41. Lauren, M. K. A Metamodel for Describing the Outcomes of the MANA Cellular Automaton Combat Model Based on Lauren's Attrition Equation. *http://arxiv.org/abs/nlin/0610036v1*, October 16, 2006 (accessed August 12, 2011).

42. Lauren, M. K. A New Framework for Combat Risk Assessment Using the MANA Model. In the *Proceedings of the 38th Annual Operational Research Society of New Zealand Conference*, 2001.

43. Lauren, M. K.; Stephen, R. T. Fractals and Combat Modeling: Using MANA to Explore the Role of Entropy in Complexity Science. *Fractals-Complex Geom. Patterns Scaling Nat. Soc.* **2002**, *10* (4), 481–489.

44. Lauren, M. K.; Stephen, R. T. Map-Aware Non-Uniform Automata (MANA) - A New Zealand Approach to Scenario Modelling. *Journal of Battlefield Technology* **2002**, *5* (1).

45. Moffat, J.; Smith, J.; Witty, S. Emergent Behaviour - Theory and Experimentation Using the MANA Model. In the *Proceedings of the 9th International Command and Control Research and Technology Symposium (ICCRTS 2004)*, U.S. Department of Defense, Command and Control Research Program: Alexandria, VA, 2004.

46. Moffat, J.; Smith, J.; Witty, S. Emergent Behaviour: Theory and Experimentation Using the MANA Model. *Journal of Applied Mathematics and Decision Sciences* 2006, *2006* (Article ID 54856), 13 pages.

47. Zhang, M. Z.; Hu, X. F.; Si, G. Y.; Zhao, Z. L. Research on Modeling Method of SoS Combat Simulation Based on Agent. *Journal of System Simulation* **2005**, *17* (11), pp 2785–92.

48. OneSAF Public Site. *http://www.onesaf.net/community/*, April 20, 2011(accessed July 11, 2011).

49. Kaste, R.; O'May, J.; Heilman, E. From Simulation to Insights: Experiments in the Use of a Multi-Criterial Viewer to Develop Understanding of the COA Space. In the *Proceedings of the U.S. Army Research Laboratories Collaborative Technology Alliances Symposium*, 2003.

50. Chandrasekaran, B.; Josephson, J. R.; O'May, J.; Heilman, E.; Kaste, R. C. Mining Simulation Data for Insights About a Decision Space: Application to an Urban Combat COA. In the *Proceedings of the Society of Photo-Optical Instrumentation Engineers*, vol. 5423; Trevisani, D. A.; Sisti, A. F., eds.; The International Society for Optical Engineering: Bellingham, WA, 2004, pp. 32–42.

51. Karr, C. R. Conceptual Modeling in OneSAF Objective System (OOS). In the *Proceedings of the 2005 Interservice/Industry Training, Simulation & Education Conference (I/ITSEC 2005)*, 2005.

52. Wittman, R. L., Jr.; Surdu, J. OneSAF Objective System: Toolkit Supporting User and Developer Lifecycles Within a Multi-Domain Modeling and Simulation Environment. In the *Proceedings of the Tenth International Conference on Simulation Technology and Training (SimTecT '05)*, 2005.

53. Josephson, J. R.; Chandrasekaran, B.; Carroll, M.; Iyer, N. S. Multi-Criterial Decision Making System and Method. U.S. Patent 7437343, October 14, 2008.

54. Lustick, I. S.; Miodownik, D. Everyone I Know Is Doing It: Tipping, Political Cascades, and Individual Zones of Knowledge. In the *Proceedings of the Annual Meeting of the American Political Science Association*, 2004.

55. Eidelson, B. M.; Lustick, I. VIR-POX: An Agent-Based Analysis of Smallpox Preparedness and Response Policy. *JASSS* **2004**, *7* (3), 6.

56. Lustick, I. S.; Miodownik, D.; Eidelson, R. J. Secessionism in Multicultural States: Does Sharing Power Prevent or Encourage It? *Amer. Polit. Sci. Rev.* **2004**, *98* (2), pp 209–229.

57. Eidelson, R. J.; Lustick, I. S. National Identity Repertoires, Territory, and Globalization. In *Europe without borders: Re-Mapping Territory, Citizenship and Identity in a Transnational Age*, Berezin, M.; Schain, M., eds., Johns Hopkins University Press: 2004, pp. 89–117.

58. Lustick, I. Secession of the Center: A Virtual Probe of the Prospects for Punjabi Secessionism in Pakistan and the Secession of Punjabistan. *JASSS* **2011**, *14* (1), 7.

59. PS-I Agent-Based Simulation Toolkit. *http://www.polisci.upenn.edu/ps-i/index.html*, 2011 (accessed July 12, 2011).

60. Fu, D.; Houlette, R.; Jensen, R. A Visual Environment for Rapid Behavior Definition. In the *Proceedings of the 12th Conference on Behavior Representation in Modeling and Simulation*, 2003.

61. Fu, D.; Houlette, R.; Warcon: A Wargame Construction Toolset for Military Simulations. In the *Proceedings of the Society of Photo-Optical Instrumentation Engineers*, vol. 5423; Trevisani, D. A.; Sisti, A. F., eds., The International Society for Optical Engineering: Bellingham, WA, 2004, pp 179–189.

62. Bitinas, E. J.; Henscheid, Z. A.; Truong, L. V. Pythagoras: A New Agent-Based Simulation System. *Technology Review Journal* **2003**, *11* (1), pp 45–58.

63. *Pythagoras User's Manual,* version 2.0; Northrup Grumman: Arlington, VA, 2008.

64. Ferris, T. P. Modeling Methodologies for Representing Urban Cultural Geographies in Stability Operations. Master's Thesis, Naval Postgraduate School, Monterey, CA, June 2008.

65. Seitz, T. Representing Urban Cultural Geography in Stabilization Operations: Analysis of a Social Network Representation in Pythagoras. Master's Thesis, Naval Postgraduate School, Monterey, CA, June 2008.

66. Pythagoras Agent-Based Model. *http://harvest.nps.edu/software.html*, 2011 (accessed July 21, 2011).

67. U.S. Army *Counterinsurgency*; Field Manual 3–24; Headquarters, Department of the Army: Washington, DC, December 15, 2006.

68. U.S. Army *Stability Operations and Support Operations*; Field Manual 3–07; Headquarters, Department of the Army: February 20, 2003.

69. Hanley, N.; Gaffney, H. The Peace Support Operations Model: Modeling Techniques Present and Future. *Journal of Defense Modeling and Simulation* **2011**, *8* (2), pp 79–84.

70. Strong, P. The Peace Support Operations Model: Strategic Interaction Process. *Journal of Defense Modeling and Simulation* **2011**, *8* (2), 85–92.

71. Gaffney, H.; Vincent, A. Modeling Information Operations in a Tactical-Level Stabilization Environment. *Journal of Defense Modeling and Simulation* **2011**, *8* (2), pp 105–112.

72. Warren, G.; Rose, P. Representing Strategic Communication and Influence in Stabilization Modeling. *Journal of Defense Modeling and Simulation* **2011**, *8* (2), pp 93–104.

73. Talbot, O.; Wilde, N. Modeling Security Sector Reform Activities in the Context of Stabilization Operations. *Journal of Defense Modeling and Simulation* **2011**, *8* (2), pp 113–121.

74. Bottom, C.; Lindow, D.; Gunzelman, K.; Larson, A.; Marlin, B.; Shaw, J. Team 10: Peace Support Operations Model. In the *Proceedings of the 18th International Data Farming Workshop (IDFW 18)*, 2009, pp. 34–36.

75. Ilachinski, A. Irreducible Semi-Autonomous Adaptive Combat (ISAAC): An Artificial-Life Approach to Land Combat. In the *Proceedings of the MORS Mini-Symposium on Complexity and Warfare*, 1997.

76. Ilachinski, A. *Irreducible Semi-Autonomous Adaptive Combat (ISAAC): An Artificial-Life Approach to Land Warfare*; Research Memorandum CRM 97-61.10; Center for Naval Analyses: Alexandria, VA, August 1997.

77. Wolfram, S. *A New Kind of Science;* Wolfram Media: Champaign, IL, 2002.

78. Ilachinski, A. ISAAC/EINSTein: An Artificial-Life “Laboratory” for Exploring Self-Organized Emergent Behavior in Land Combat. *http://www.cna.org/isaac*, 2011 (accessed August 8, 2011).

79. Mert, E.; Jilson, E. W. Modeling Conventional Land Combat in a Multi-Agent System Using Generalizations of the Different Combat Entities and Combat Operations. Master of Science Thesis, Naval Postgraduate School, Monterey, CA, September 2001.

80. Lauren, M. K. Describing Rates of Interaction Between Multiple Autonomous Entities: An Example Using Combat Modelling. *http://arxiv.org/abs/nlin/0109024v1*, September 18, 2001 (accessed August 12, 2011).

81. Press, W. H.; Flannery, B. P.; Teukolsky, S. A.; Vetterling, W. T. *Numerical Recipes in FORTRAN: The Art of Scientific Computing;* 2nd ed.; Cambridge University Press: Cambridge England, 1992.

82. Hinkson, D.; Puff, C.; McDonald, M.; McCourt, R.; Oh, K. Team 1: The Logistical Impact of Marine Corps Enhanced Company Operations. In the *Proceedings of the 18th International Data Farming Workshop (IDFW 18)*, 2010; pp 2–3.

83. Lauren, M.; Silwood, N.; Chong, N. E.; Low, S.; McDonald, M.; Rayburg, C.; Yildiz, B.; Pickl, S.; Sanchez, R. Team 1: Maritime Force Protection Study Using MANA and Automatic Co-Evolution (ACE). In the *Proceedings of the 18th International Data Farming Workshop (IDFW 18)*, 2009, pp 2–6.

84. Umit, A.; Murat, D.; Yao, M. T. Using the MANA Agent-Based Simulation Tool to Evaluate and Compare the Effectiveness of Ground-Based and Airborne Communications Jammers in Countering the IED Threat to Ground Convoys. In the *Proceedings of the 2007 Spring Simulation Multiconference*, vol. 2, Ades, M. J., ed., Society for Computer Simulation International: Norfolk, VA, 2007, pp 113–118.

85. SEED Center for Data Farming. *http://harvest.nps.edu/*, 2011 (accessed August 18, 2011).

86. Holland, J. H. *Hidden Order: How Adaptation Builds Complexity;* Perseus Books: Reading, MA, 1996.

87. Holland, J. H. *Emergence: From Chaos to Order;* Perseus Books: Reading, MA, 1999.

88. Bratman, M. *Intention, Plans, and Practical Reason;* Harvard University Press: Cambridge, MA, 1987.

89. Cao, Z.; Ma, Y. Research on Composability of Operation Modeling. *Journal of System Simulation* **2007**, *19* (7), pp 1421–23.

INTENTIONALLY LEFT BLANK

Appendix A. The Economics-Based City Development Multi-Agent Simulation (CityDev)

The *City Development Multi-Agent Simulation* (CityDev) model was developed by Semboloni to study the dynamics of simulated urban development within a decentralized System of Systems (SoS) consisting of developers, industrial firms, commercial firms, service providers (public and private sector), and households of consumers/laborers (*24–27*). The simulator of the CityDev model runs on a three-dimensional (3-D) spatial pattern organized in 3-D cells (figure A-1), and is based on an interactive supply-and-demand macro-economy populated by agents, goods, and markets (figure A-2). Each agent within an economic SoS (i.e., a network of households, industrial firms, commercial firms, service firms, and developers) produces goods and services (i.e., labor, constructed buildings, and consumer goods) by using other goods and services, and then exchanges the goods/services in the various urban markets. Because each agent needs a building to live or work in, the urban fabric of a developing cityscape is both initially produced and dynamically transformed, as a function of agent-based economic interactions. CityDev, here applied to a developing simulation of the city of Florence, Italy, allows human users to actively participate in the evolving simulation through the Internet. In this context, CityDev experiments demonstrate multi-agent participatory simulation—one or more urban developers building a new quarter in a suburb of Florence.

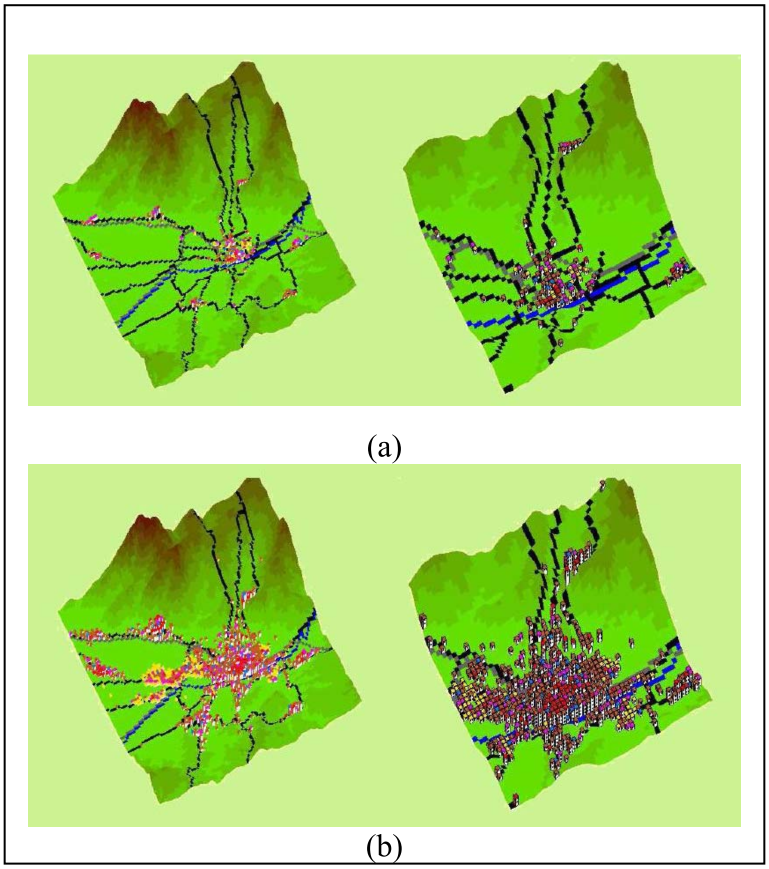

Figure A-1. Screenshots from a CityDev simulation.
Note: City Dev simulation illustrating the evolving development of a notional urban cityscape (i.e., Florence, Italy) at two different simulation time steps (where images on the right are magnifications of those on the left): (a) time step = 10; (b) time step = 300. Here, each cell within the simulation represents a landscape surface area of 200 m × 200 m.

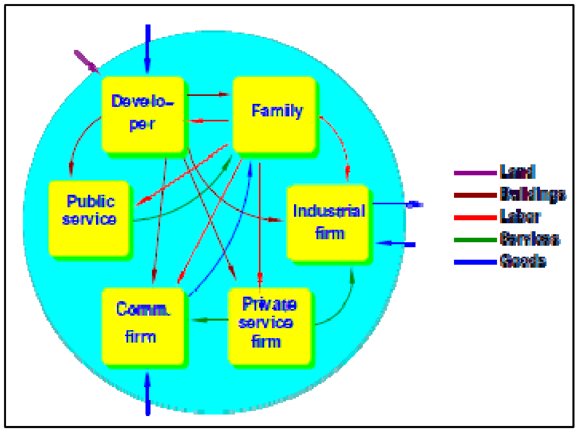

Figure A-2. Diagram of the macro-economic ontology driving the dynamics of CityDev.
Note: An outside demand for an industrial product, or for commercial and service activities (usual in a big city), stimulates the production in exogenously related business sectors, such as industry. Households are then generated whose members provide labor for industrial firms. Because households demand final consumer goods, commercial firms are subsequently generated, which in turn employ laborers and so the process continues. Each agent requires a building to live or work in; this in turn generates developers who build the evolving urban fabric.

The results of applying CityDev to an aggregate of different agent types within an economic (vice military) context essentially serves to demonstrate an SoS dynamically functioning at a single coarse-grained timescale as a sociotechnical system, which utilizes an attenuated form of nested concepts and inter-related purposes. Here we replace commander intent and military concept of operations with the consensus vision of an urban planning committee of purposeful human agents. Regarding the evaluation of simulation results, CityDev demonstrates basic time-series and limited state-space analysis functionality (figures A-3 [a] and A-3 [b], respectively), but little else in the way of desired SoSA capability. Accordingly, table A-1 reports our estimates of the associated (a) SoS and (b) SoSA ontology attribute similarity-scoring values for CityDev, while table A-2 reports SoS and SoSA constituent component developmental maturation scoring values associated with the CityDev software. Given that rights to the CityDev software appear to currently belong exclusively to the University of Florence, Italy (although potential users can freely access the CityDev simulation via the Internet to participate in urban planning experiments), we have assigned an accessibility status of *red* to this agent-based simulation software. Finally, figure A-4 depicts the quadrant chart states of the SoS and associated SoSA designs associated with CityDev (i.e., [1.68, 3.00, *red*] and [1.69, 3.00, *red*], respectively).

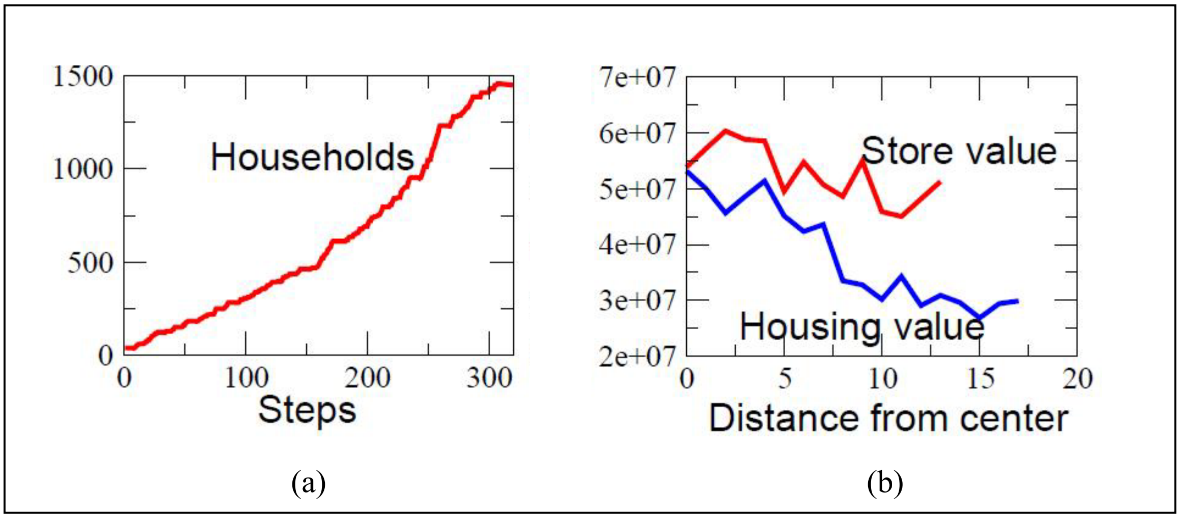

Figure A-3. Analysis of CityDev simulation results.
Results: (a) growth of the household cell population as a function of simulation time step (where each household cell represents 400 human inhabitants); (b) housing value (in notional monetary units) as a function of household cell distance from the city center.

Table A-1. SoS and SoSA ontology attribute similarity scores associated with the CityDev agent-based simulation.
Note: see section 3 for a discussion of the criteria, attributes, and weights.

(c) System of systems				
	Attribute			
	Description	Number	Weight	Score
Nested Concepts and Inter-Related Purposes →	Commander Intent	1	10	1
	Concept of Operations	2	10	1
	Purposeful Systems	3	7	3
Dynamic and Evolving Processes →	Synchronous Behaviors	4	10	1
	Multiple Time Scales	5	7	1
	Non-Stationary and Non-Ergodic Processes	6	5	3
	Sociotechnical Systems Theory	7	7	3
(d) System of systems analysis				
Analysis of Nested Concepts and Inter-Related Purposes →	Commander Intent	1	7	1
	Concept of Operations	2	7	1
	Synthetic Approach	3	10	2
Analysis of Dynamic and Evolving Processes →	State-Space Analysis	4	5	2
	Time-Series Analysis	5	10	3
	Frequency-Domain Analysis	6	5	1
Sociotechnical Systems Analysis →	Adaptive Multiscale Techniques	7	7	1

Table A-2. SoS and SoSA software component maturation scores associated with the CityDev agent-based simulation.

	Component			
	Description	Number	Weight	Score
(a) System of systems	Simulation Engine Software	1	10	3
	GUI Software	2	10	3
(b) System of systems analysis	Analysis Software	1	10	3

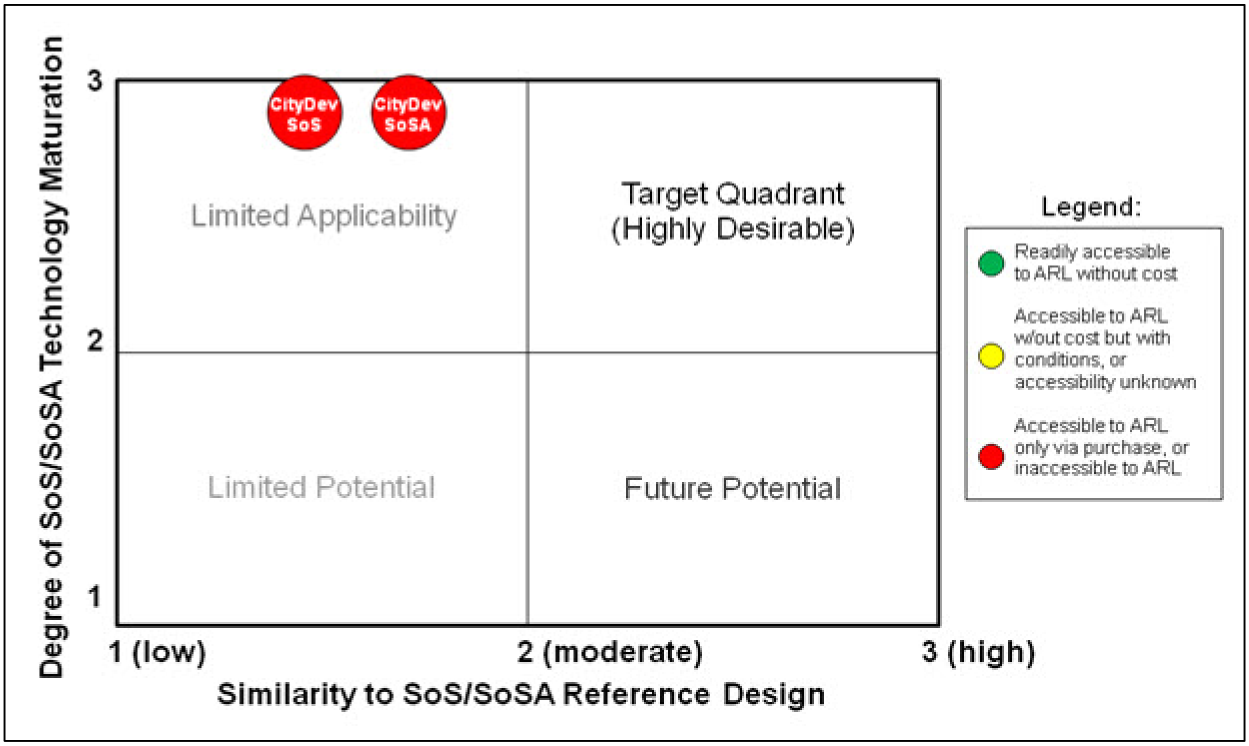

Figure A-4. Quadrant chart addressing SoS and associated SoSA designs for the CityDev agent-based simulation.

INTENTIONALLY LEFT BLANK.

Appendix B. The Seeker-Filter-Viewer Architecture (S-F-V)

As part of the U.S. Army Research Laboratory (ARL) Advanced Decision Architectures Collaborative Technology Alliance (ADA CTA), Chandrasekaran and Josephson of Ohio State University's Laboratory for Artificial Intelligence Research (LAIR) developed a multicriterion decision technology called the *Seeker-Filter-Viewer* architecture (S-F-V) (*28*). Among the objectives for developing this software in the ADA CTA was to evaluate its utility for a variety of Army decision-making tasks. As such, S-F-V supports the military mission planner in generating a number of alternative plans, evaluating the plans along a number of performance dimensions of interest, filtering the plan to obtain the Pareto-Optimal subset, and visually examining the Pareto-Optimal set in tradeoff diagrams to narrow the choices further.

Specifically, S-F-V has three synergistic components:

1. The *seeker* supervises the generation and evaluation of many different course of action (COA) alternatives along several different user-supplied situational/contextual criteria.
2. The *filter* utilizes Pareto-Optimization techniques to find the subset of alternatives that are optimal over a range of criteria (i.e., filtered survivors comprise the Pareto subset wherein no COA alternative dominates another).
3. The *viewer* utilizes a visual exploration environment with active cross-linked charts to analyze the relationships between various dimensions of the filtered COA alternatives.

The S-F-V architecture abstraction is depicted in. When linked with a wargame simulation, the viewer component can also be used to explore correlations within the simulation data along different criteria, COA specifications, and intermediate simulation events.

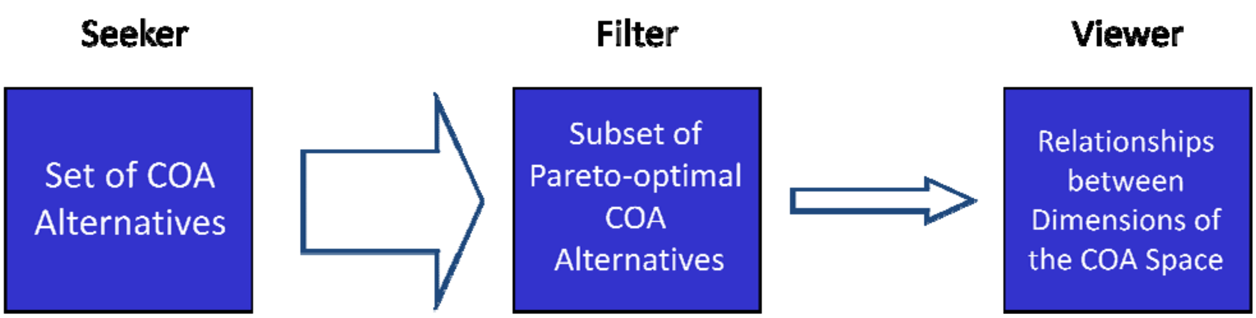

Figure B-1. The S-F-V architecture abstraction.
Note: The seeker provides a set of decision alternatives, evaluated along multiple criteria; the filter outputs the Pareto-Optimal subset, and the viewer supports visually examining the relationship between various dimensions of the COA alternatives.

S-F-V has been utilized by ARL researchers, in collaboration with LAIR personnel, to support the development of battle simulation technologies and techniques for understanding a scenario-specific military decision space via simulation data mining. For these studies, the One Semi-Automated Forces (OneSAF) Test Bed (OTB) combat simulation (*48*) was used in conjunction with S-F-V by ARL and LAIR to explore a basic Blue-versus-Red killer/victim scenario (*49*) and a Blue-offensive/Red-defensive urban combat scenario (*50*). In the latter study, a Blue company attack on a city sector is carried out in two distinct phases, where phase 1 consists of the mechanized portion of the company's attack via a three-pronged approach to encircle the urban area and drive defending Red forces from positions around the sector (figure B-2). To demonstrate the usefulness of the viewer capability of S-F-V, the analytic query, "Is Blue company operational health at any of the attack directions at the end of phase 1 predictive of the final mission outcomes?" was explored using OneSAF simulation data. The results of this analysis are presented in figure B-3.

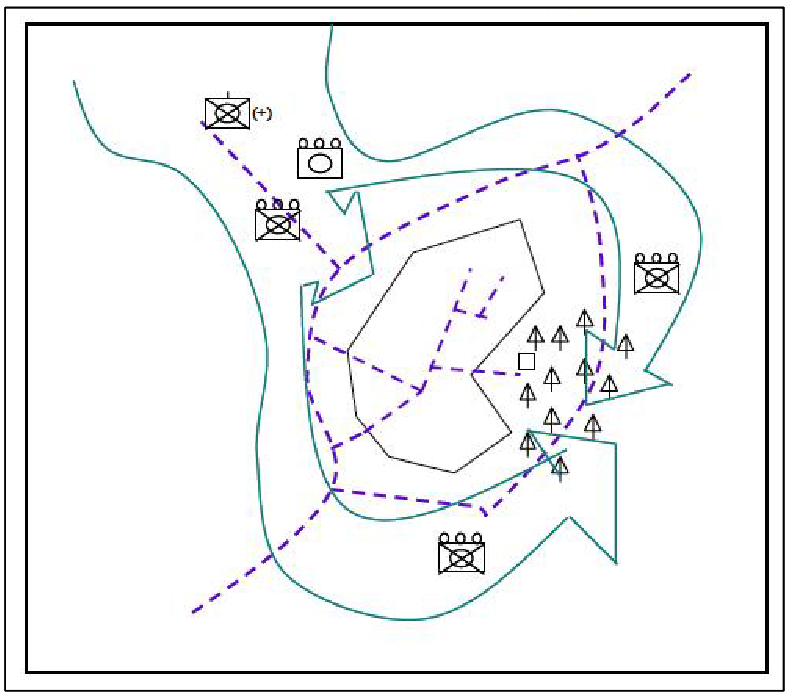

Figure B-2. Diagram of the planned phase 1 assault by the Blue mechanized company within the OneSAF military operations in urban terrain (MOUT) scenario.

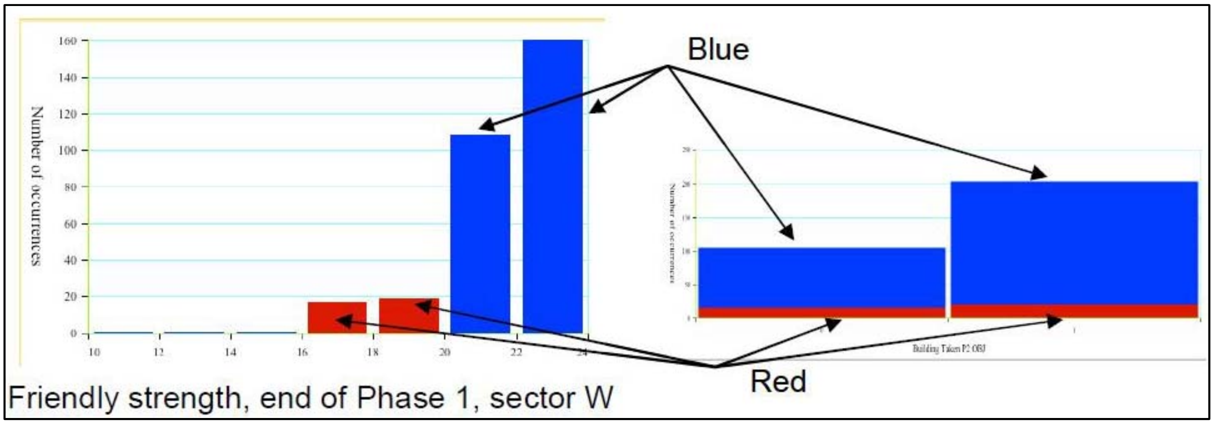

Figure B-3. Relating Blue mechanized company strength at a sector at the end of phase 1 to the odds of Blue taking control of a building serving as Red force-operational headquarters (where the latter is the Blue company mission objective).

Note: In the left-hand-side plot, histograms report on the occurrence frequency of the number of surviving Blue platforms positioned in sector W at the conclusion of phase 1 (where red columns refer to unacceptably low population levels of Blue survivors, while blue columns refer to healthy population levels). In the right-hand-side plot, histograms report on the frequency of Blue company mission accomplishment (right column) vs. mission failure (left column). In both plots, the same simulation occurrence data contributes to a specific color-coded section of a histogram, meaning that relatively low Blue strength at the end of phase 1 in sector W (data contributing to the red columns in both plots) does *not* significantly affect likelihood of successful Blue mechanized company mission accomplishment.

When evaluating S-F-V, we considered the Blue force as potentially modeled in the most recent OneSAF software release—the OneSAF Objective System (OOS) (*51, 52*)—as representing the SoS in this context. Given that OOS supports scripted XML-formatted entity behaviors (utilizing a common set of behavior primitives), this reflects a military SoS demonstrating the following;

- limited nested concepts in terms of commander intent and concept of operations but *not* purposeful adaptive behavior,
- dynamical processes potentially unfolding over multiple time scales but with limited evolutionary capability, and
- limited sociotechnical system capability due to a lack of adaptive behavior in simulated entities.

However, in terms of SoSA capabilities, S-F-V (which is really an analytic vice simulation tool) potentially demonstrates a robust ability to analyze nested concepts and inter-related purposes, dynamic and evolving processes, and adaptive multiscale behavior exhibited by a military SoS. Accordingly, table B-1 reports our estimates of the associated SoS and SoSA ontology attribute similarity scoring values for (a) OOS and (b) S-F-V, while table B-2 reports SoS and SoSA constituent-component-developmental maturation scoring values associated with (a) OOS and (b) S-F-V. Although OOS is readily releasable to all Department of Defense organizations, Ohio State University holds the patent rights to S-F-V (currently licensed to Aetion Technologies LLC) (*53*); thus, we have assigned this SoS and SoSA software accessibility status values of *green* and *red*, respectively. Finally, figure B-4 depicts the quadrant chart states of the SoS and associated SoSA designs associated with OOS and S-F-V (i.e., [2.32, 2.67, *green*] and [2.80, 3.00, *red*], respectively).

Table B-1. SoS and SoSA ontology attribute similarity scores associated with the OOS and S-F-V, respectively. Note: see section 3 for a discussion of the criteria, attributes and weights.

(a) System of systems				
	Attribute			
	Description	Number	Weight	Score
Nested Concepts and Inter-Related Purposes →	Commander Intent	1	10	3
	Concept of Operations	2	10	3
	Purposeful Systems	3	7	1
Dynamic and Evolving Processes →	Synchronous Behaviors	4	10	3
	Multiple Time Scales	5	7	3
	Nonstationary and Nonergodic Processes	6	5	1
	Sociotechnical Systems Theory	7	7	1
(b) System of systems analysis				
Analysis of Nested Concepts and Inter-Related Purposes →	Commander Intent	1	7	3
	Concept of Operations	2	7	3
	Synthetic Approach	3	10	3
Analysis of Dynamic and Evolving Processes →	State-Space Analysis	4	5	3
	Time-Series Analysis	5	10	2
	Frequency-Domain Analysis	6	5	3
Sociotechnical Systems Analysis →	Adaptive Multiscale Techniques	7	7	3

Table B-2. SoS and SoSA software component maturation scores associated with OOS and S-F-V.

	Component			
	Description	Number	Weight	Score
(a) System of systems	OOS Tools Layer	1	10	3
	OOS Model Layer	2	10	2
	OOS Services Layer	3	10	3
(b) System of systems analysis	S-F-V Seeker Software	1	10	3
	S-F-V Filter Software	2	10	3
	S-F-V Viewer Software	3	10	3

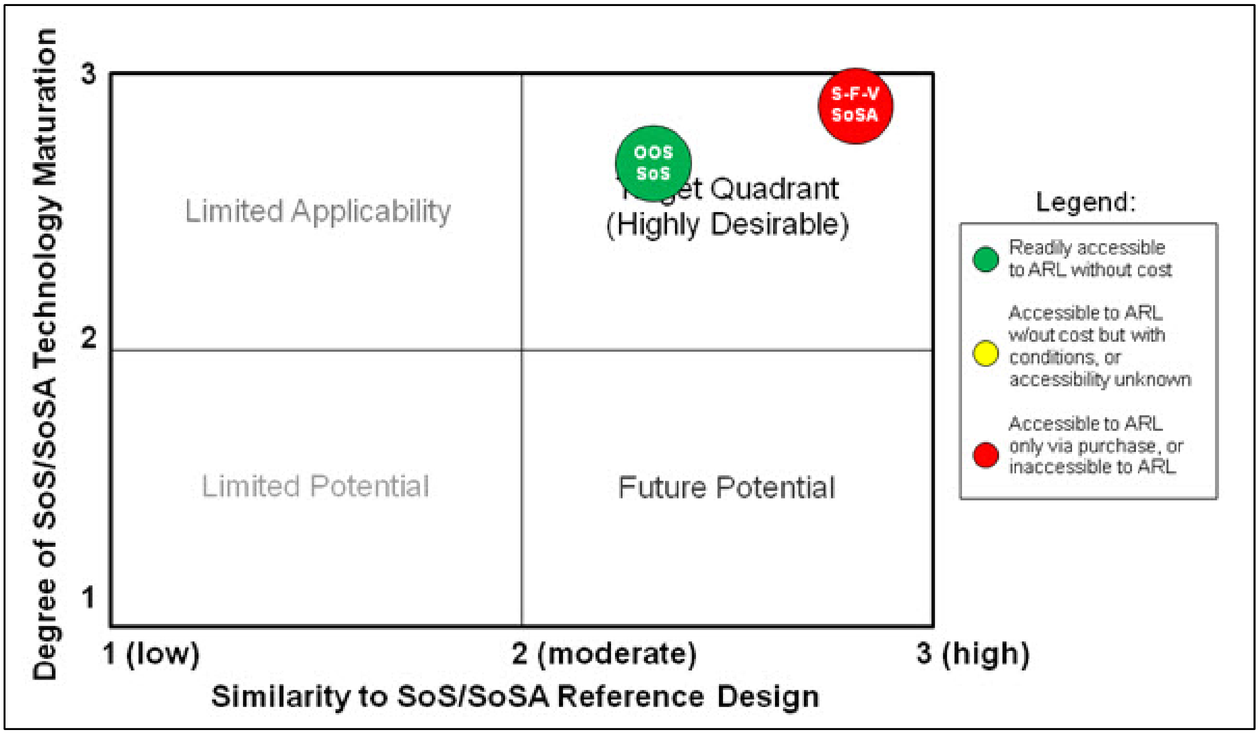

Figure B-4. Quadrant chart addressing SoS and associated SoSA designs for the synergistic OOS/S-F-V software combination.

INTENTIONALLY LEFT BLANK.

Appendix C. The Political Science-Identity (PS-I) Agent-Based Simulation Toolkit

The *Political Science-Identity* (PS-I) agent-based computer simulation platform was originally developed by Lustick and various colleagues at the University of Pennsylvania in order to operationally simulate, refine, and test competing versions of political constructivist identity theory (*29*). Based on an earlier prototype, the Agent-Based Identity Repertoire (ABIR) model, purpose-oriented agents with repertoires of "identities" (i.e., an agent's publicly demonstrated political orientation observable by other agents within a simulation), interact in geographic localities of specifiable size, and are influenced as well by identity values attached to other specific cross-landscape neighboring agents. These identity values (e.g., agent membership in a specific political party; agent agreement/alliance with a specific opinion regarding a political issue) are assumed to be dynamic, thereby simulating conditions in which, individuals may express latent identities—or learn to use new identities—because of local social pressures toward conformity and/or overall shifts in the relative attractiveness of different political identities. Large batches of controlled virtual histories (i.e., simulated social system state-/phase-space trajectories) are used for comparative and statistical analysis. PS-I was specifically designed to promote systematic correspondence between simulated agent political decision making processes (DMPs) and corroborated theoretical positions in political science and psychology. The continuing development of PS-I has been motivated by the desire of social scientists using constructivist theories of identity to analyze and understand patterns of mobilization, attachment, and conflict arising from cultural, ethnic, religious, or other traits (*54*-*57*).

Figure C-1 displays the various model and agent parameter configuration editing options available to the user in the PS-I graphical user interface (GUI). These include the following.

- The *model specification editor* is used to (i) establish or change simulation landscape size (where a "landscape" is a two-dimensional [2-D] array of cells with one agent—or none—occupying each cell);[*] (ii) define the set of identities available to an agent (where only one identity can be "activated," or publicly demonstrated by an agent at a time); and (iii) specify the attributes of different agent classes within a scenario (e.g., number of different potential identities associated with a particular agent class, and the probability that an agent's activated identity can change as a function of time or social pressure from neighboring agents).
- The *model parameter editor* is used to define various scenario-specific parameter settings.

[*]It must be noted that, within the context of PS-I, an "agent" can be scaled to represent anything from a single human individual to a collection of spatially-clustered individuals all sharing the same "identity" or subjective opinion/orientation regarding a political issue.

These include:

- *Bias volatility* is the rough probability within a simulation update that any particular identity within an agent will be eligible for a change in the bias assigned to it.
- *Average tension of a landscape (ATI)* is the total tension in a simulation landscape divided by the number of active agents (where "tension" refers to the number of encounters an agent demonstrating one specific active identity has in its local surrounding neighborhood with agents demonstrating an opposing active identity).
- *Dominant identity (DI)* is the identity activated by more agents than any other available identity.

- The *agent selection editor* allows the user to implement desired initial condition distributions of agents in a landscape prior to simulation execution. The selection editor may also be used to examine and statistically analyze which kinds of agents with varying expressed identities and attributes exist in what types of patterns in a landscape, at different points of time throughout a simulation.
- The *effect tool* allows the user to implement changes in the attributes of any set of designated agents—at any point in time—during a PS-I simulation run.

In this fashion, PS-I can be configured to simulate a heterogeneous collective of opinionated microscale individuals whose sociopolitical opinions can evolve over time, and thus generate emergent macroscale patterns of political uniformity, and/or diversity.

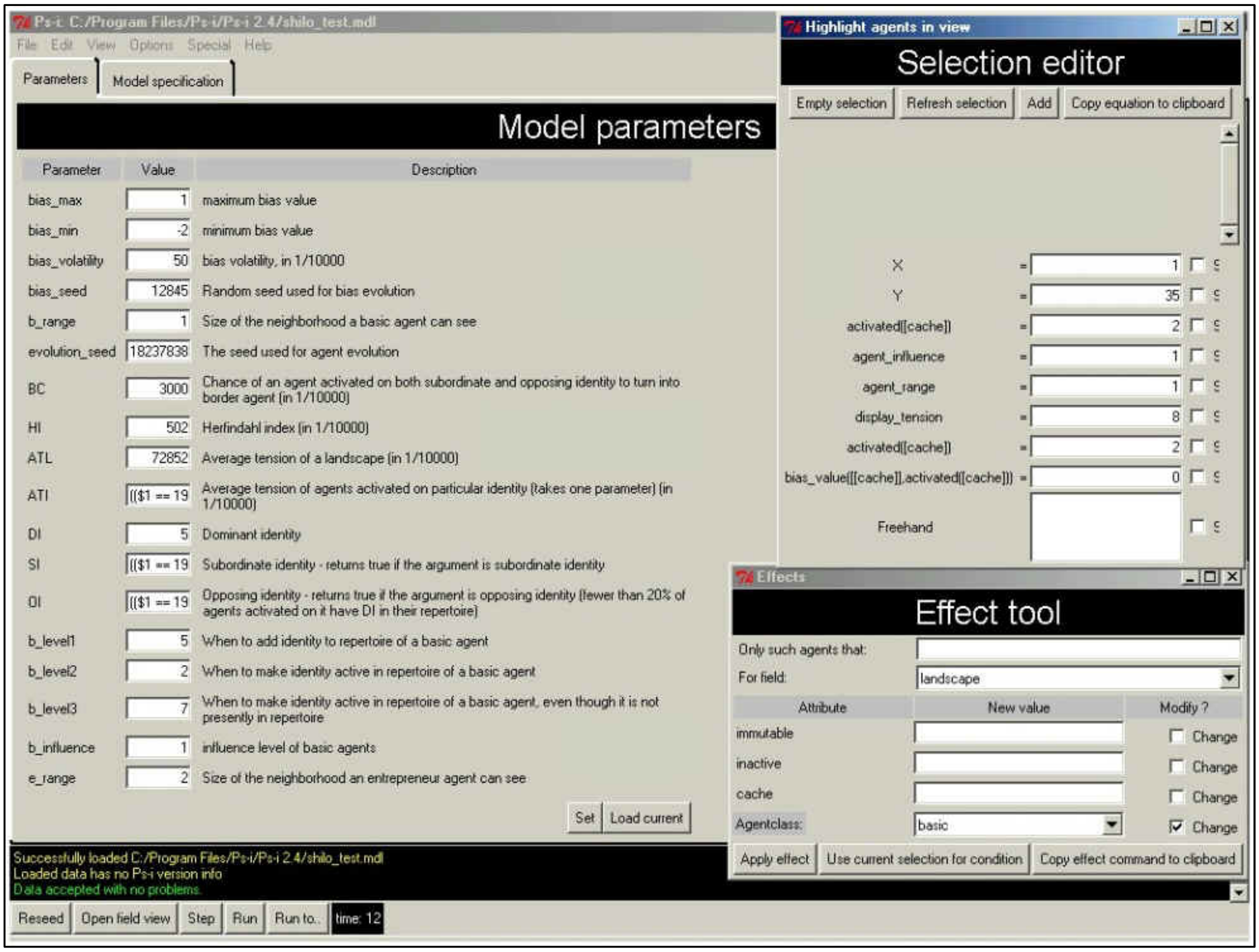

Figure C-1. Screenshot of the PS-I simulation configuration GUI.

Over the past several years, perhaps the most ambitious application of PS-I was Lustick's attempt to create an agent-based political model of Pakistan, within which, clear and widely accepted principles of political competition among bounded rational sociocultural groups and individuals (e.g., Punjabi, Muslim, Pakistani Government, Military, etc.), were implemented via a set of relatively simple DMPs. Starting with a set of reasonable initial conditions (i.e., the best data available for distributions of sociopolitical influence and affiliation among Pakistanis), a "Virtual Pakistan" (VirPaK) was constructed using the PS-I toolbox in order to study circumstances conducive toward a notional future secession of the center geographic region in Pakistan (*58*). Figure C-2 presents the VirPak simulation landscape populated with agents expressing a total of 29 potential identities (graphically represented by cells showing different colors and icons), and representing either membership in various Pakistani sociocultural groups, or alliance with different Pakistani political movements and parties. In figure C-2(a), the initial state of VirPak is shown (i.e., time step = 0), while figure C-2(b) depicts the three simulated VirPak future states (time step = 608, out of a sum total of 100 simulated potential future states), wherein central region Punjabi secession is realized. Finally, for the purposes of analyzing these three "rare event" potential future states, figure C-3 presents a comparison of the state-space trajectories of eleven of the VirPak parameters (i.e., total agent population associated with, and bias weighting demonstrated by, by five different socio/cultural/political identities as a function of time) associated with “Future 24.”

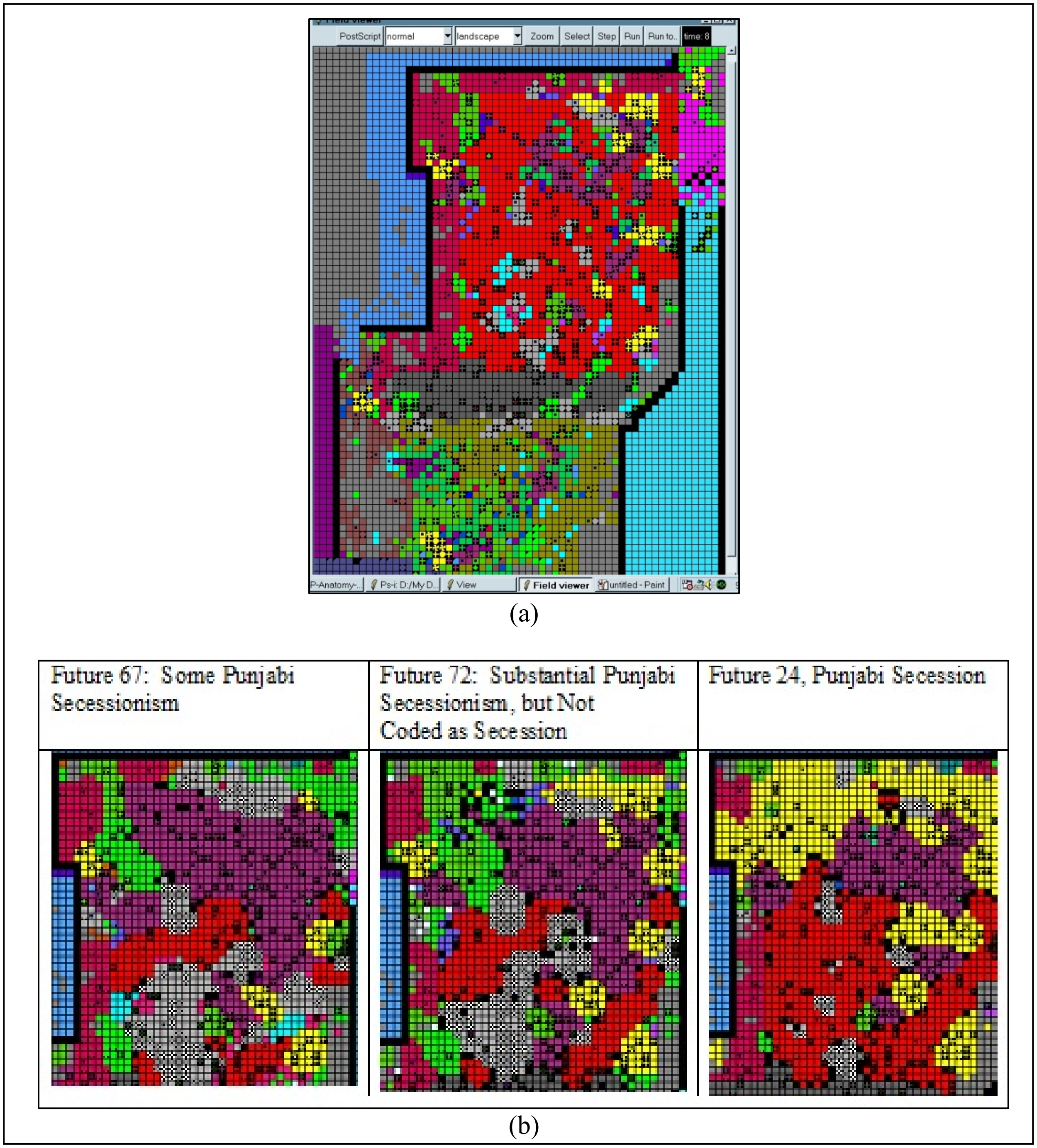

Figure C-2. States of the VirPak simulation: (a) initial state at time step = 0; (b) three examples of simulated Punjabi secession states at time step = 608.

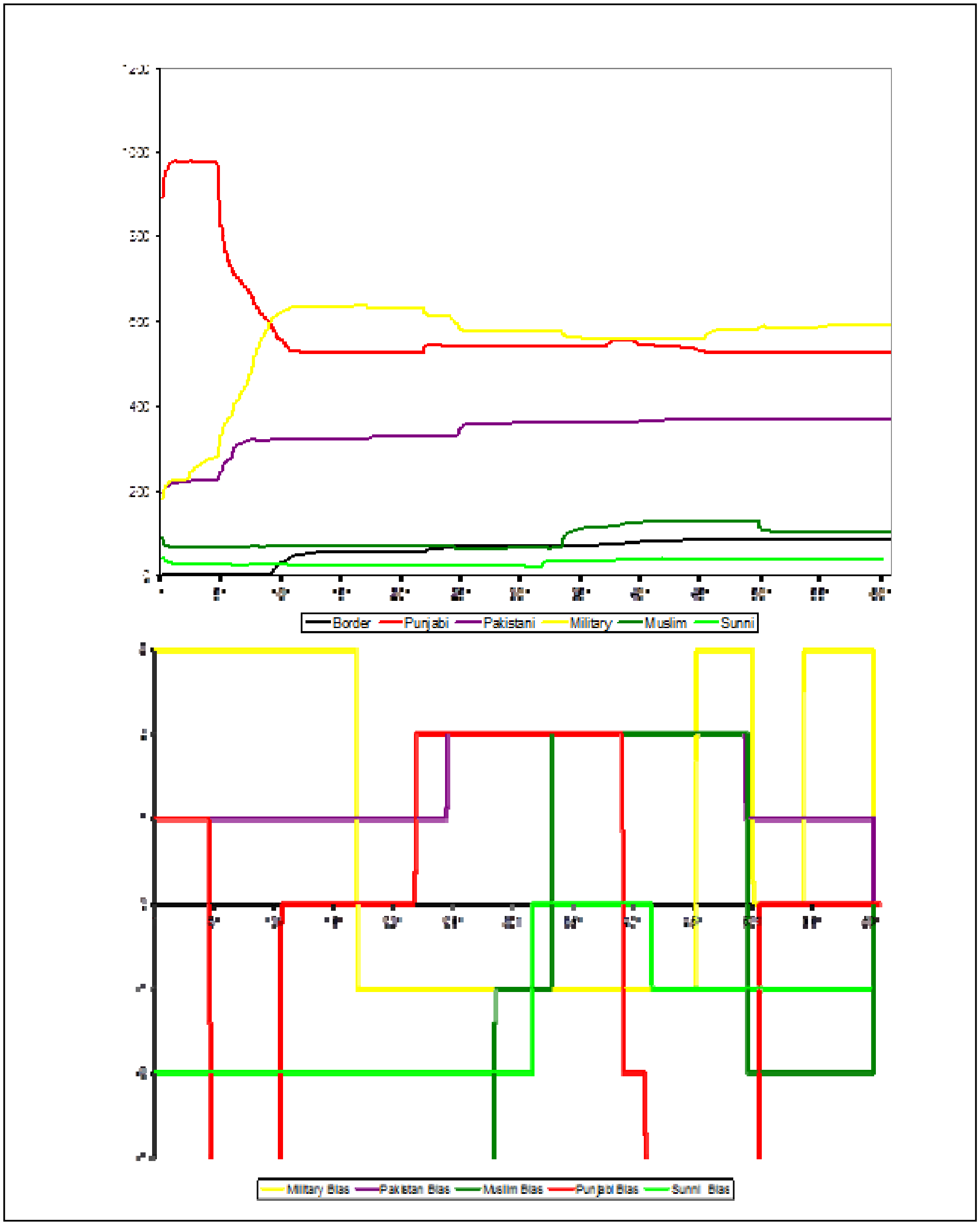

Figure C-3. State-/phase-space trajectories of total agent population associated with six different socio/cultural/political identities and bias weighting demonstrated by five different socio/cultural/political identities, as a function of simulation time associated with VirPak "Future 24."

As was the case with CityDev (see appendix A), the results of applying PS-I to an aggregate of different agent identity types within a socio/cultural/political (vice military) context essentially serves to demonstrate an SoS dynamically functioning at a single coarse-grained timescale; although, increasing the graphical resolution of a PS-I landscape *does* imply an associated finer-grained timescale. At this coarse-grained timescale, we view this SoS as a sociotechnical system, which utilizes a vastly decentralized form of nested concepts and inter-related purposes, where commander intent and military concept of operations are replaced by the emergent political aggregation of groups of purposeful *human* agents. However, from an SoSA perspective, PS-I simulation results can be addressed via application of state-space, time-series and frequency-domain analysis techniques. In addition, the evolving suite of analysis tools in the PS-I toolbox can also (to some degree) address multiscale analysis adaptively via dynamic user interaction, with an evolving simulation instance, using the PS-I effect tool. Accordingly, table C-1 reports our estimates of the associated (a) SoS and (b) SoSA ontology attribute similarity scoring values for PS-I , while table C-2 reports SoS and SoSA constituent component developmental maturation scoring values associated with the still-evolving (via user contributions) PS-I software (tables C-2[a] and C-2[b], respectively). Given that access to the PS-I software is openly available to anyone in the world with Internet access (*59*), we have assigned an accessibility status of *green* to this agent-based simulation software. Finally, figure C-4 depicts the quadrant chart states of the SoS and associated SoSA designs associated with PS-I (i.e., [1.98, 2.82, *green*] and [2.31, 2.60, *green*], respectively).

Table C-1. SoS and SoSA ontology attribute similarity scores associated with the PS-I agent-based simulation. Note: see section 3 for a discussion of the criteria, attributes, and weights.

(a) System of systems				
	Attribute			
	Description	Number	Weight	Score
Nested Concepts and Inter-Related Purposes →	Commander Intent	1	10	1
	Concept of Operations	2	10	1
	Purposeful Systems	3	7	3
Dynamic and Evolving Processes →	Synchronous Behaviors	4	10	2
	Multiple Time Scales	5	7	2
	Nonstationary and Nonergodic Processes	6	5	3
	Sociotechnical Systems Theory	7	7	3
(b) System of systems analysis				
Analysis of Nested Concepts and Inter-Related Purposes →	Commander Intent	1	7	1
	Concept of Operations	2	7	1
	Synthetic Approach	3	10	3
Analysis of Dynamic and Evolving Processes →	State-Space Analysis	4	5	3
	Time-Series Analysis	5	10	3
	Frequency-Domain Analysis	6	5	3
Sociotechnical Systems Analysis →	Adaptive Multiscale Techniques	7	7	2

Table C-2. SoS and SoSA software component maturation scores associated with the PS-I agent-based simulation.

	Component			
	Description	Number	Weight	Score
(a) System of systems	Simulation Engine	1	10	3
	Field Viewer	2	10	3
	Agent Viewer	3	10	3
	Selection Editor	4	10	3
	Effect Tool	5	10	3
	Potential Simulation Add-ons (under development by user community)	6	5	1
(b) System of systems analysis	Selection Editor	1	10	3
	Statistics Toolbox	2	10	3
	Potential Analysis Toolboxes (under development by user community)	3	5	1

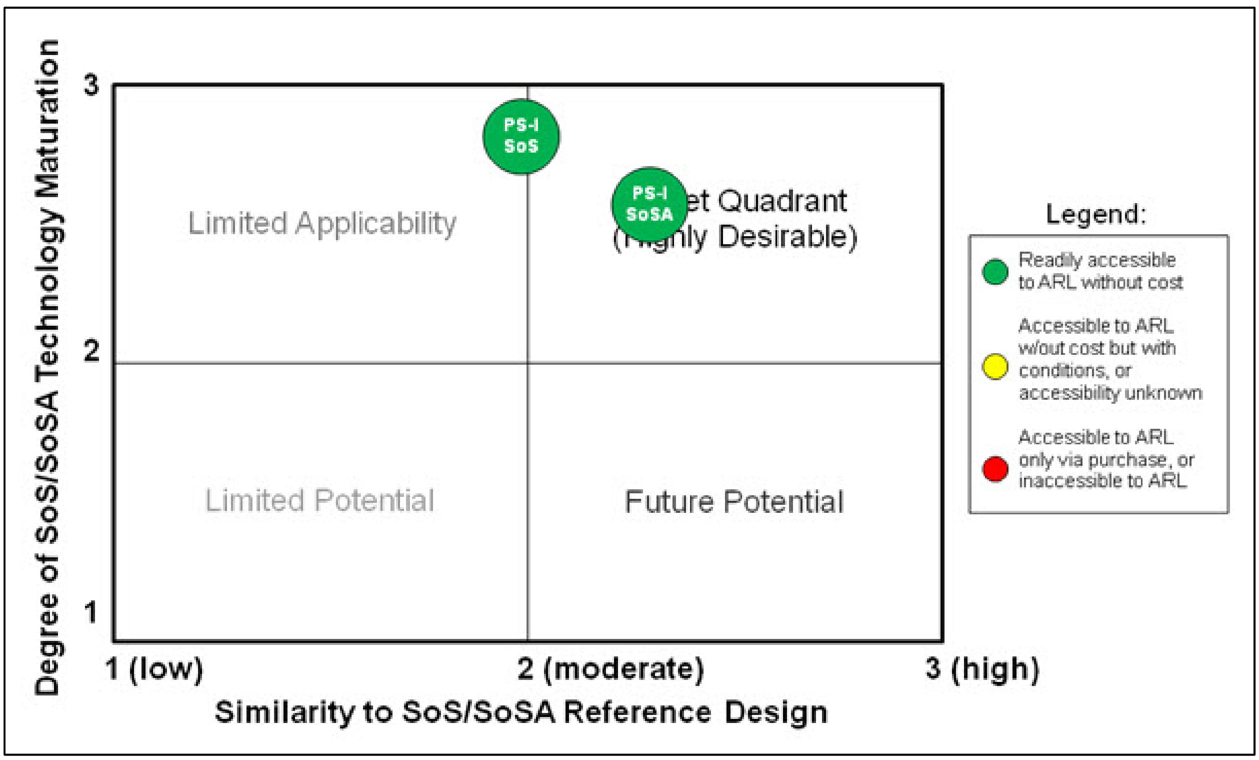

Figure C-4. Quadrant chart addressing SoS and associated SoSA designs for the PS-I agent-based simulation toolbox.

INTENTIONALLY LEFT BLANK.

Appendix D. The Wargame Construction Toolset (Warcon) for Military Simulation

Via a contract vehicle with Stottler Henke Associates, Inc., the U.S. Air Force has been developing the *Wargame Construction Toolset* (Warcon) that would empower Air Force instructors to create small-scale instructional wargames, which embody modern military doctrine and warfighting principles—including Network-Centric Warfare (*30*). The aim of this toolset is to make the wargame simulation authoring process accessible to a wide range of military instructors, via an intuitive visual interface and advanced authoring assistant, which eliminates the need for user programming. The toolset was conceived to ultimately feature a customizable adjudication engine with advanced features for modeling effects-based operations, Psychological Operations (PSYOP), Military Operations Other Than War (MOOTW), and other aspects of modern warfare. It features user-authoring components to facilitate rapid development of simulations, most prominently an adaptive authoring interface and a collaborative authoring assistant.

Fu and Houlette of Stottler Henke Associates, Inc., created a proof-of-concept prototype to investigate the viability of the proposed Warcon design concept. Most of their prototyping addressed the Warcon adaptive user interface, which resulted in three types of graphical user interfaces (GUIs) that can collectively provide a simulation construction capability to all potential users, given their varying computer programming skill levels.

- *Warcon Edit* is designed for the novice user who wants to make relatively minor, quick changes to an already constructed wargame (figure D-1). In this mode, the user can rename locations and assets, change victory achievement conditions, tune entity-level DMPs, and generally alter any parameterized element of a scenario.
- *Warcon Build* is targeted at the intermediate user who intends to actually construct new wargames, but does not want to delve too deeply into the underlying mechanics of the wargame simulation engine (figure D-2). This mode features a "building-blocks" approach to simulation creation, where the user can select entities, behaviors, and other components from a library of existing scenarios and DMPs, and then assemble them into a new wargame simulation.
- *Warcon Forge* is the most powerful of the three authoring modes. It enables the expert user to create entirely new types of wargame simulations from scratch, with total control over their constituent entities and DMPs. Warcon Forge will give the author the ability to visually construct new gaming doctrine, new kinds of simulation entities with different attributes, new behaviors for entities, and virtually every other type of wargame component.

As last reported, the Warcon prototype successfully demonstrated the Edit and Build user-interface modes, with the Forge mode still under development.

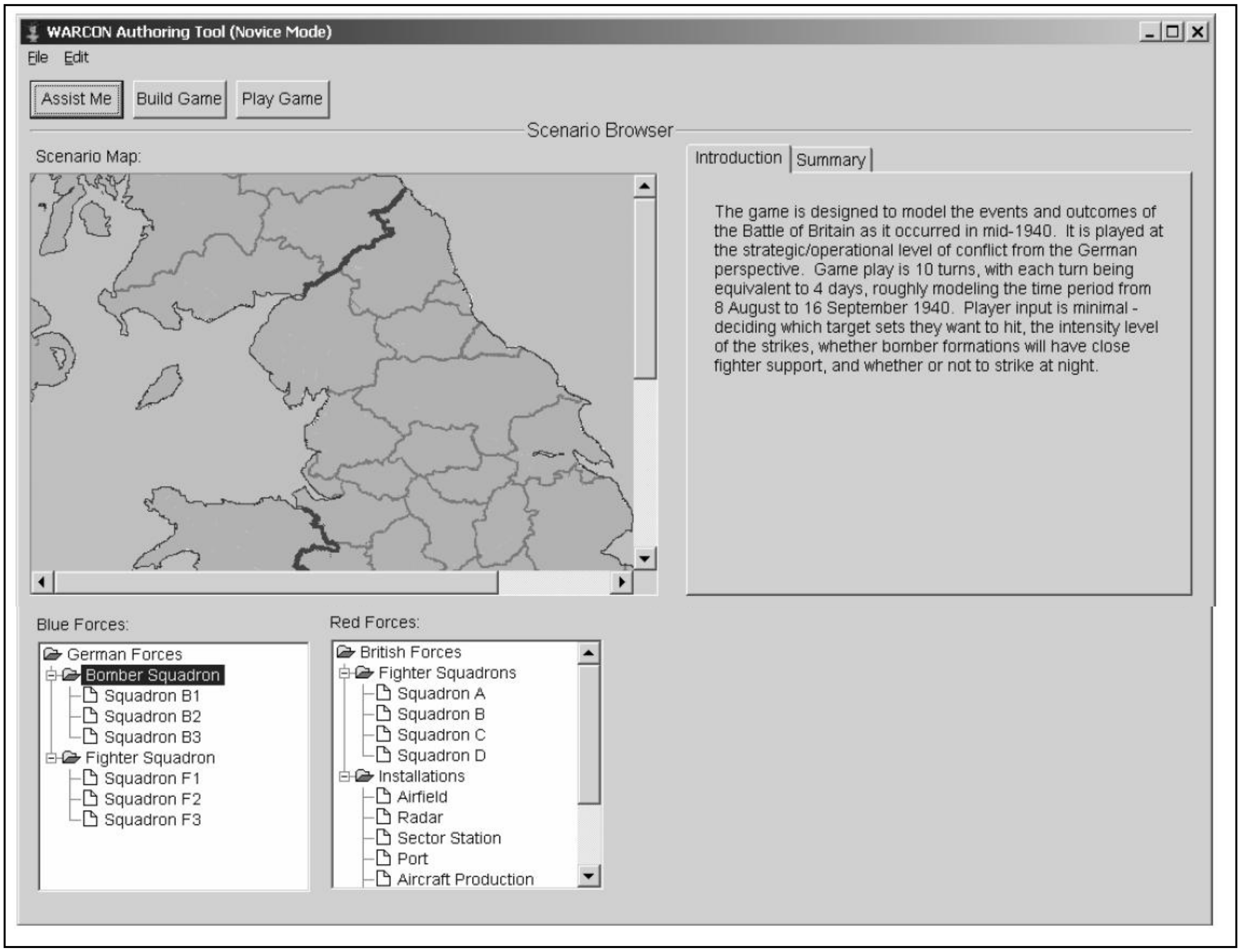

Figure D-1. Screenshot of the Warcon Edit GUI showing Fu and Houlette's adaptation of the World War II Battle of Britain from an existing U.S. Air Force Air University simulation.
Note: Here, the user can change some aspects of a wargame simulation scenario (e.g., force size and asset composition), which can be represented by numerical values. At the top left, there are three buttons: Assist Me, Build Game, and Play Game. The "Assist Me" button invokes a collaboration wizard to help novices modify a simulation, the "Build Game" button invokes a compiler to take all the wargame specifications and output a simulation, and the "Play Game" button runs the modified simulation.

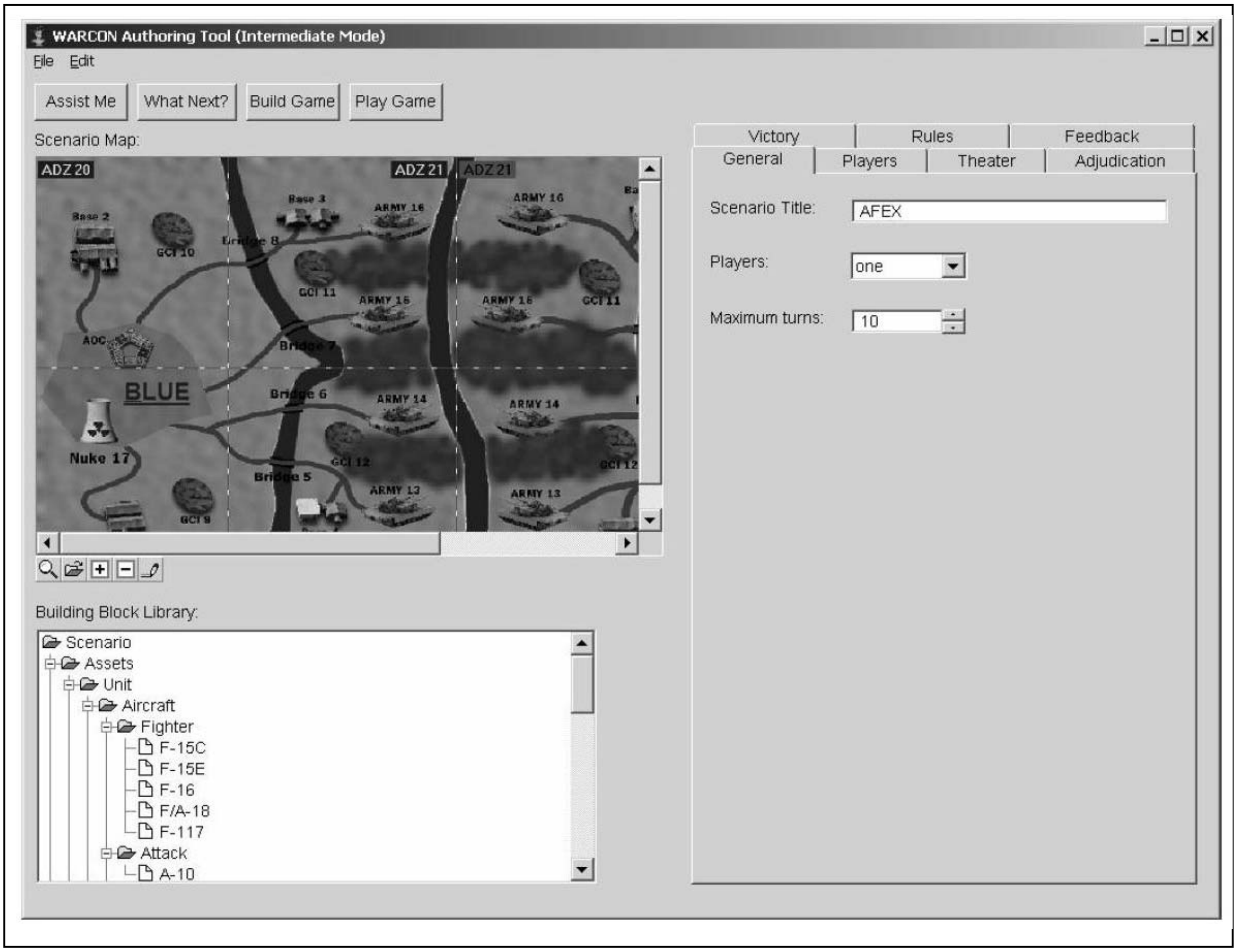

Figure D-2. Screenshot of the Warcon Build GUI showing Fu and Houlette's adaptation of the U.S. Air Force Exercise (AFEX) wargame simulation.
Note: Here, the user can reference already-existing scenario building blocks as shown in the lower left corner of the GUI. The tabs to the right in figure D-2 show various editable aspects of an AFEX scenario, including "rules" (i.e., the DMPs associated with simulated entities experiencing specific situations), combat "adjudication" (e.g., interactive graphical interface displays of kinetic energy weapon probabilities of hit-and-kill given a hit, and expected postinteraction damage assessment), and "victory" (i.e., the wargame's end-state termination conditions under which the simulation will stop).

Given that the Warcon simulation toolset was primarily designed for purposes of educating Air Force military personnel about the complexities of wargaming, it does not appear to demonstrate any immediate capability to facilitate SoS modeling and associated SoSA (although the *potential* capability is implied by the intended scope of eventual Warcon application to simulation of emerging types of warfare and doctrine). As far as simulating military entity behavior is concerned, Warcon utilizes a previously-developed finite state machine (FSM) approach to DMP modeling, similar to that used in current commercial computer games (*60, 61*). Thus, it is likely that Warcon could be (at best) applied to construct a simulation demonstrating a "quasi-" SoS, dynamically operating at a single timescale and utilizing a limited form of nested concepts and inter-related purposes, where commander intent and military concept of operations are clearly represented (given the military design purpose of the toolset), but purposeful behavior by simulated entities is not. As stated previously, from an SoSA perspective, Warcon would appear to demonstrate little capability at best. Thus, table D-1 reports our estimates of the associated SoS and SoSA ontology attribute similarity scoring values for Warcon (table D-1 [a] and table D-2 [b], respectively). Given the uncertainty reflecting the current status of Warcon software development (beyond the demonstration prototype described above), table D-2 reports on evidenced associated SoS and SoSA constituent component developmental maturation scoring values (table D-2 [a] and table D-2 [b], respectively). Given that access to the Warcon software executable was likely intended to be available to all Air Force military instructors—but current developmental state of the toolset is unknown—we have assigned an accessibility status of *yellow* to this item. Finally, figure D-3 depicts the quadrant chart states of the SoS and associated SoSA designs associated with Warcon (i.e., [1.89, 1.75, *yellow*] and [1.00, 1.00, *yellow*], respectively).

Table D-1. SoS and SoSA ontology attribute similarity scores associated with the Warcon simulation tool set. Note: see section 3 for a discussion of the criteria, attributes, and weights.

(a) System of systems				
	Attribute			
	Description	Number	Weight	Score
Nested Concepts and Inter-Related Purposes →	Commander Intent	1	10	3
	Concept of Operations	2	10	3
	Purposeful Systems	3	7	1
Dynamic and Evolving Processes →	Synchronous Behaviors	4	10	2
	Multiple Time Scales	5	7	1
	Nonstationary and Nonergodic Processes	6	5	1
	Sociotechnical Systems Theory	7	7	1
(b) System of systems analysis				
Analysis of Nested Concepts and Inter-Related Purposes →	Commander Intent	1	7	1
	Concept of Operations	2	7	1
	Synthetic Approach	3	10	1
Analysis of Dynamic and Evolving Processes →	State-Space Analysis	4	5	1
	Time-Series Analysis	5	10	1
	Frequency-Domain Analysis	6	5	1
Sociotechnical Systems Analysis →	Adaptive Multiscale Techniques	7	7	1

Table D-2. SoS and SoSA software component maturation scores associated with the Warcon simulation tool set.

	Component			
	Description	Number	Weight	Score
(a) System of systems	Simulation Engine	1	10	1
	Edit GUI	2	10	2
	Build GUI	3	10	2
	Forge GUI	4	10	2
(b) System of systems analysis	Simulation Analysis Software	1	10	1

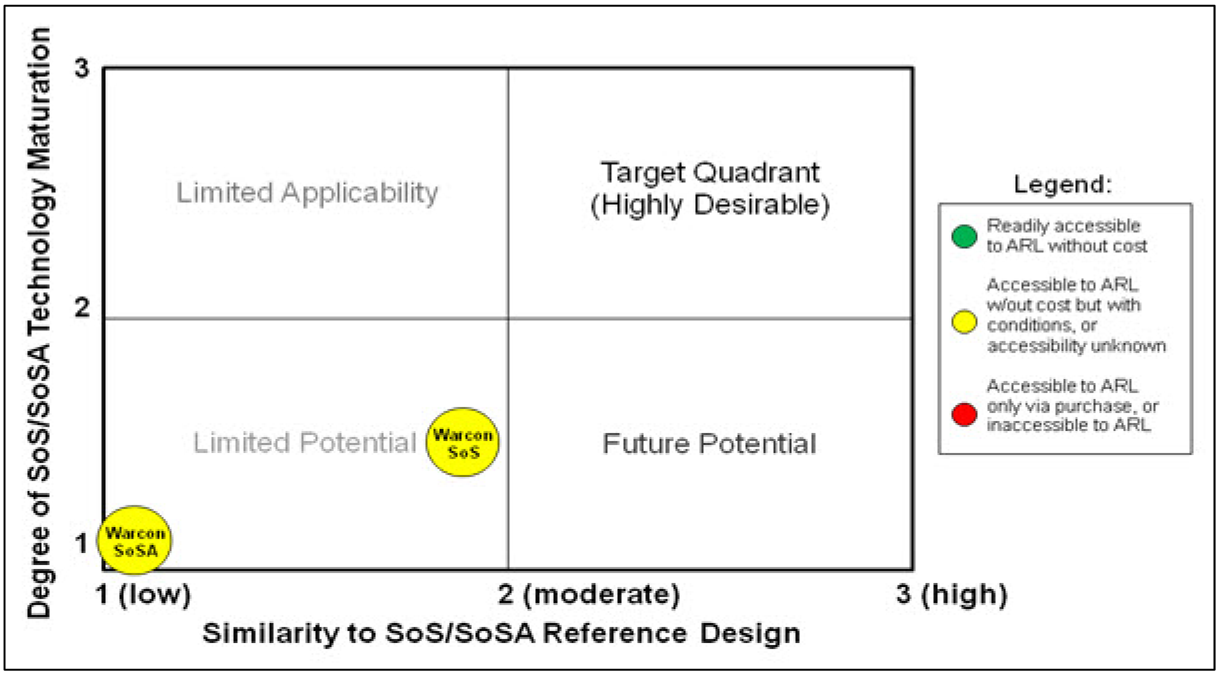

Figure D-3. Quadrant chart addressing SoS and associated SoSA designs for the Warcon simulation toolset.

INTENTIONALLY LEFT BLANK.

Appendix E. Measures of Tipping Points, Robustness, and Path Dependence (MTPRPD) Methodology

The analysis of nonstationary complex systems needs to be able to capture the dynamical processes inherent within the system, rather than just a sequence of static "snapshots." Existing statistical techniques are ill-suited to measure properties of system dynamics. The *Measures of Tipping Points, Robustness, and Path Dependence* (MTPRPD) methodology is a novel approach created by Bramson (University of Michigan) to address the analysis of complex system dynamics in terms of "tipping points," system robustness, and system state-space path dependence (*31-33*). For each of these concepts, a formal definition is provided, which utilizes Markov model representations of system states and related dynamical behavior in the context of a stylized system state transition diagram.

The MTPRPD system analysis methodology can be summarized by the following concept definitions.

- A *Markov model* is comprised of a set of measurable system states and the probabilistically weighted transitions amongst those states, where a *state* S_i is a complete specification of the N measurable attributes of the i^{th} specific configuration of the system:

  $$S_i = \{X_1(i), X_2(i), X_3(i), \dots, X_N(i)\} \qquad (1)$$

  where, $X_N(i)$ is the state of the N^{th} system attribute within the i^{th} system configuration.* Empirical data from an experimental trial, or one simulation instance of a model of the system dynamics, can be mapped into a time-ordered sequence of states, which the system progressively occupies over an interval of time. Combining the time-ordered system state sequences over a set of experimental trials, or simulation instances, produces a state-transition diagram, which captures all measured system dynamical trajectories through an associated system state-space (figure E-1). By using the frequencies of measured interstate transitions, the analyst can construct the desired Markov model representation of system dynamics.

- A *path* within a system's state space is an ordered collection of states and transitions within a Markov model, such that there exists a positive probability that governs a transition from a given state to a successor state within the collection. A *cycle* is a path that starts and ends with the same state. These concepts are illustrated in figure E-2.

---

*Continuous-valued attributes of a system can be mapped into discrete-valued system state attributes via the use of value intervals and/or functional categories. For example, a materiel system capability may be actively present (represented by an attribute value of 1), or absent (an attribute value of 0) as a function of system constituent component function/dysfunction.

- The *support* of a system state is the set of states that have a path to it. A system state that always transitions to itself is called an *equilibrium state*. An *orbit* is a set of states such that if the system enters that set it will always revisit every member of the set and the system can never leave that set. An *attractor* (denoted A_i), is either an equilibrium state or an orbit of the system. Those states from which the system will eventually move into a specific attractor are said to be in that attractor's *basin of attraction*. Each of these concepts is illustrated in figure E-3.

- Using Markov models and the concepts defined above as a starting point, Bramson then goes on to define and describe several terms applicable to the analysis of complex dynamical systems. These include the following.

  - A *tipping point* is a type of system state that (for whatever reason) behavior is different before and after some transition. Identifying tipping points (as a property of system dynamics) should not depend on whether humans are in control of system behavior, or what kind of process is driving the dynamics.

  - The *robustness* of a set of system states is the average cumulative long-term probability density measured over the states in the set given, so that the system may start out in *any* realizable state within the associated Markov model. Starting with this concept, one can then define associated state-based definitions for the terms sustainable, resilient, recoverable, stable, and static.

  - *Path dependence and sensitivity* of time-evolving system states is not a single, well understood, or properly defined concept. For example, different features of system dynamics can have the path sensitivity property: process outcomes can be path sensitive, processes can be path sensitive, measures can be path sensitive, and state-space paths themselves, can be path sensitive. As a specific example, *trajectory forcing* through a system's state space occurs when a particular systemic transition forces the evolving system dynamics down through a specific sequence of time-ordered states (figure E-4).

Although originally designed by Bramson for application in the areas of sociocultural and sociotechnical systems analysis, the MTPRPD methodology is general enough for application to the analysis of any type of complex dynamical system.

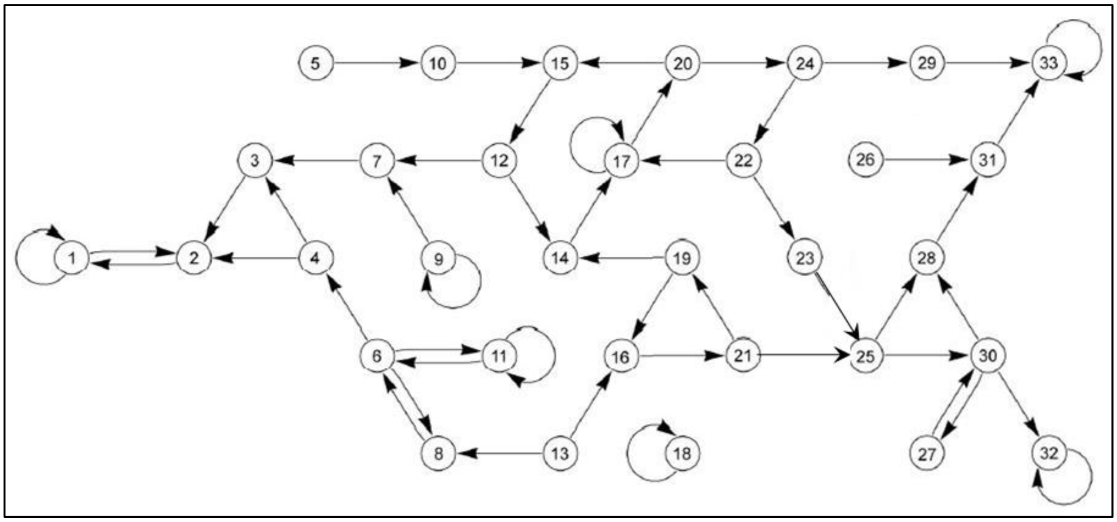

Figure E-1. State transition diagram describing all system dynamical trajectories through an associated space of 33 discrete system states as measured via either, (i) experimental trials, or (ii) simulation instance histories, as generated by a system model.

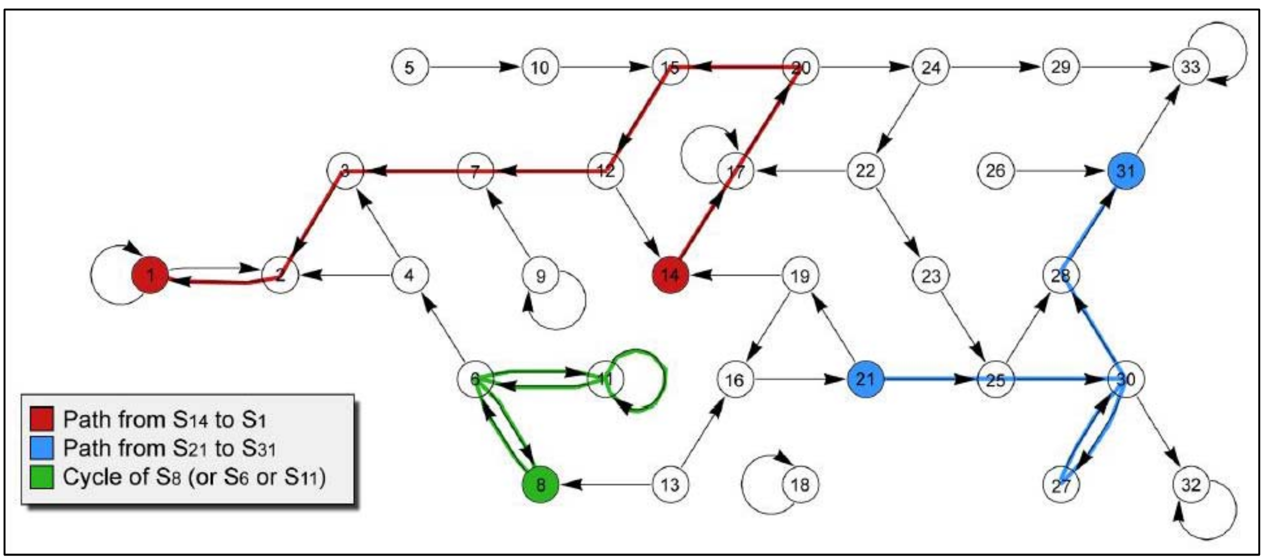

Figure E-2. Paths and cycle associated with the 33-state transition diagram depicted in figure E-1.

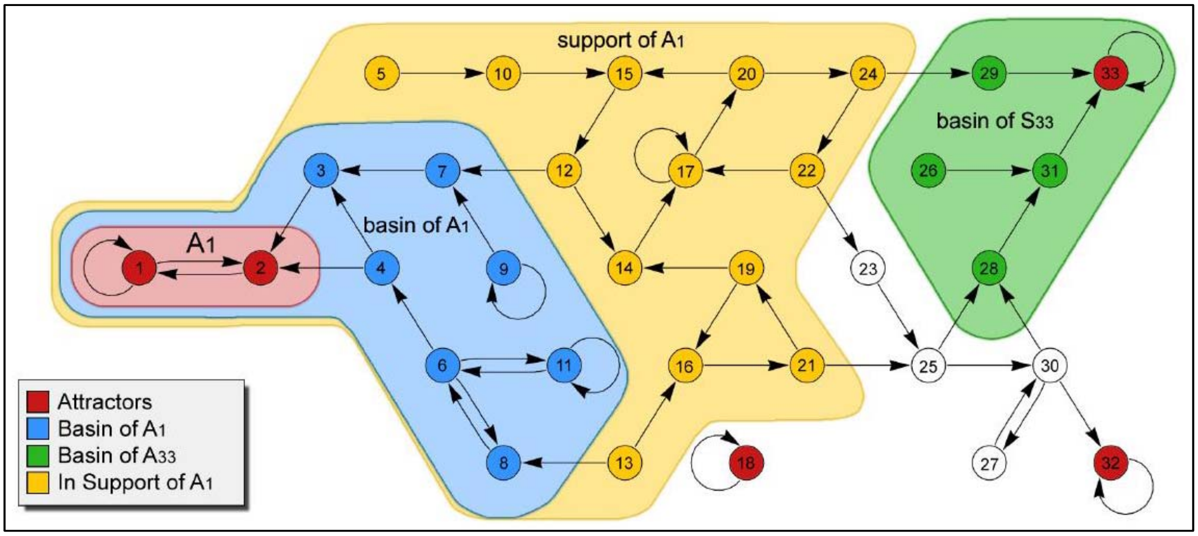

Figure E-3. Attractors, basins of attraction, and support structure associated with the 33-state transition diagram depicted in figure E-1.
Note: attractor A_1 is a two-state orbit, while attractors A_{18}, A_{32}, and A_{33} are all continuously-recurring equilibrium states.

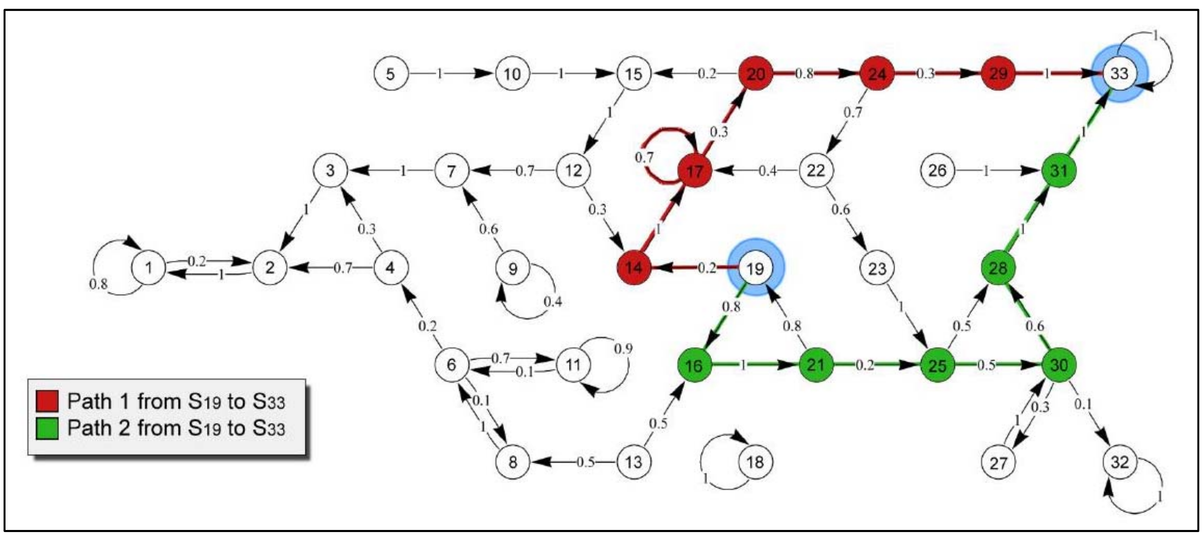

Figure E-4. Trajectory forcing of two distinct system state-space paths from S_{19}–S_{33} (encircled in light blue) associated with the 33-state transition diagram depicted in figure E-1.
Note: Here, the numerical values inserted in the center of the interstate (and occasional intrastate) arrows in the figure represent the state-to-state transition probabilities associated with measured system dynamical behavior. In this context, the total "force" associated with an exact path from S_i to S_j is defined as the product of the probabilities of all the state-to-state transitions required to stay that specific course. In the case of the two paths depicted in the figure, the forces associated with the red and green state-space trajectories are 0.010 and 0.048, respectively.

Given that the MTPRPD methodology is purely a generalized dynamical system analytic abstraction, rather than a purported model of any particular system or causal apparatus, there is no associated SoS model. However, we believe the methodology does demonstrate a strong potential for application to certain aspects of SoSA.

Bramson addresses his specific purpose in the design of the MTPRPD methodology accordingly:

> It is meant to be completely abstract and general and therefore capable of measuring (the described) system properties in any system. Because it does not model any generating process it cannot address the "why" or "how" questions. It is not meant to. This (methodology) answers the "whether" and "how much" questions … A general methodology provides a framework through which all modelers (and some data analysts) can determine whether and how much of each of these properties of system dynamics obtains … and compare results across models regardless of the generating mechanisms (*31*).

Thus, the MTPRPD methodology clearly demonstrates ability to directly analyze dynamic and evolving processes in terms of both state-space analysis and time-series analysis (and, by implication, frequency-domain analysis) techniques. In addition, given that this analysis methodology was intentionally designed to be generalized for application to the analysis of any type of dynamical system (and, again by implication, any type of SoS), it should prove to be adaptable to any scale demonstrated by system (or SoS) time-history data being analyzed, and therefore is applicable to sociotechnical systems analysis. However, the methodology's purposeful focus upon answering questions of "whether" and "how much" (vice "why" and "how") tends to severely limit its applicability to the analysis of nested concepts and inter-related purposes. Given these considerations, table E-1 reports our estimates of the associated SoSA ontology attribute similarity scoring values for the MTPRPD methodology. Also, given that Bramson has yet to release any evolving analysis software associated with his methodology (although he has reported that it is to be "under development"), table E-2 reports SoSA constituent component developmental maturation scoring values associated with the methodology. In addition, given that Bramson ultimately intends to freely and openly distribute the MTPRPD software (*31*), we have assigned this SoSA software the accessibility status of green. Finally, figure E-5 displays the quadrant chart state of the SoSA design associated with the MTPRPD methodology (i.e., a value of [2.06, 1.00, green]).

Table E-1. SoSA ontology attribute similarity scores associated with the MTPRPD methodology.
Note: see section 3 for a discussion of the criteria, attributes, and weights.

System of systems analysis				
	Attribute			
	Description	Number	Weight	Score
Analysis of Nested Concepts and Inter-Related Purposes →	Commander Intent	1	7	1
	Concept of Operations	2	7	1
	Synthetic Approach	3	10	1
Analysis of Dynamic and Evolving Processes →	State-Space Analysis	4	5	3
	Time-Series Analysis	5	10	3
	Frequency-Domain Analysis	6	5	3
Sociotechnical Systems Analysis →	Adaptive Multiscale Techniques	7	7	3

Table E-2. SoSA software component maturation scores associated with the MTPRPD methodology.

	Component			
	Description	Number	Weight	Score
System of systems analysis	Markov Data Model Construction Software	1	10	1
	State-Space Analysis Software	2	10	1

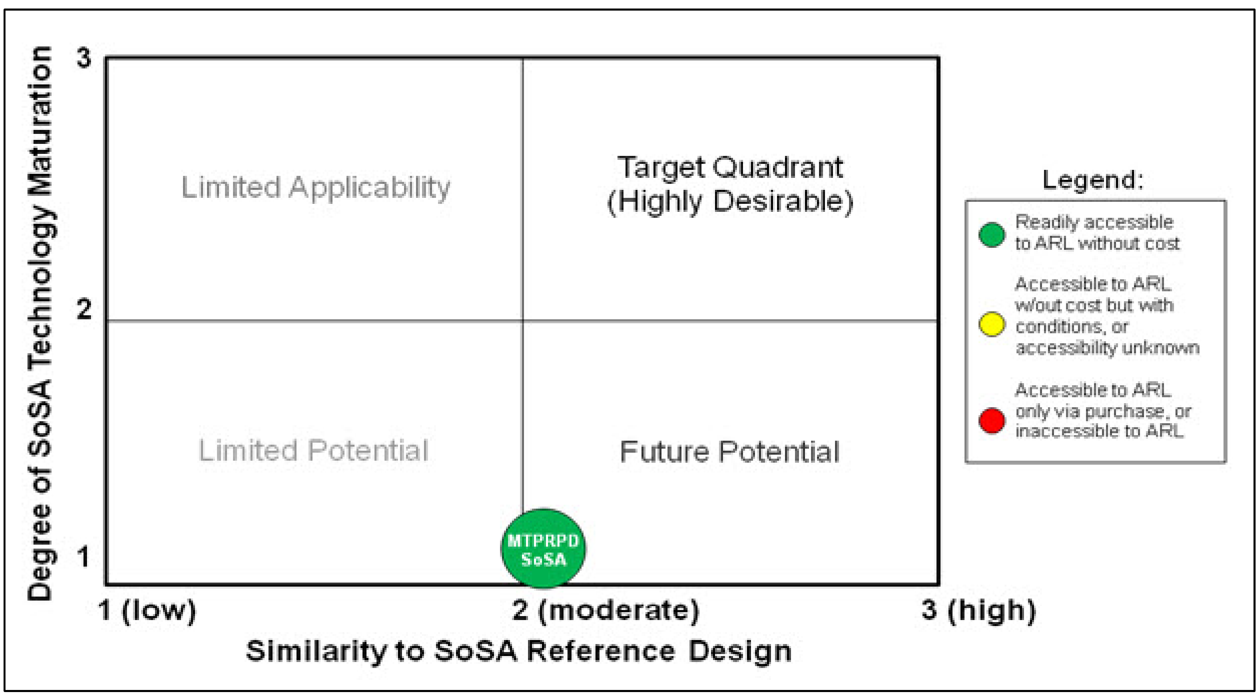

Figure E-5. Quadrant chart addressing SoSA design for the MTPRPD methodology.

Appendix F. The Pythagoras Agent-Based Simulation System

In 1997, the U.S. Congress authorized an applied research project to evaluate nontraditional combat simulation techniques with the potential to quantitatively address three areas largely overlooked in traditional combat modeling: nonlinear effects, incorporation of intangibles (such as human factors and leadership capability), and co-evolving DMPs between adversaries. As a result, the U.S. Marine Corps Combat Development Command (MCCDC) conceived of and initiated Project Albert to address this need. As part of this project, the Northrop Grumman Corporation designed and constructed an easy-to-use agent-based modeling environment called *Pythagoras*, which focused on the simulation and analysis of both human factors in military combat and noncombat situations (*34*). Pythagoras enables a user to create intelligent agents and assign them behaviors based on motivators and detractors. The agents can either act as individuals, or be loosely or tightly controlled by one or more leader agents within a command-and-control (C2) hierarchy. Pythagoras is written in Java,* making it platform-independent. It can be run in a multithreaded batch mode allowing for a large number of simulation instances to be run on a network of computers in a short time; it can also be used and run interactively on a PC through a graphical user interface (GUI); or it can be run in batch mode from a PC command prompt (*62*).

Figure F-1 depicts the intended design approach utilized by the Pythagoras modeling environment for the purpose of replicating a realistic military combat environment (*63*). To achieve these design objectives, Pythagoras offers a targeted set of capabilities in the area of time-stepped agent-based simulation. A description of these various capabilities follows.

*Java is a registered trademark of Oracle and its affiliates.

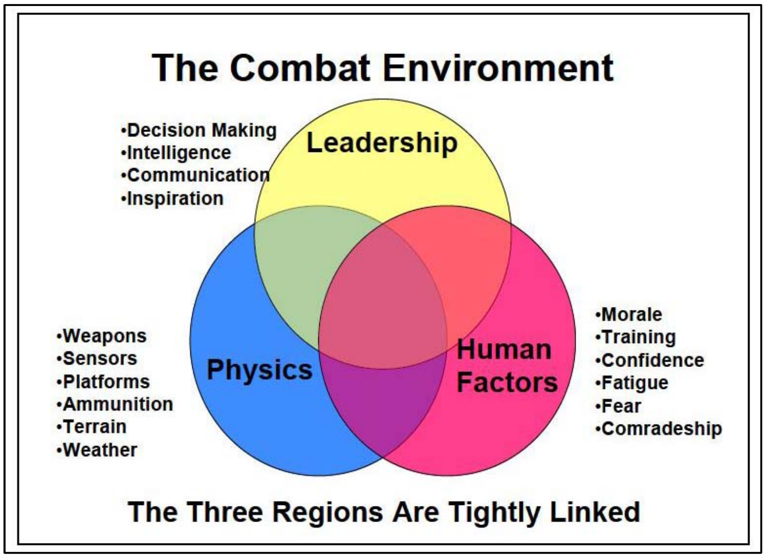

Figure F-1. An overview of what the Pythagoras modeling environment is designed to capture with regard to a combat environment.

A feature called *soft decision rules* allows the user to assign each agent its own behavior-triggering threshold level for all decision variables within an associated DMP. This approach models variation between individual agents by establishing a midpoint for the decision variable in question, and then allowing the user to provide a uniformly distributed range around that value. When an agent is instantiated at the beginning of a simulation run, it selects its decision variable values from the distribution at random. By modulating the spread, the Pythagoras user can instantiate agents as either very homogeneous (e.g., well-trained, disciplined military troops), heterogeneous (e.g., a crowd of various types of people within an urban environment), or some value in between.

Agents can be assigned *movement desires* to determine their movement paths as a scenario unfolds. In addition, agents can be assigned *shooting desires* to determine which potential targets an agent will engage with weapon fire. During each simulation decision cycle, an agent establishes which desires are active based on user-configured behavior triggering values. Then, based on decision variable computations, the agent uses the strengths of the movement and shooting desires to determine a direction of movement, or a specific target to engage, respectively.

As assigned by the user to agents, *sidedness*—or sociopolitical/military affiliation represented by an agent's color value—is governed by soft rules at the start of the simulation and can be changed over the course of the simulation by various events and actions. Pythagoras uses the terms *greenness*, *blueness*, and *redness* to make the properties generic (and to allow for visual display of the property in the scenario playback tool). Each of the three properties can take a value from 0–255 (corresponding to standard color monitor settings). As an example, figure F-2 illustrates the blueness property range for different agents within a notional Pythagoras scenario, where a deployed U.S. force is working jointly with a host nation (HN) indigenous force to combat an insurgent force integrated into the HN nonmilitary population (*64*). In this case, blueness is the agent sidedness property set by the user via the Pythagoras GUI (figure F-3) to encode the relative degree of "friendliness"—or friendly affiliation—an agent demonstrates toward the occupying U.S. force. Finally, for purposes of C2, military agents with similar color (as measured by the difference in absolute value) are considered to be members of the same higher-echelon unit, where smaller color differences between agents can be used to indicate membership in lower-echelon units, and identical color values between agents indicate membership in the same specific unit.

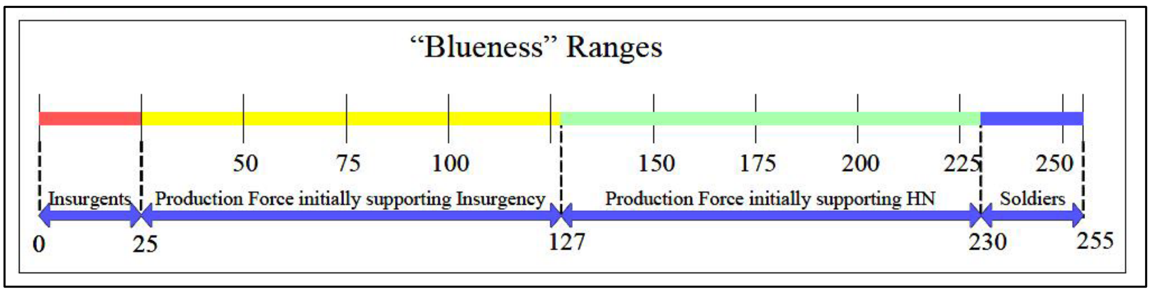

Figure F-2. Blueness property range for different military and paramilitary agents within a notional Pythagoras scenario.
Note: Here, values of blueness are used to affiliate scenario agents with membership in either, the deployed U.S. force (where, $230 \leq$ value ≤ 255); the host nation (HN) indigenous force supporting the currently recognized HN government (where, $127 \leq$ value < 230); the HN force supporting the insurgency (where, $25 <$ value ≤ 127); or the insurgent force (where, $0 \leq$ value ≤ 25).

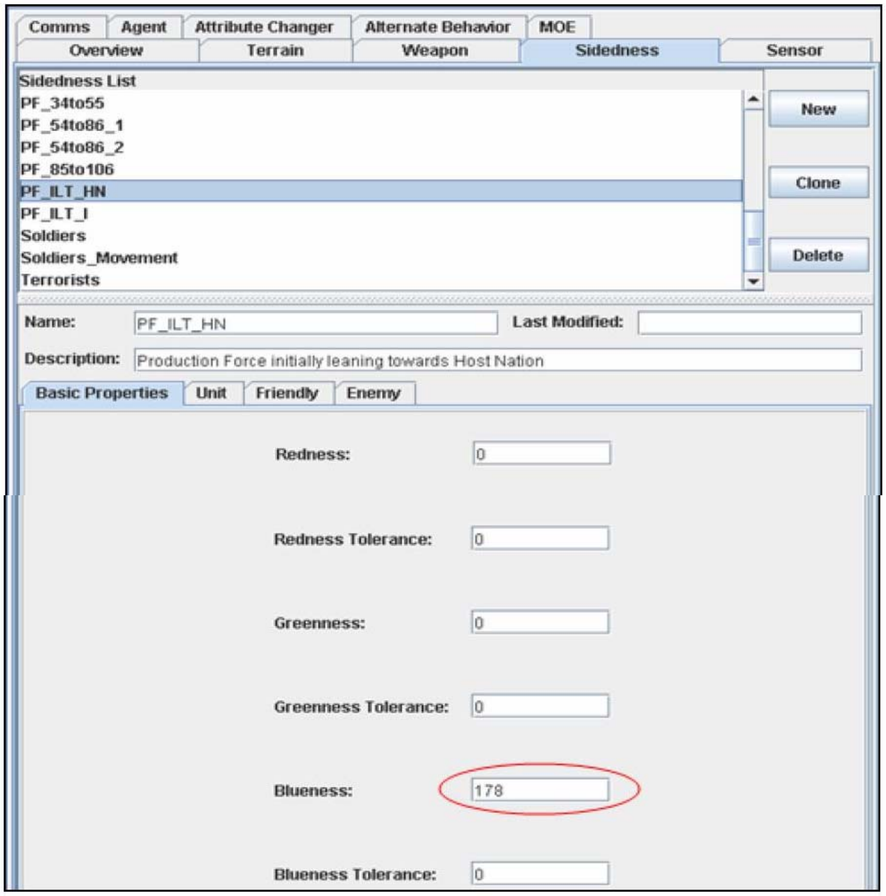

Figure F-3. Screenshot of the Pythagoras GUI.
Note: Here, the user has selected the "Sidedness" tab to set the blueness property value for agents belonging to the indigenous production force, initially loyal to the recognized HN government (PF_ILT_HN), within the notional Pythagoras scenario.

Similar to sidedness, Pythagoras also has three generic agent attributes—*alpha*, *beta*, and *gamma*. These generic attributes act as a supplement to sidedness, and as such they do not affect an agent's affiliation/sidedness. The meaning of alpha, beta, and gamma is up to the user to determine, based on the user's scenario. They could be used to represent intangible items such as fear, hunger, and morale, or something more concrete, such as health or wealth. The generic attributes are also governed by soft rules at the start of the simulation and can be changed over the course of the simulation by various events and actions. A change in the value of generic attribute can also cause a behavior-change event.

Pythagoras provides the user the option to configure agents with materiel-based weapon, sensor, and communications capabilities. Agents may carry as many as three different weapons, any of which can be set up as either *direct-fire* (which requires a line of sight), or *indirect-fire* (which does not). Additionally, each agent may have up to three different sensors, each of which operates in a specific signature band (labeled A, B, or C). Similarly, an agent can possess up to three communication devices, each of which operates either via line-of-sight or in a broadcast mode, and allows the agent to use the devices to talk, listen, or both. Finally, each communication device also operates via a specific channel or channels (up to three allowed).

Within a Pythagoras simulation instance, all agents exist in a user-defined two-dimensional (2-D) "playbox" of up to 1000 × 1000 pixels (figure F-4). In the playbox, a user can create terrain features; e.g., polygons representing buildings that have a floor, a ceiling, and factors for mobility; shapes that can provide agent concealment in each of the three signature bands; and other terrain shapes providing protection that reduce a weapon's effectiveness. Currently, a pixel can be associated with one terrain feature at the most. Agents can either move on the terrain or operate at an altitude above the terrain.

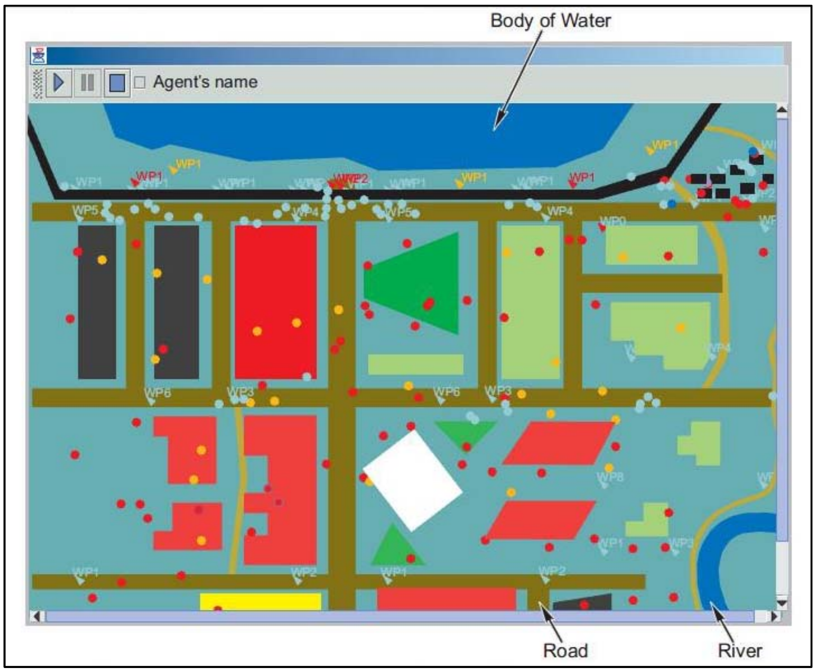

Figure F-4. Pythagoras simulation playbox associated with an antiterrorist/force-protection in village patrol scenario (*62*).

Considering that the primary focus of Pythagoras lies in the modeling of various human factors within a military context (and then simulating the resultant operational dynamics within that context), we believe this agent-based simulation system provides some limited capability to facilitate SoS modeling and associated SoSA. Given that agent behaviors in a Pythagoras simulation are based on user-supplied *a priori* (vice experience-based) motivators and detractors, it is likely that, as was the case with Warcon (see appendix D), this type of simulation system could at best emulate a "quasi-" SoS dynamically operating at a single timescale (given that Pythagoras is a time-stepped model) and utilizing a limited form of nested concepts and inter-related purposes. In this case, commander intent and military concept of operations can be indirectly represented to some degree (by using the agent sidedness property in combination with agent behavior, triggering to architect a somewhat inefficient C2 structure); but purposeful behavior—as demonstrated by participating military agents—would be difficult to capture.

On the other hand, its primary focus on accurately representing human factors in an operational context suggests that Pythagoras does a good job addressing and simulating the sociotechnical aspects of a military SoS. From an SoSA perspective, Pythagoras does appear to demonstrate a standard time-series analysis capability, but little else. Thus, table F-1 reports our estimates of the associated SoS and SoSA ontology attribute similarity scoring values for Pythagoras (table F-1 [a] and [b], respectively). Given that Pythagoras was initially released in 2003 and continues to be actively developed by the U.S. Naval Postgraduate School (NPS) in Monterey, CA (*64, 65*), table F-2 reports on evidenced associated SoS and SoSA constituent component developmental maturation scoring values (table F-2 [a] and [b], respectively). Because NPS makes the Pythagoras *executable* easily available to U.S. Government personnel and their affiliates (*66*), and although Pythagoras *source code* is likely (but not definitely) Northrop Grumman intellectual property, we have assigned an accessibility status of *yellow* to this item. Finally, figure F-5 depicts the quadrant chart states of the SoS and associated SoSA designs associated with Pythagoras (i.e., [1.88, 2.67, *yellow*] and [1.39, 3.00, *yellow*], respectively).

Table F-1. SoS and SoSA ontology attribute similarity scores associated with the Pythagoras agent-based simulation system.
Note: see section 3 for a discussion of the criteria, attributes, and weights.

(a) System of systems				
	Attribute			
	Description	Number	Weight	Score
Nested Concepts and Inter-Related Purposes →	Commander Intent	1	10	2
	Concept of Operations	2	10	2
	Purposeful Systems	3	7	1
Dynamic and Evolving Processes →	Synchronous Behaviors	4	10	2
	Multiple Time Scales	5	7	1
	Nonstationary and Nonergodic Processes	6	5	2
	Sociotechnical Systems Theory	7	7	3
(b) System of systems analysis				
Analysis of Nested Concepts and Inter-Related Purposes →	Commander Intent	1	7	1
	Concept of Operations	2	7	1
	Synthetic Approach	3	10	1
Analysis of Dynamic and Evolving Processes →	State-Space Analysis	4	5	1
	Time-Series Analysis	5	10	3
	Frequency-Domain Analysis	6	5	1
Sociotechnical Systems Analysis →	Adaptive Multiscale Techniques	7	7	1

Table F-2. SoS and SoSA software component maturation scores associated with the Pythagoras agent-based simulation system.

	Component			
	Description	Number	Weight	Score
(a) System of systems	Simulation Engine	1	10	2
	User GUI	2	10	3
	Playbox Display	3	10	3
(b) System of systems analysis	Simulation Analysis Software	1	10	3

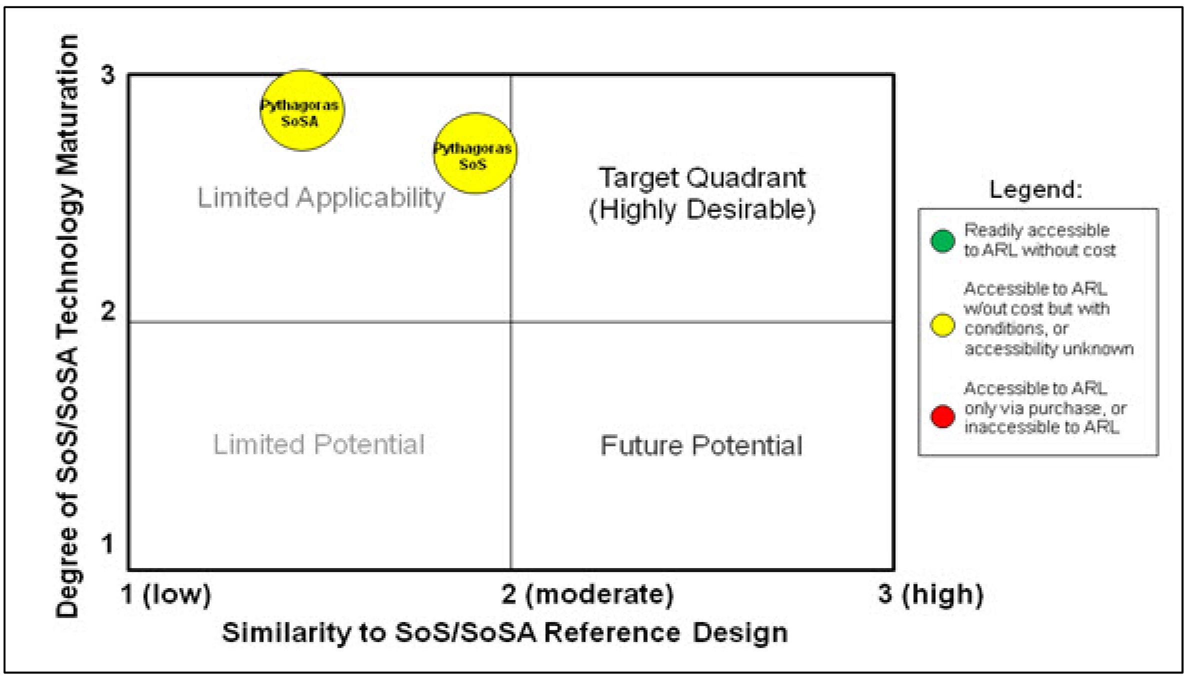

Figure F-5. Quadrant chart addressing SoS and associated SoSA designs for the Pythagoras agent-based simulation system.

INTENTIONALLY LEFT BLANK.

Appendix G. The Peace Support Operations Model (PSOM)

In 2004, the United Kingdom (UK) Ministry of Defence (MOD) mandated the creation of a dedicated program to analyze the Peace Support Operations (PSO) "problem space." This led to the consequent development by the UK Defence Science and Technology Laboratory (DSTL) of the *Peace Support Operations Model* (PSOM), a faction-to-faction, time-stepped, cellular geography, semi-agent-based model that was designed initially to represent a range of civil and military aspects of PSO (*35*). It encompasses Crisis Management Operations (North Atlantic Treaty Organization [NATO] Crisis Response), Security and Stabilization Operations (including counter-insurgency [COIN] activity) in conflict and postconflict environments (including preventive deployment, postintervention and the early stages of civil war). The military doctrine utilized within PSOM is generally consistent with emerging U.S. and UK operational concepts (for example, US FM 3-24] and FM 3-07] and their approximate UK parallels JDP 3-40 and AFM COIN). The general purpose and intent of the model is to demonstrate the impact of—and links between—policy decisions, strategic choices, and subsequent operational effects.

PSOM operates at two mutually dependent levels corresponding to different levels of command-and-control (C2) granularity (*69*). As described by Strong (*70*), the first of these levels is the superior *Strategic Interaction Process* (SIP) level that simulates political and strategic decision making, which interfaces with the population-centric framework established by the subordinate Operational Game (OG) level (figure G-1). Technically, the SIP is a human-in-the-loop wargaming exercise wherein the "control team" (i.e., the human players) take the roles of,

- "Blue," military force commander (and hierarchical subordinates as required);
- "Green," host nation government (e.g., president) and local security forces (e.g., army and police force command);
- "Red," insurgent, terrorist, militia, and criminal group leaders within the scenario;
- "White," leadership from other government departments (OGDs), nongovernmental organizations (NGOs), and international organizations (IOs);
- "Grey," leadership from various internal host nation civilian population groups as required.

During each PSOM update cycle, the personnel within this control team generates the following products:

- *Strategic Summary Slide*, which outlines what data each wargamed faction (as represented by members of the control team), used as the basis of its decision-making process (DMP) (figure G-2);

- *Media Slide* representing the local, regional, and international media's perceptions of recent events and developing trends following each wargame update, and the higher level political analysis (journal and think tank publications) following every third update;
- *Intent Slide*, outlining the overall strategic dispositions, planning assumptions, and intentions of key military components and units (figure G-3).

Finally, the information contained in these control team products is used to generate the sets of operational orders (with one set of orders associated with each wargamed faction) that are fed into the OG level.

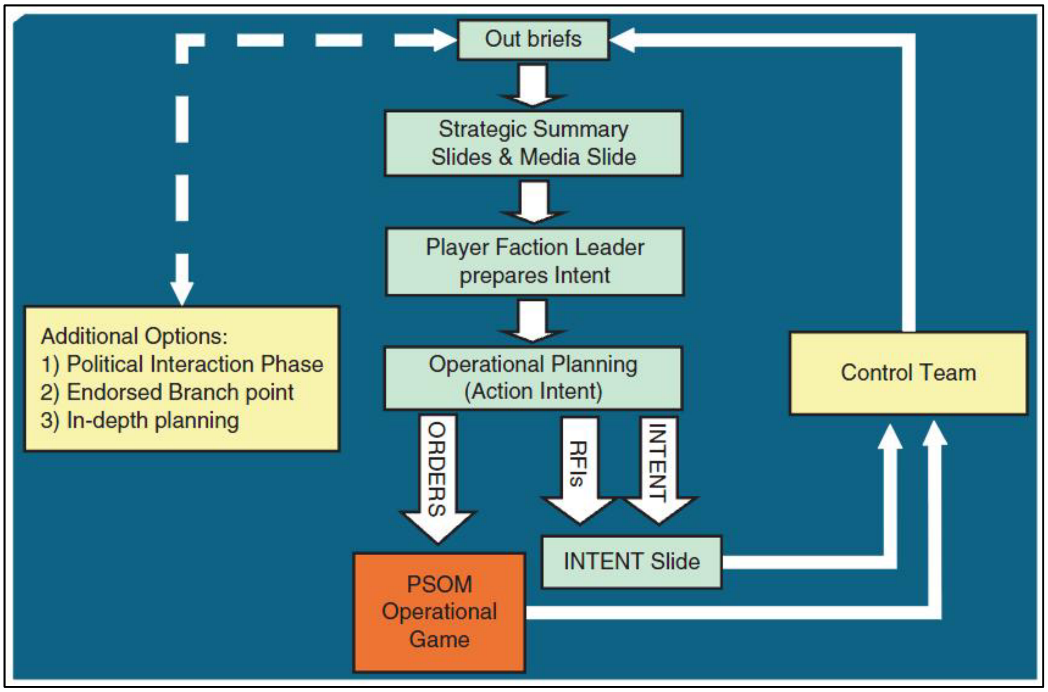

Figure G-1. Overview of the SIP.

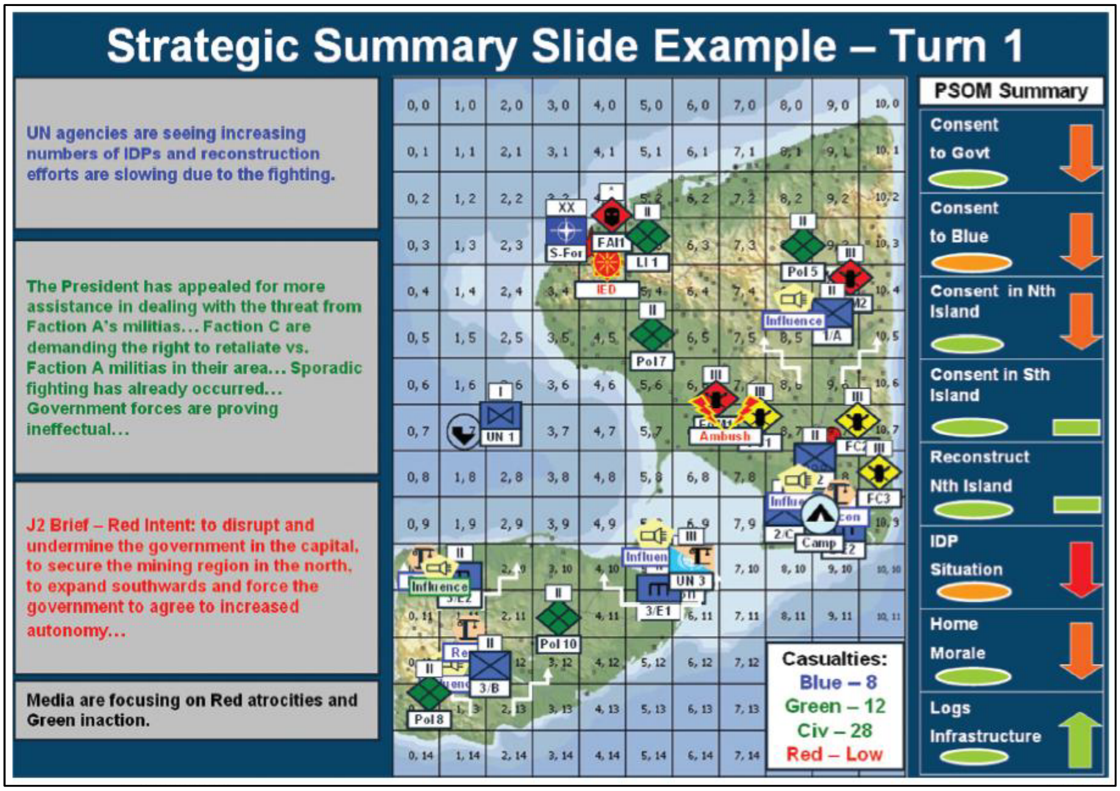

Figure G-2. Example of a Strategic Summary Slide as generated within the PSOM SIP.

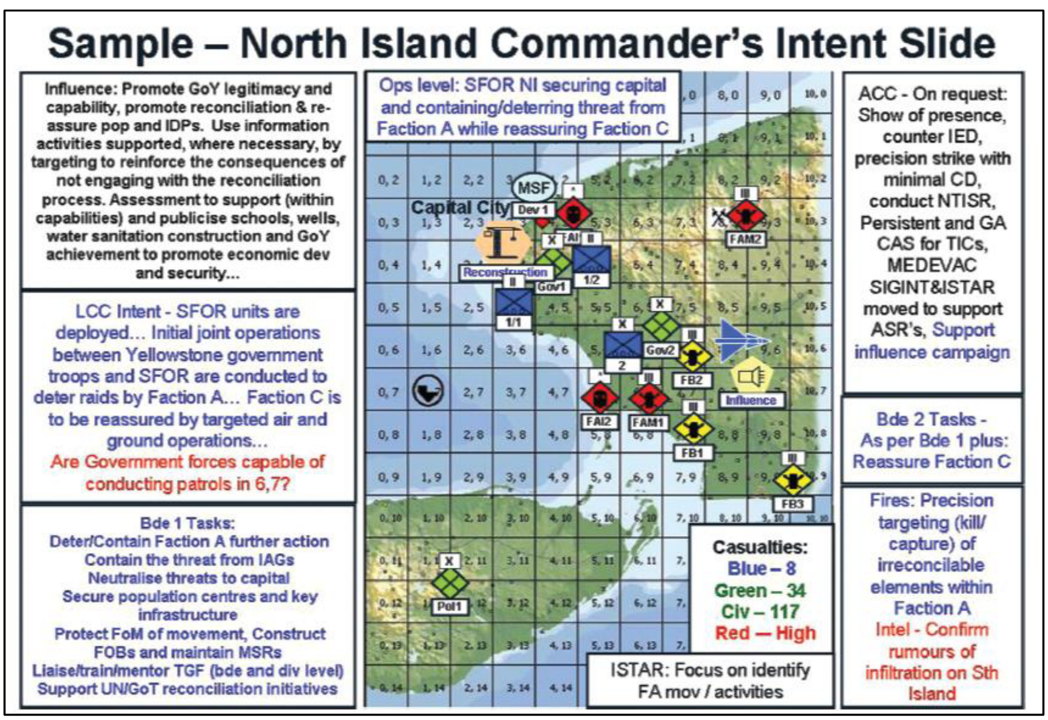

Figure G-3. Example of an Intent Slide as generated within the PSOM SIP.

As described by Body and Marston (*35*), and Hanley and Gaffney (*69*), the subordinate OG level within PSOM translates the strategic decisions made by the SIP into tactical campaign effects generated by military units and civilian "teams." It is a computer simulation where the civilian population is modeled as a set of discrete agents, with their own behaviors, DMPs, and information gathering properties. Conversely, the military, in-theater insurgent, NGO and OGD units, which comprise the OG level, do not have the information-gathering or decision-making capabilities that typify an agent model. However, these operational units do have some very limited decision options available to them, such as prosecution of enemy targets via kinetic weapon engagement. The OG also represents a number of nonkinetic engagement activities that can be undertaken by all factions, such as influencing the opinions of the host nation population through psychological Information Operations. The primary military maneuver units are battle-groups (i.e., infantry battalions or armored regiments), while their civilian equivalents are reconstruction "groups." Finally, the level of unit granularity within the OG level can be scaled down to military company or civilian agency team as a function of simulation scenario size, but parallel work on the (currently under development) tactical level Stabilization Operations Analysis Tool (STOAT) is intended to directly address this fine-grained simulation requirement.*

The OG adjudication process within a PSOM simulation update involves three sequential stages: (1) operational units engage and influence the general state of the theater of operations; (2) economic effects occur in response to military engagement activities; (3) the civilian population reacts to the new resultant situation. The first of these stages reflects traditional military modeling, and comprises the elements of information gathering, contact generation, target prosecution and casualty calculation (figure G-4). The second-stage economic response model first assesses each faction's stock of human capital, state of physical infrastructure elements (e.g., electricity plants) and liquid capital (i.e., cash), then combines these factors to estimate a subsequent produced quantity of economic goods via a Cobb-Douglas production function (figure G-5). Finally, the reactive population attitude model represents the opinions of individual agents (collectively representing the host nation civilian population) in response to prior military and economic activities as a function of "threat" (the degree of involuntary support induced by a faction on an agent), and "consent" (the degree of voluntary support for a faction offered by an agent). These two factors are assessed in most population-based decisions, and drive overall societal behaviors, such as population recruitment and human intelligence (HUMINT)

---

*As described by Gaffney and Vincent, the design purpose behind STOAT is to support the representation of human-based IO at the tactical level (*71*). Conceptually, STOAT is a multisided, time-stepped, computer-based wargame simulation, consisting of a number of factions controlled by human players and a civilian population represented by autonomous agents. However, the geographic area represented within a STOAT simulation instance is much smaller than that represented in PSOM (typically, the entire STOAT Area of Operations will be of a similar size to a single PSOM map cell), and the former's update cycles represent a shorter period of time (i.e., one STOAT simulation update will represent one week of real time). Finally, STOAT will model populations as groups, and thus will focus on differences in the trust of information sources at the group level, rather than attempting to represent variation between individual humans.

generation. “Threat” and “consent” are properties of individual agents, representing an agent's subjective perception of each faction.

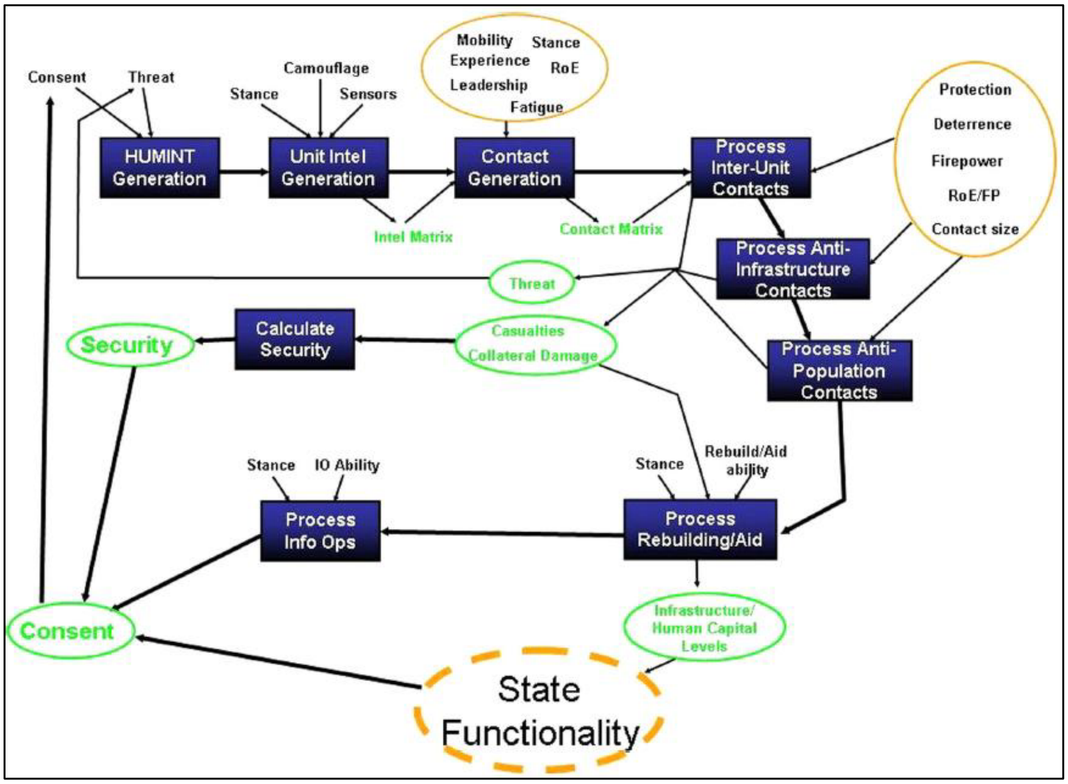

Figure G-4. Military models within the PSOM OG level.

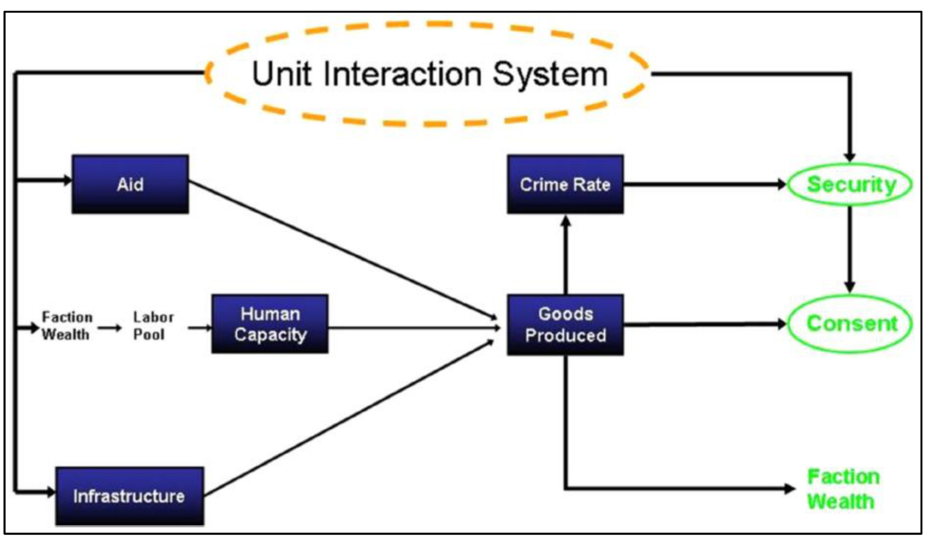

Figure G-5. Economic effects model within the PSOM OG level.
Note: Here, the economy of the simulated theater of operations responds to the general state of affairs resulting from interfaction military combat operations.

When the SIP and OG levels are run in conjunction during a PSOM wargame instance (i.e., a combination of human-in-the-loop and computer-based simulation), a standard situation update cycle typically represents one month of real time, with no finer-grained time breakdown within an update. PSOM scenario runs are normally kept to a minimum of 12 update cycles (i.e., 12 months of real time), with the maximum number of cycles run in a wargame instance to date being 48 cycles of game time (i.e., four years of real time). Geographically, PSOM breaks a scenario map down into a grid of individual square cells (where, 1 cell = 50 km × 50 km) on a total map area representing 1000 km × 1000 km. Each cell has various properties, such as overall terrain type within the cell, population density, economic features and ethnic breakdown. In addition, dynamic interaction between cells is limited to the transfer of information from the populations and factions occupying one cell to those occupying neighboring cells.

Given that the decision-making entities within a standard PSOM scenario-oriented wargame include a combination of humans (representing strategic-level decision makers), situationally aware and responsive software agents (representing a nonmilitary population of civilian decision makers), and far more primitive simple reactive processes (representing operational/tactical-level military and paramilitary forces with very limited decision-making capability); it is difficult to ascertain precisely how much utility this model could provide to military SoS simulation and analysis. Consequently, we believe this "quasi-agent-based" simulation system provides at best some limited capability to facilitate SoS modeling and associated SoSA. With regard to the former, this type of human-in-the-loop/computer-based simulation could effectively emulate a military SoS only at the strategic level (where purposeful human "agents" could intelligently apply all aspects of nested concepts and inter-related purposes, and also operate as a realistic sociotechnical *strategic* system); at higher granularities, however, effective military SoS emulation is unlikely. Also, considering PSOM's *very* coarse-grained characteristic timescale, its ability to properly emulate dynamic and evolving processes at typical military operations timescales is nonexistent. From an SoSA perspective, PSOM does appear to demonstrate a very minimal time-series analysis capability, but little else. Thus, table G-1 reports our estimates of the associated SoS and SoSA ontology attribute similarity scoring values for PSOM (table G-1 [a] and [b], respectively). Given reported continuing PSOM development and application in the study areas of strategic communication influence on civilian populations for Stabilization Operations (*72*) and operational aspects of Security Sector Reform (SSR) activities (*73*), table G-2 presents on evidenced associated SoS and SoSA constituent component developmental maturation scoring values* (table G-2 [a] and [b], respectively). Although PSOM has recently been used by the U.S. Naval Postgraduate School to analyze peace-keeping operations (*74*)—implying that the PSOM software *executable* is available to U.S. Government personnel and their affiliates—it is unclear whether PSOM *source code* is equally available. Accordingly, we

*These values consider continuing development, at the primary developer UK DSTL as well as at the U.S. Naval Postgraduate School (*39*).

have assigned an accessibility status of *yellow* to this model. Finally, figure G-6 depicts the quadrant chart states of the SoS and associated SoSA designs associated with PSOM (i.e., [1.82, 2.25, *yellow*] and [1.67, 2.00, *yellow*], respectively).

Table G-1. SoS and SoSA ontology attribute similarity scores associated with the PSOM. Note: see section 3 for a discussion of the criteria, attributes, and weights.

(a) System of systems				
	Attribute			
	Description	Number	Weight	Score
Nested Concepts and Inter-Related Purposes →	Commander Intent	1	10	2
	Concept of Operations	2	10	2
	Purposeful Systems	3	7	2
Dynamic and Evolving Processes →	Synchronous Behaviors	4	10	1
	Multiple Time Scales	5	7	1
	Nonstationary and Nonergodic Processes	6	5	2
	Sociotechnical Systems Theory	7	7	3
(b) System of systems analysis				
Analysis of Nested Concepts and Inter-Related Purposes →	Commander Intent	1	7	2
	Concept of Operations	2	7	2
	Synthetic Approach	3	10	2
Analysis of Dynamic and Evolving Processes →	State-Space Analysis	4	5	1
	Time-Series Analysis	5	10	2
	Frequency-Domain Analysis	6	5	1
Sociotechnical Systems Analysis →	Adaptive Multiscale Techniques	7	7	1

Table G-2. SoS and SoSA software component maturation scores associated with the PSOM.

	Component			
	Description	Number	Weight	Score
(a) System of systems	Human-in-the-loop SIP	1	10	3
	OG Military Model	2	10	2
	OG Economic Model	3	10	2
	OG Civilian Population Model	4	10	2
(b) System of systems analysis	Simulation Analysis Software	1	10	2

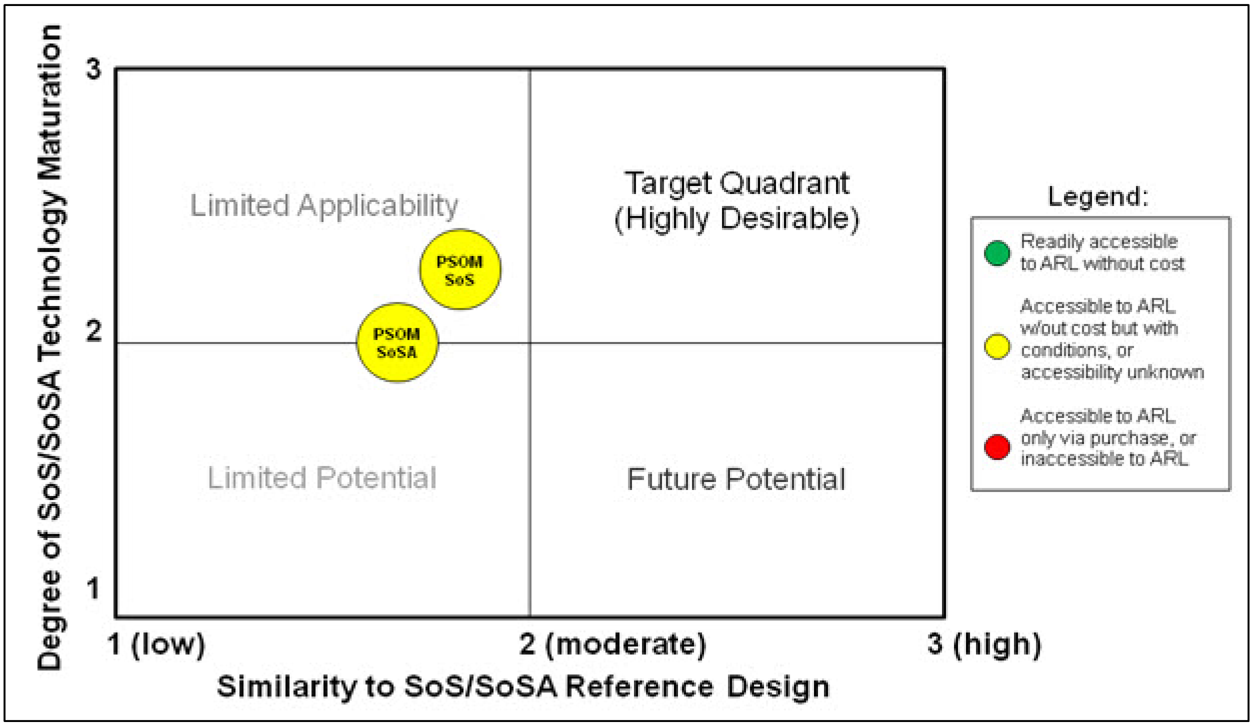

Figure G-6. Quadrant chart addressing SoS and associated SoSA designs for PSOM.

Appendix H. The Enhanced ISAAC Neural Simulation Toolkit (EINSTein)

As designed and constructed by Ilachinski, the *Enhanced ISAAC Neural Simulation Toolkit* (EINSTein) is an adaptive agent-based model of land-based military combat, which evolved out of a more far-reaching project to develop a new fundamental theory of warfare based upon Complexity Theory (*36–40*). EINSTein is actually an extension of an earlier proof-of-concept cellular automata- (CA) based combat model called *Irreducible Semi-Autonomous Adaptive Combat* (ISAAC) that was previously developed by Ilachinski for use by the U.S. Marine Corps (*75, 76*). Given that "artificial life" techniques (i.e., agent-based models and evolutionary learning algorithms) could potentially provide novel technical insight into understanding some of the fundamental processes of war; EINSTein was designed to function as a simple artificial-life-like "toy model" of combat, for purposes of concept exploration. In particular, EINSTein is designed to illustrate how the network of dynamic interactions evolving between and among notional combatants at the microscale can produce certain resultant patterns of land combat, which can be viewed at the macroscale as self-organized, emergent phenomena. This is achieved via EINSTein's bottom-up/synthetic approach to the modeling of combat (vice the more traditional top-down/reductionist approach taken by conventional military models), representing a step towards the ultimate development of a complex systems theoretic toolbox for identifying, exploring, and possibly exploiting self-organized emergent collective patterns of behavior on the real battlefield.

A screenshot of the main EINSTein front-end graphical user interface (GUI) from a typical user session is portrayed in figure H-1. Here, the user is presented with an assortment of configuration options for setting up and running a two-sided (i.e., Blue force and Red force) EINSTein combat simulation. The screenshot contains three active windows illustrating the state of a simulated two-dimensional (2-D) battlefield.

- *Main battlefield view* (includes passable and impassable terrain elements) is a visual representation of the discrete cellular space that mobile automata* agents occupy and operate within.
- *Trace view* displays color-coded territorial occupancy as occupied by a particular force.
- *Combat view* provides a gray-scaled filter of relative combat intensity across the battlefield.

**Mobile automata* are a class of automata similar to cellular automata, but which have a single "active" cell (that appears to move through progressive time-stepped cell updating), instead of updating all cells in parallel. In a mobile automaton, the cell updating rules apply only to the active cell, and also specify how the active cell "moves" from one simulation update to the next. All cells that are not currently active remain the same from one update to the next. Mobile automata can, therefore, be considered a hybrid between elementary cellular automata and Turing machines (*77*).

All of these views of combat adjudication are simultaneously updated during a simulation time step. On the right-hand side of the figure two data dialogs appear, which display adjustable Blue and Red agent parameter values. Appearing on the lower left side of the figure are time-series graphs of Blue and Red force center-of-mass coordinates (as measured from the Red flag, to reach what defines the Blue mission objective in this context) and the average number of agents within the Blue and Red agents' sensor ranges. Finally, on the lower right side of the figure is a dialog box, which allows the user to configure communication networks among individual squads of agents, within a specific force.

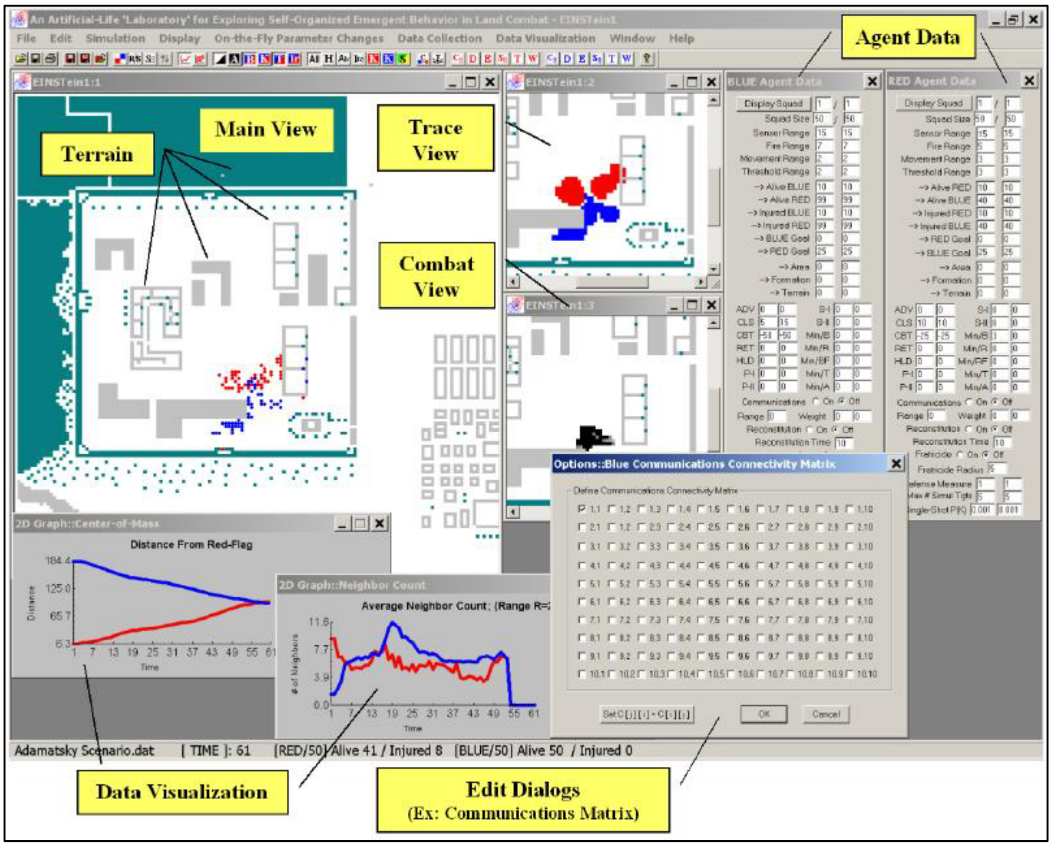

Figure H-1. Screenshot of EINSTein GUI.

Figure H-2 displays various screen captures of macroscale spatial patterns resulting from 16 different microscale agent DMPs—illustrating the diversity of agent behaviors—which emerge out of a relatively simple set of rules. It should be noted that the sample patterns shown here correspond to opposing Blue and Red forces consisting of a *single* company-size military unit. EINSTein can potentially facilitate higher echelons of Blue force/Red force combat scenarios, in which agents belonging to different companies obey different DMPs, and interact with one another according to an additional layer of rules. It is also important to note that the agent behaviors displayed in this figure are not "hard-wired," or scripted, but are rather, an emergent property of a decentralized—but dynamically interdependent—swarm of agents. Such behaviors can be evolved over multiple generations of agents by the EINSTEIN user via application of an embedded genetic algorithm (GA) capability, allowing the user to experimentally "breed" agent teams for prosecution of specific mission types (wherein a set of selectable macroscale measures of effectiveness function as the GA fitness function relative to a specific mission scenario).

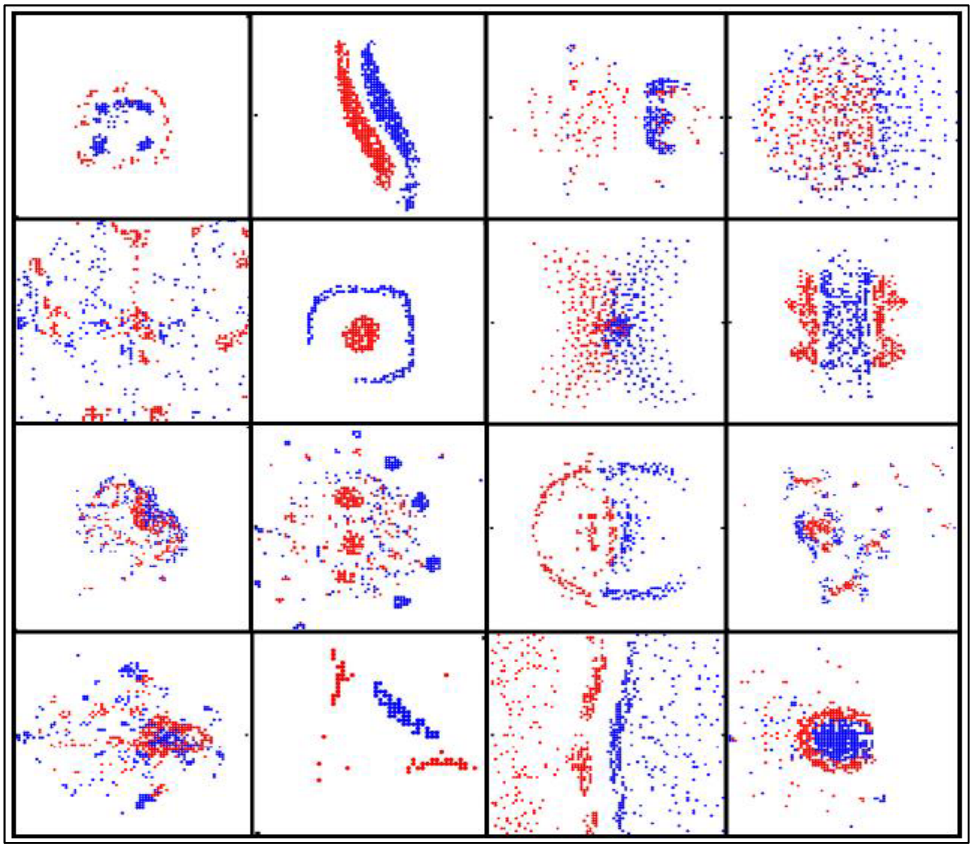

Figure H-2. Examples of macroscale emergent spatial patterns resulting from EINSTein simulation of opposing Blue force/Red force combat interactions, reflecting a variety of microscale agent DMPs.
Note: Here, each of the 16 squares represents a single screenshot of a simulation instance, where each instance utilizes different microscale agent DMPs, for Blue and Red forces.

As part of its simulation analysis capability, EINSTein has a data visualization package to facilitate the exploration and understanding of multiple high-dimensional co-evolutionary fitness landscapes. Ilachinski (the EINSTein developer) interprets the latter concept as a type of response hyper-surface that inter-relates all of the different measurable parameters, characterizing a combat force into an integrated multidimensional contextual representation of demonstrated mission effectiveness (*36, 37, 39, 40*). To this end, figure H-3 illustrates an example two-parameter fitness landscape, generated by an EINSTein batch-run sequence of simulations, which progressively and parametrically explore a user-defined 2-D "slice" of the full *N*-dimensional parameter space associated with the Red force. In this situational context, the Blue force materiel and behavioral configuration remains fixed, or "clamped," and nonmutable throughout a given batch-run set of simulations. To generate the data displayed in the figure, the user identifies the two Red force parameters over which the combat unit's operational behavior will be sampled. In this case, those two parameters are "aggressiveness"—i.e., an agent's weighted tendency to either, advance toward (positive values), or retreat away from (negative values), an armed enemy agent—and Red agent sensor range. To each (x, y) combination of variable parameters, with all other Red agent parameters held constant, EINSTein associates a notional measure of "mission fitness." Mission fitness is a quantitative measure of how well the Red force agents have achieved a user-defined mission objective normalized to an optimal level of mission achievement, and computed as an average over a desired number of scenario initial conditions (for Blue and Red initial force disposition). Once generated, this fitness landscape representation of simulation data provides an intuitive visual means to facilitate the EINSTein user's understanding of (in this case) the nonmonotonic emergent relationship existing between a force's combat aggressiveness, sensor capability (in terms of maximal range), and relative mission success.

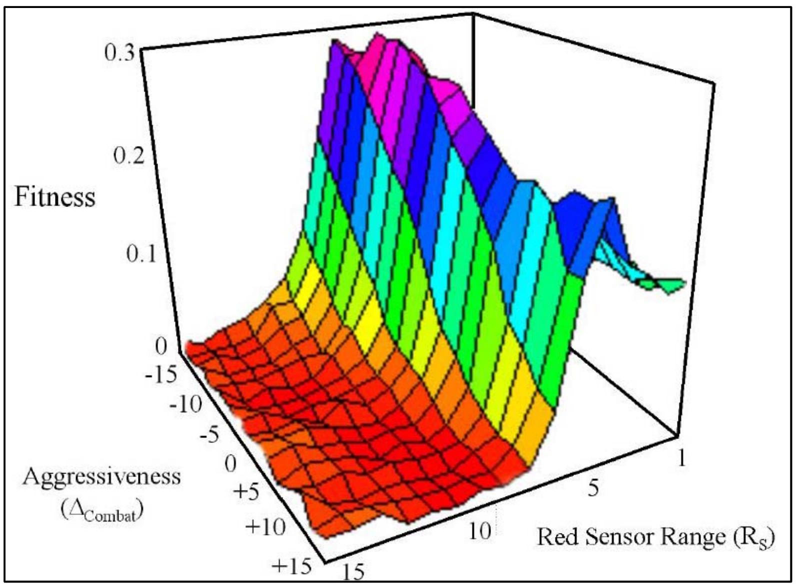

Figure H-3. 2-D fitness landscape.

Note: 2-D fitness landscape for the Red force mission objective "maximize number of Red agents near Blue flag" (i.e., a fitness value normalized to an optimal quantifiable realization of Red force operational success) as a function of combat aggressiveness (Δ_{Combat}) and Red agent sensor range (R_S). In this context, higher fitness values indicate better Red force performance. Note that this particular measure of mission fitness *does not* scale monotonically with sensor range.

Given that Ilachinski has explicitly described EINSTein as a so-called "toy model" of combat, utilizing organizationally "decentralized but dynamically interdependent" agent combatants, we believe this agent-based simulation toolkit provides some application-conditional capability to facilitate SoS modeling and associated SoSA (e.g., in certain types of urban situations where simple combat behaviors and loosely-organized command-and-control (C2) might be mission appropriate). As was the case with Pythagoras and Warcon, this type of simulation system could at best emulate a "quasi-" SoS, dynamically operating at a single timescale (given that EINSTein is a relatively simple time-stepped model) and utilizing a *very* limited and weakly-coupled form of nested concepts and inter-related purposes. In this approach, however, only the most indirect and implicit application of the paradigms of commander intent, military concept of operations, and purposeful behavior could be realized (in the last case, only an "artificial" emulation of co-evolving multigenerational agent purpose can be represented via the EINSTein GA capability). Additionally, the simplistic "toy model" nature of agent DMPs within an EINSTein simulation provides little contribution from the perspective of military sociotechnical system emulation.

From an SoSA perspective, EINSTein presents a somewhat stronger case. The time-series analysis capability offers several nonstandard metrics available to the user (e.g., dynamic "combat entropy").* Also, the fitness landscape representation of simulation data provides an innovative parametric approach to facilitate user understanding of the synthesis of macroscale combat effects and patterns from microscale agent attributes, and the overall state-/phase-space structure of a dynamic (albeit vastly simplified) military SoS.

Table H-1 reports our estimates of the associated SoS and SoSA ontology attribute similarity scoring values for EINSTein (table H-1 [a] and [b], respectively). Given that EINSTein was consistently and progressively developed from its initial release in 1999 until 2004, but has apparently remained at an ambiguous developmental status since that time, table H-2 reports on evidenced associated SoS and SoSA constituent component developmental maturation scoring values (table H-2 [a] and [b], respectively). As was the case with several of the other SoS/SoSA projects we have reviewed, the EINSTein *executable* is readily available to anyone with Internet access (*78*), while EINSTein *source code* availability status remains unknown. Accordingly, we have assigned an accessibility status of *yellow* to this item. Finally, figure H-4 depicts the quadrant chart states of the SoS and associated SoSA designs associated with EINSTein (i.e., [1.27, 3.00, *yellow*] and [2.08, 2.50, *yellow*], respectively).

---

*Carvalho-Rodriques has suggested using entropy (as computed from casualty reports) as a predictor of combat outcomes (*48*). This is consistent with the interpretation of military combat as a dissipative dynamical system. Thus, the casualty-based *combat entropy* $E_i(t)$ is defined as

$$E_i(t) = \frac{c_i(t)}{N_i} \log \frac{c_i(t)}{N_i} \tag{1}$$

where $c_i(t)$ represents the casualty count (as a function of time t) and N_i represents the initial force strength of the i^{th} military combat force. Adversary (red or blue). This function has a peak at about 0.37, which Woodcock and Dockery (using extensive data drawn from detailed historical records of evolving casualty counts in various battle scenarios) interpret as the point in battle where the integrative combat capability of a military force begins to disintegrate with further casualties (*49*).

Table H-1. SoS and SoSA ontology attribute similarity scores associated with the Enhanced ISAAC Neural Simulation Toolkit (EINSTein).
Note: see section 3 for a discussion of the criteria, attributes, and weights.

(a) System of systems				
	Attribute			
	Description	Number	Weight	Score
Nested Concepts and Inter-Related Purposes →	Commander Intent	1	10	1
	Concept of Operations	2	10	1
	Purposeful Systems	3	7	1
Dynamic and Evolving Processes →	Synchronous Behaviors	4	10	2
	Multiple Time Scales	5	7	1
	Nonstationary and Nonergodic Processes	6	5	2
	Sociotechnical Systems Theory	7	7	1
(b) System of systems analysis				
Analysis of Nested Concepts and Inter-Related Purposes →	Commander Intent	1	7	1
	Concept of Operations	2	7	1
	Synthetic Approach	3	10	3
Analysis of Dynamic and Evolving Processes →	State-Space Analysis	4	5	3
	Time-Series Analysis	5	10	3
	Frequency-Domain Analysis	6	5	2
Sociotechnical Systems Analysis →	Adaptive Multiscale Techniques	7	7	1

Table H-2. SoS and SoSA software component maturation scores associated with the Enhanced ISAAC Neural Simulation Toolkit (EINSTein).

	Component			
	Description	Number	Weight	Score
(a) System of systems	Simulation Engine	1	10	3
	GUI	2	10	3
(b) System of systems analysis	Data Collection Software	1	10	3
	Data Visualization Software	2	10	2

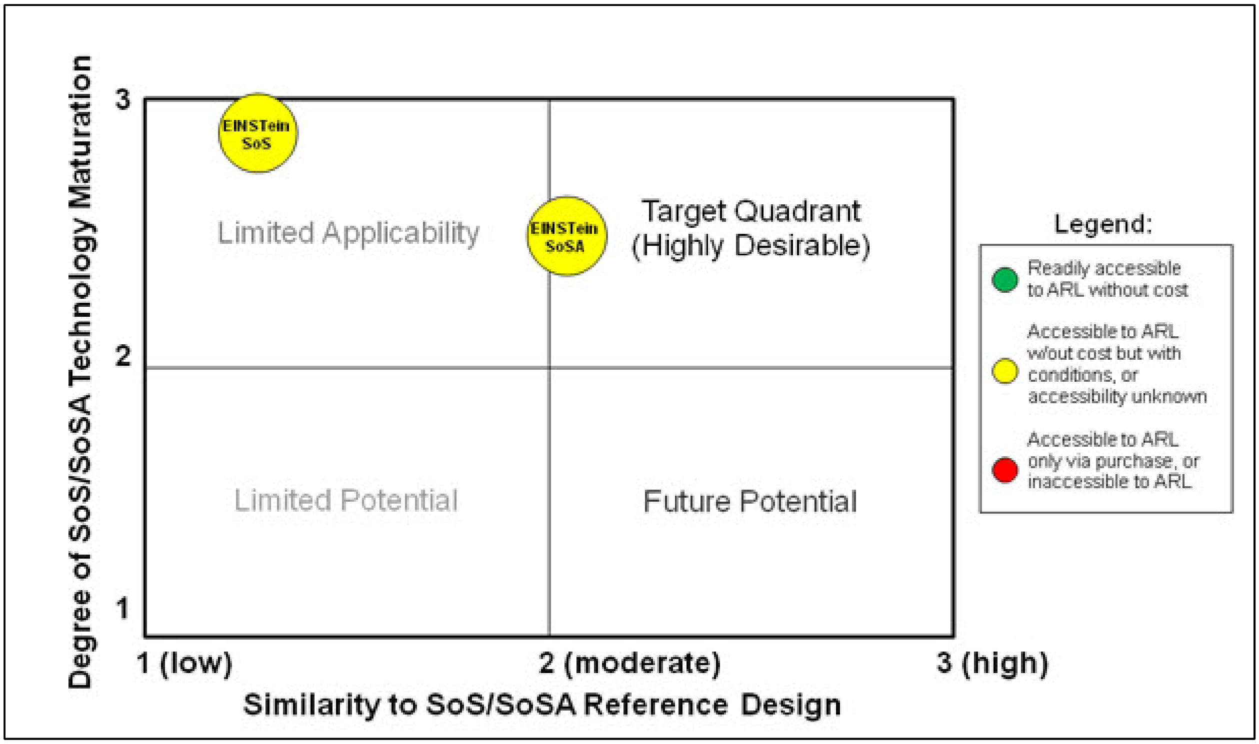

Figure H-4. Quadrant chart addressing SoS and associated SoSA designs for the Enhanced ISAAC Neural Simulation Toolkit (EINSTein).

INTENTIONALLY LEFT BLANK.

Appendix I. The Map Aware Nonuniform Automata (MANA) Agent-Based Model

Map Aware Nonuniform Automata (MANA) is an agent-based model developed by the Operations Analysis team at the Defence Technology Agency (DTA) in New Zealand (*41–46*). Initial development of MANA by DTA commenced in approximately the year 2000, with primary technical design inspiration coming from Ilachinski's mobile cellular automata (CA)-based model ISAAC (*75,76*). As a result of this inspiration, combative Blue and Red mobile automata agents within a MANA simulation inhabit a two-dimensional (2-D) cellular world (figure I-1), where the agent movement DMP is modulated by personality vector weightings in a similar way as used in the previously-discussed agent-based model, ISAAC. The agents' rules for movement are a function of ten different competing goal conditions that are weighted to reflect various personality types (e.g., aggressive, cautious, curious, etc.), where the MANA user can also define specific movement behavior triggering conditions as a function of the state of the agent's perceived local environment. On the other hand, the enemy engagement DMP utilized by MANA agents reflects no purposeful goal-directed target selection capability; instead, targets are chosen at random from those available. In this sense, MANA is from a class of military agent-based models that are purposefully developed without detailed physical attributes of the military entities concerned (with the assumption that such details are not expected to have any bearing on the study at hand). This allows scenarios to be run relatively fast—over many excursions—in order to discover unique situations or tactics where friendly forces can achieve dominance over an enemy force. Because of this last capability, MANA is being used by a number of military colleges and operations-research-based organizations amongst member nations of The Technical Cooperation Program (TTCP), and has also been used for various Master's theses at the Naval Postgraduate School (NPS) in Monterey, CA (*e.g., 79*).

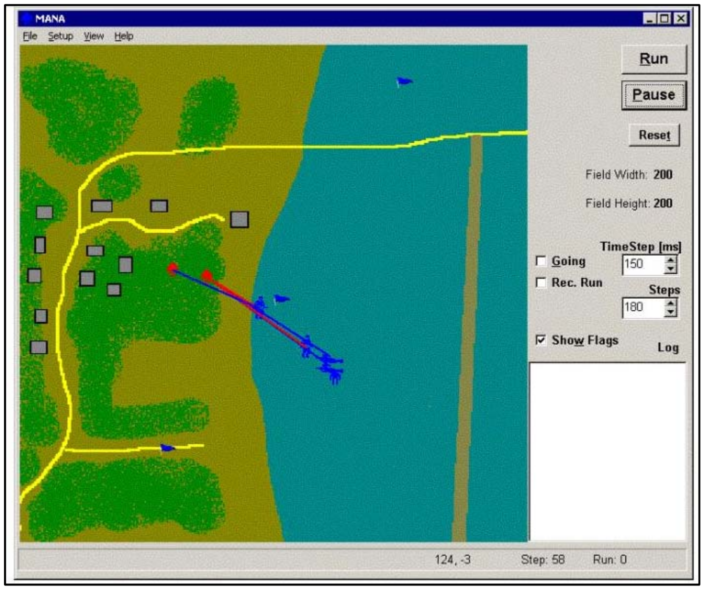

Figure I-1. Screenshot of the MANA GUI portraying a notional "ambush-at-dusk" scenario.
Note: In this situation, a MANA "squad" of agents was designed to represent Blue force riflemen in the fire teams and the machine gun sections. The fire teams have basic infantry values for the characteristics, while the machine gun teams have a larger "firing range" and "max targets per step." The Red force is made up of agents with basic infantry values, but with slightly less durability than counterpart Blue agents (*79*).

Being a precompiled executable, MANA runs relatively quickly so that many scenario simulation instances can be run through within a reasonable space of time. Furthermore, MANA has been designed with a built-in "data farming" capability, allowing a scenario's parameter space to be rapidly explored and analyzed. Additionally, a data-streaming capability was recently added so that MANA can now be used for human-in-the-loop experiments. An agent's sensor and weapon characteristics can be represented using either a simple distance-insensitive "cookie-cutter" approach, or with tables of range-dependent probabilities of sensor-target acquisition and weapon hit/kill. Furthermore, MANA was designed to simulate communications links for information sharing between groups of agents. For the purposes of modeling terrain features, MANA utilizes color-coded bitmaps. This reportedly provides the MANA user the advantage of quickly editing terrain features "on-the-fly," while a scenario is being developed. Finally, agent properties are specified for groups of agents defined to be "squads" (effectively adjustable to represent either platoons or companies). The linked concepts of an interagent communications network and a shared situational awareness (SA) map for squads, reportedly allows rudimentary aspects of network-centric warfare (NCW) to be modeled and analyzed using MANA.

In the context of simulation data analysis, MANA was designed with the capability to post-process results from multiple scenario runs into a number of time-dependent, multirun averages, which can then be graphed as time-series plots. In addition, Lauren (the principal MANA developer at DTA) has demonstrated a valuable frequency-domain analytic approach using MANA simulation data (*80*), as illustrated in figure I-2. Here, a Blue and Red force (each roughly the size of a moderate platoon) engage in a simple “movement-to-contact” scenario. In figure I-2a, the time of every casualty occurrence in either force over the course of 600 simulation runs has been recorded, yielding an aggregated time-series plot of the total number of casualties from all runs occurring at a given simulation time step (a net total of 250-agent casualties). In figure I-2b, the degree of temporal correlation across the aggregated casualty data is characterized by computing the power spectrum* of the time-series data presented in figure I-2a. In this second simulation data plot, a power-law structure exists on the left-hand side, while the right-hand side displays a flat, white noise spectrum. This composite frequency-domain data structure has been interpreted by Lauren and Stephen to indicate that disorder is highest on the smallest scales of combat, while the intermediate scales are neither completely ordered, nor disordered (*43*).

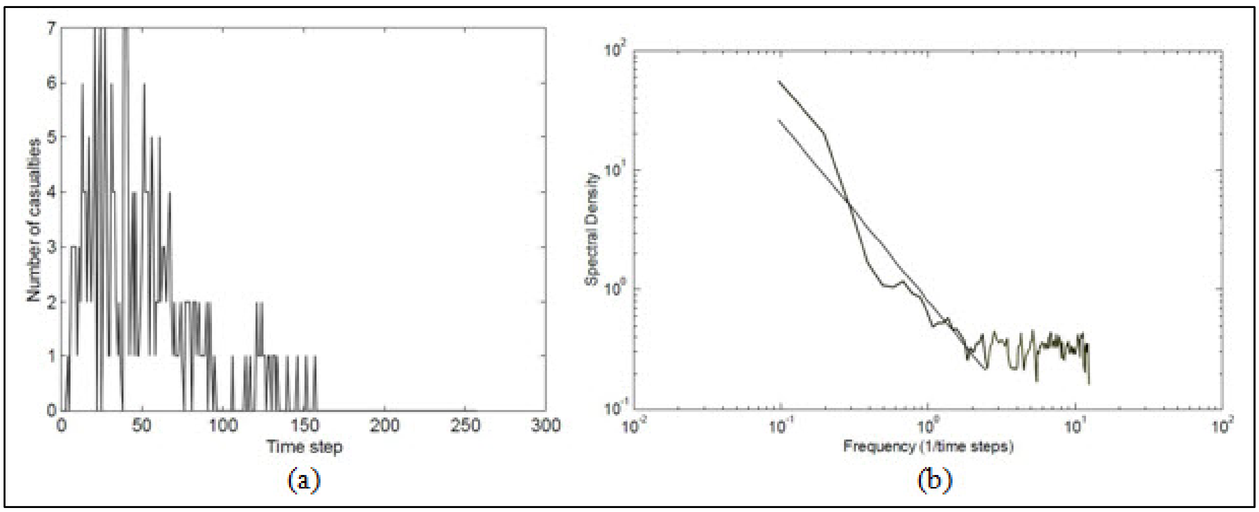

Figure I-2. Time- and frequency-domain analysis of casualties occurring with a MANA “movement-to-contact” scenario: (a) time-series plot of total aggregated number of casualties as a function of simulation time; (b) power-spectrum plot of aggregated casualty data presented in (a).

Although somewhat similar in design to the previously-discussed EINSTein (with a similar design inspiration originating from the spatially-discrete CA-based ISAAC model, see appendix H), the MANA agent-based model does boast a human-in-the-loop run-mode option similar to that demonstrated by PSOM (appendix G), plus a moderately-sophisticated approach to sensor and kinetic weapon modeling. Thus, we believe this agent-based simulation system

---

*For a given time-series data sequence of a measurable dynamic quantity $X(t)$, the *power spectrum* provides a plot of the portion of the measured quantity's power (i.e., energy per unit time) falling within given frequency bins (*81*). The most common way of generating a power spectrum is by using a discrete Fourier transform. Within the context of MANA simulation data analysis, the “power”—in this case—is the absolute value of the discrete Fourier transform of the $X(t)$ time-series data squared. This provides a means to characterize the degree of temporal correlation across the frequency of occurrence spectrum of different measured values of $X(t)$ in a MANA simulation.

provides a moderate capability to facilitate SoS modeling. As was the case with PSOM, running MANA as a human-in-the-loop/computer-based simulation could effectively emulate a military SoS at the strategic/operational level, where purposeful human decision makers could intelligently apply all aspects of nested concepts and inter-related purposes, and also demonstrate realistic sociotechnical system characteristics. When running at higher-tactical granularities, however, effective military SoS emulation by MANA is similar to that demonstrated by EINSTein (i.e., a *very* limited and weakly-coupled form of nested concepts and inter-related purposes combined with minimal potential for military sociotechnical system emulation). Also similar to EINSTein, is MANA's somewhat coarse-grained characteristic timescale, which limits the latter model's ability to properly emulate dynamic and evolving processes at typical military operations timescales.

On the other hand, the existing potential opportunities for effective and meaningful military SoSA utilizing MANA by itself are *very* limited. To justify this assessment, we point out the numerous military scenario modeling and analysis projects at NPS utilizing MANA (*79, 82–84*). In all cases, the MANA user(s) deployed either commercial-off-the-shelf (COTS) or internally developed software to facilitate simulation data analysis. As previously described, MANA itself does demonstrate a basic time-series analysis capability. However, the Power Spectrum analysis methodology proposed by Lauren and associates at DTA does promise a rather innovative approach to frequency-domain SoSA.

Consequently, table I-1 reports our estimates of the associated SoS and SoSA ontology attribute similarity scoring values for MANA (table I-1 [a] and [b], respectively). Given that MANA continues to be developed by various personnel at DTA (and utilized for projects and thesis research purposes at NPS), table I-2 reports on evidenced associated SoS and SoSA constituent component developmental maturation scoring values (table I-2 [a] and [b], respectively). In terms of software accessibility, the MANA *executable* is currently available only to U.S. institutions that have previously set up a formal agreement with DTA (such as NPS); although, it is reasonable to assume its conditional availability to other U.S. Government institutions. However, it is made clear in DTA-authored literature that MANA *source code* availability will remain exclusive to DTA developers (the latter whom are nevertheless eager to adaptively work with "official" MANA users external to the New Zealand government). As a result, we have assigned an accessibility status of *red* to this item. Finally, figure I-3 depicts the quadrant chart states of the SoS and associated SoSA designs associated with MANA (i.e., [1.88, 3.00, *red*] and [1.39, 3.00, *red*], respectively).

Table I-1. SoS and SoSA ontology attribute similarity scores associated with the Map Aware Nonuniform Automata (MANA) agent-based model.
Note: see section 3 for a discussion of the criteria, attributes, and weights.

(a) System of systems				
	Attribute			
	Description	Number	Weight	Score
Nested Concepts and Inter-Related Purposes →	Commander Intent	1	10	2
	Concept of Operations	2	10	2
	Purposeful Systems	3	7	2
Dynamic and Evolving Processes →	Synchronous Behaviors	4	10	2
	Multiple Time Scales	5	7	1
	Non-Stationary and Nonergodic Processes	6	5	2
	Sociotechnical Systems Theory	7	7	2
(b) System of systems analysis				
Analysis of Nested Concepts and Inter-Related Purposes →	Commander Intent	1	7	1
	Concept of Operations	2	7	1
	Synthetic Approach	3	10	1
Analysis of Dynamic and Evolving Processes →	State-Space Analysis	4	5	1
	Time-Series Analysis	5	10	2
	Frequency-Domain Analysis	6	5	3
Sociotechnical Systems Analysis →	Adaptive Multiscale Techniques	7	7	1

Table I-2. SoS and SoSA software component maturation scores associated with the Map Aware Nonuniform Automata (MANA) agent-based model

	Component			
	Description	Number	Weight	Score
(a) System of systems	Simulation Engine	1	10	3
	GUI	2	10	3
(b) System of systems analysis	Time-Series Analysis Software	1	10	3

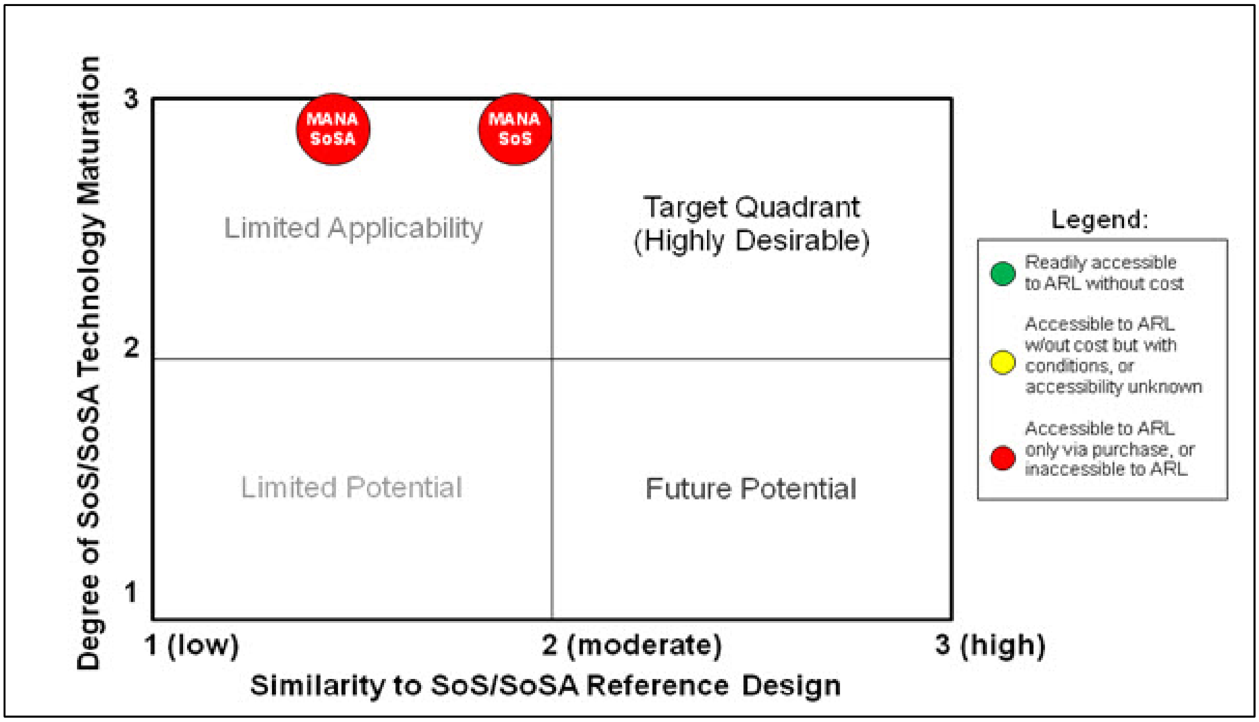

Figure I-3. Quadrant chart addressing SoS and associated SoSA designs for the Map Aware Nonuniform Automata (MANA) agent-based model.

Appendix J. Beijing National Defense University (BNDU) Agent-Based Model

As designed by Zhang et al., the *Beijing National Defense University (BNDU) agent-based model (ABM)* is a proposed new method of demonstrating System of Systems (SoS) weapon and combat equipment simulation based on the theory of complex adaptive systems (CAS), so as to meet the needs of realistic information warfare simulation and analysis (*47, in Chinese. Translated April 28, 2011*). The BNDU ABM appears to implement the CAS abstraction as defined by Holland (*86, 87*), wherein the primary characteristics of complex systems that adapt to dynamic environments can be described by means of emergent behavior and self-organization. As reportedly implemented by this model, the basic principle of ABM says that, by simulating the dynamical processes evolving in the real world, complex systems may be divided into corresponding *agents*, and by studying these components' microbehaviors, the macrobehavior of the whole system may be determined.

The BNDU ABM will model the behavior and interactions of a large number of military agents existing within an SoS in order to achieve its proposed SoS simulation goals. The model developers propose that the BNDU ABM is (or perhaps will be) classifiable as a *multi-agent system* (MAS), because (i) every different type of agent is encapsulated within a unique entity model; (ii) an appropriate multi-agent framework is used to integrate the entity-level agent models; and (iii) a credible SoS simulation can be instantiated via use of this integrated multi-agent model framework. Figure J-1 illustrates the SoS modeling and simulation process utilized by the BNDU ABM. This process is divided into five stages.

1. The *agent design analysis stage* addresses the functional requirements for the credible modeling of the various individual system types utilized within a military SoS. These types generally include one or more of the following system-level capabilities: sensing, weapons (kinetic and otherwise), mobility, and communications.

2. The *agent attribute and behavior modeling stage* addresses the software implementation of system-level capabilities and decision-making processes (DMPs) for all agent types populating the SoS. These capabilities are generally implemented via use of a standardized agent attribute framework, or template (e.g., an XML file). For agent-level DMPs, Bratman's Beliefs, Desires, and Intentions (BDI) abstraction (*88*) is utilized to determine the behavior (and subsequent actions) of every type of agent within the SoS.

3. The *agent interaction modeling stage* addresses the SoS mission design via the configuration of goal-directed entity-level interactions utilizing the BDI-agent framework. This stage reflects the SoS commander's vision and concept of operations relative to the simulated mission context.

4. The *agent encapsulation stage* addresses the process wherein the combination of individual agent DMPs and intentional multi-agent interactions generate a complete set of models, representing the capabilities and mission-oriented behaviors of every participating agent within the SoS simulation. Therefore, this stage generates a set of multiscale/multi-echelon DMPs consistent with the SoS commander's vision and concept of operations formulated in the previous stage.
5. Finally, the *MAS-based comprehensive integration phase* involves the synthesis of all agent models (including entity-level capabilities and entity-level/multiscale DMPs) into a collective multi-agent configuration. This last stage generates an overarching model ready to support SoS simulation.

Once all developmental stages of the SoS modeling and simulation process have been completed, the BNDU ABM is ready to support military SoS simulation-based analysis.

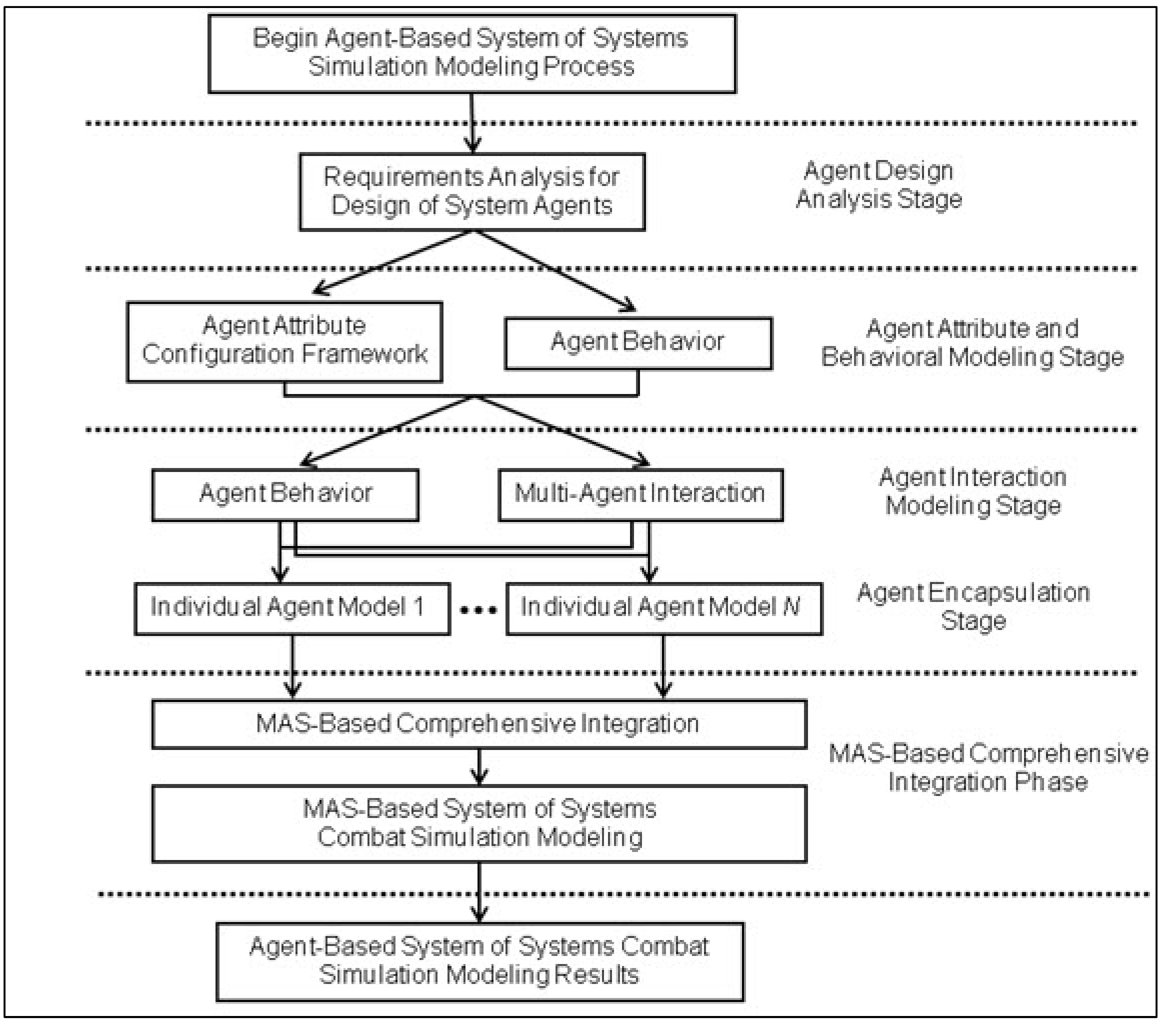

Figure J-1. The SoS combat simulation modeling process as utilized within the development of the BNDU ABM (*47, figure 2*).

As an example of what can be derived from application of this modeling process, figure J-2 depicts a notional military SoS agent framework. This example utilizes network-centric information age warfare concepts as guidance (e.g., separation of data collection units, weapon-equipped battle units, and command-and-control [C2] units), and seeks to demonstrate a high-level emergent SoS combat effectiveness as the simulation goal. The primary contents of this example SoS agent framework include information gathering systems modeling, C2 systems modeling, and battle unit weapon systems modeling. The utility of deploying such an agent framework for SoS simulation in the BNDU ABM is the ability to study and analyze the collective dynamic function and mission-oriented effects of the complete military system within an operational context. The BNDU ABM developers then claim that this facilitates the optimization of weapon system deployment and the assessment of collective SoS capability in order to determine policy for weapon systems research and development.

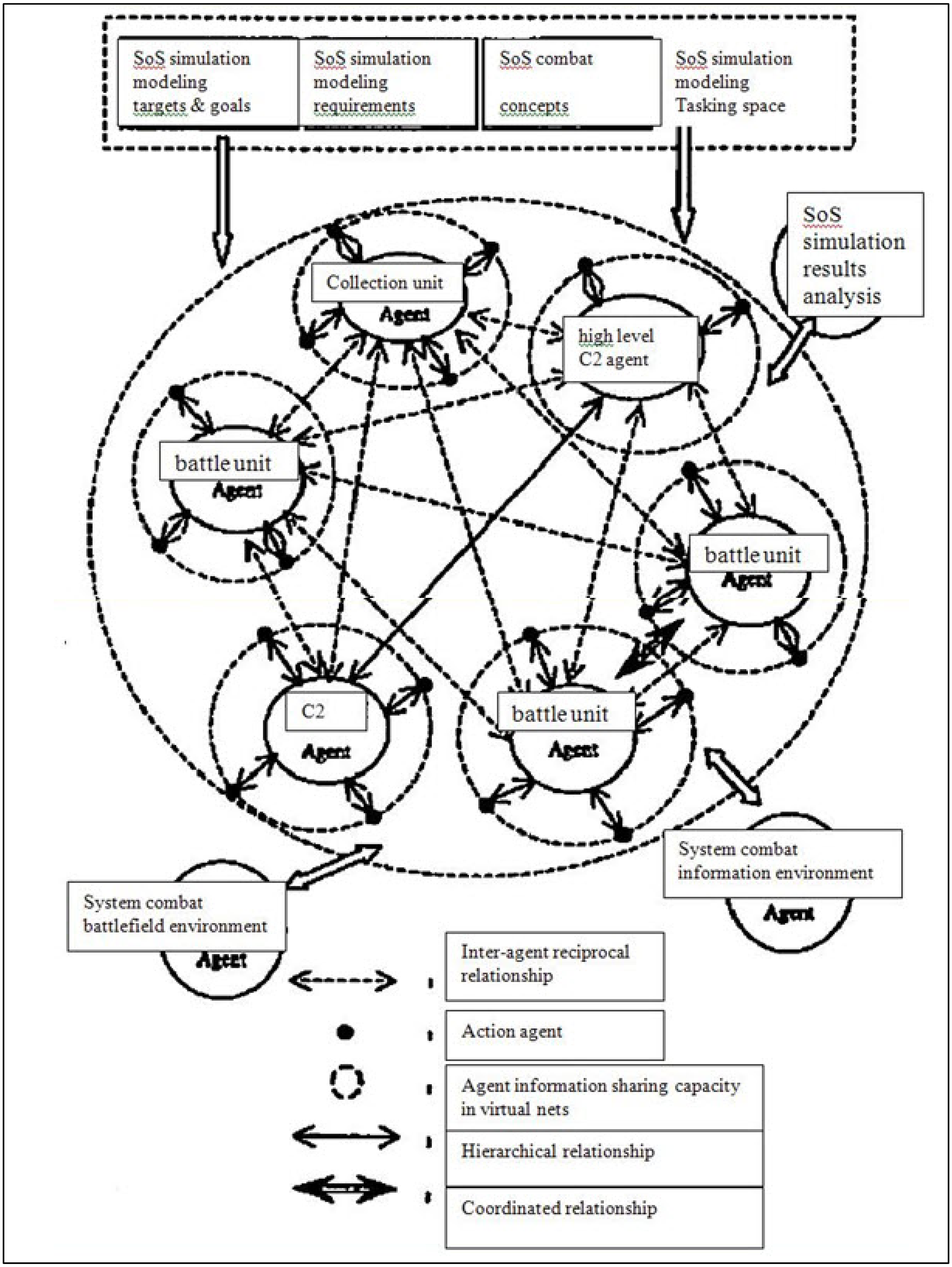

Figure J-2. Example of a conceptual framework for an agent-based SoS model (47, figure 3).

The implementation of the BNDU ABM is based upon a three-level object-oriented software composability structure inspired by the U.S. Army's High Level Architecture (HLA), Conceptual Models of the Mission Space (CMMS) and data standards (*89, in Chinese. Translated April 28, 2011*). This three-level software design structure is illustrated in figure J-3. In the foundational database level within the structure, the **c**ombat operations library contains software addressing simulation of basic physical combat actions (e.g., firing a weapon, moving over battlefield terrain), while the model service library provides a package of mathematical functions or problem-solving processes required for combat operation descriptions (e.g., system damage acquired from enemy kinetic weapon engagement).

Then, in the interim component level, a "service" component refers to a particular form of combat entity that encapsulates a specific action category from the combat operations library, while model database components encapsulate specific battle actions associated with specific combat entities. Finally, in the upper application level of the composability structure, the service oriented architecture (SOA) data bus applies component-level services as the core of the software engineering framework, wherein case-based reasoning is utilized to modulate service requests made by simulated combat entities.

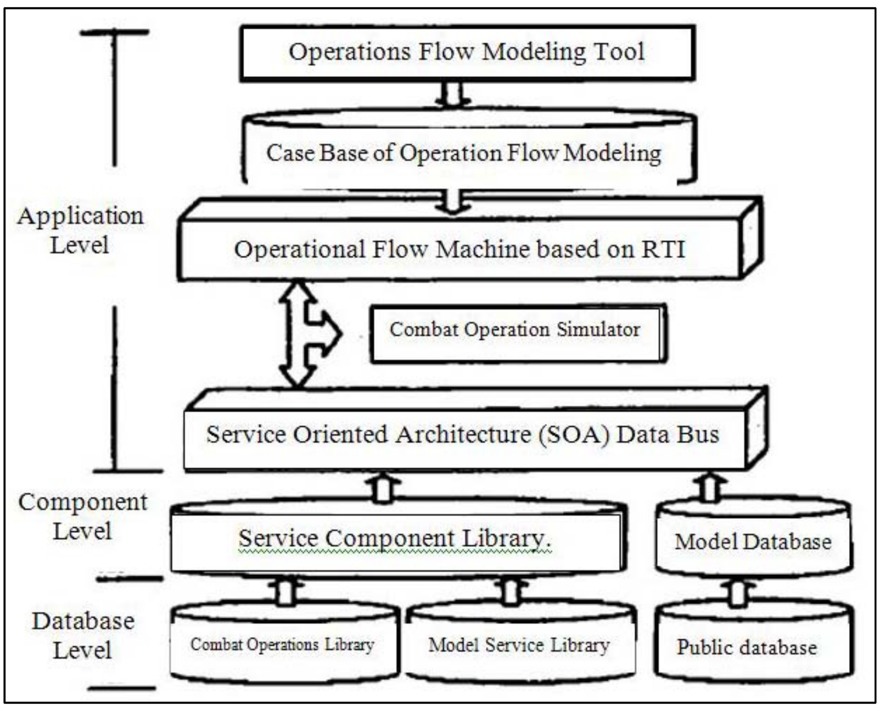

Figure J-3. The BNDU ABM three-level object-oriented software composability structure (89, figure 1).

Given that the primary design purpose of the BNDU ABM is the simulation of a military SoS within an operational context, we believe this agent-based simulation system *potentially* provides considerable capability to facilitate SoS modeling and simulation. The BNDU ABM design clearly and directly addresses the modeling of nested concepts and inter-related purposes via incorporation of commander intent and concept of operations,* and (by implication) some degree of dynamic and evolving processes. Also, the incorporation of sociotechnical systems concepts is implied by the detailed attention given to multiscalar design in the description of the BNDU ABM simulation framework. However, there is little-to-no evidence (in the referenced literature) indicating whether this model has in any way advanced beyond the initial design phase. Thus, detailed aspects and capabilities of the instantiated BNDU ABM software (if the latter even exists), cannot be explicitly evaluated for similarity to our target SoS ontology—except via implied intent on the part of the model developers. In addition, the BNDU ABM literature fails to address exactly how simulation results will be analyzed (implying that there is no clearly defined SoSA methodology associated with this model).

Thus, table J-1 reports our estimate of the associated SoS ontology attribute similarity scoring values for the BNDU ABM. Given the stated lack of evidence (in the evaluated literature) of continuing BNDU ABM software development, table J-2 describes reported SoS constituent component developmental maturation scoring values. Given that the BNDU ABM (whatever its actual developmental status) is a product associated with the defense department of the Chinese government, we have assigned an accessibility status of *red* to this item. Finally, figure J-4 depicts the quadrant chart states of the SoS design associated with the BNDU ABM (i.e., [2.23, 1.00, *red*]).

---

*However, it still remains unclear at this point whether agents within a BNDU ABM simulation are (or will be) capable of demonstrating or emulating purposeful behavior.

Table J-1. SoS and SoSA ontology attribute similarity scores associated with the BNDU agent-based model simulation system.
Note: see section 3 for a discussion of the criteria, attributes, and weights.

(a) System of systems				
	Attribute			
	Description	Number	Weight	Score
Nested Concepts and Inter-Related Purposes →	Commander Intent	1	10	3
	Concept of Operations	2	10	3
	Purposeful Systems	3	7	1
Dynamic and Evolving Processes →	Synchronous Behaviors	4	10	2
	Multiple Time Scales	5	7	2
	Nonstationary and Nonergodic Processes	6	5	2
	Sociotechnical Systems Theory	7	7	2
(b) System of systems analysis				
Analysis of Nested Concepts and Inter-Related Purposes →	Commander Intent	1	7	1
	Concept of Operations	2	7	1
	Synthetic Approach	3	10	1
Analysis of Dynamic and Evolving Processes →	State-Space Analysis	4	5	1
	Time-Series Analysis	5	10	2
	Frequency-Domain Analysis	6	5	3
Sociotechnical Systems Analysis →	Adaptive Multiscale Techniques	7	7	1

Table J-2. Software component maturation scores associated with the BNDU agent-based model simulation system.

	Component			
	Description	Number	Weight	Score
System of systems	Database level	1	10	3
	Component level	2	10	3
	Application level	3	10	3

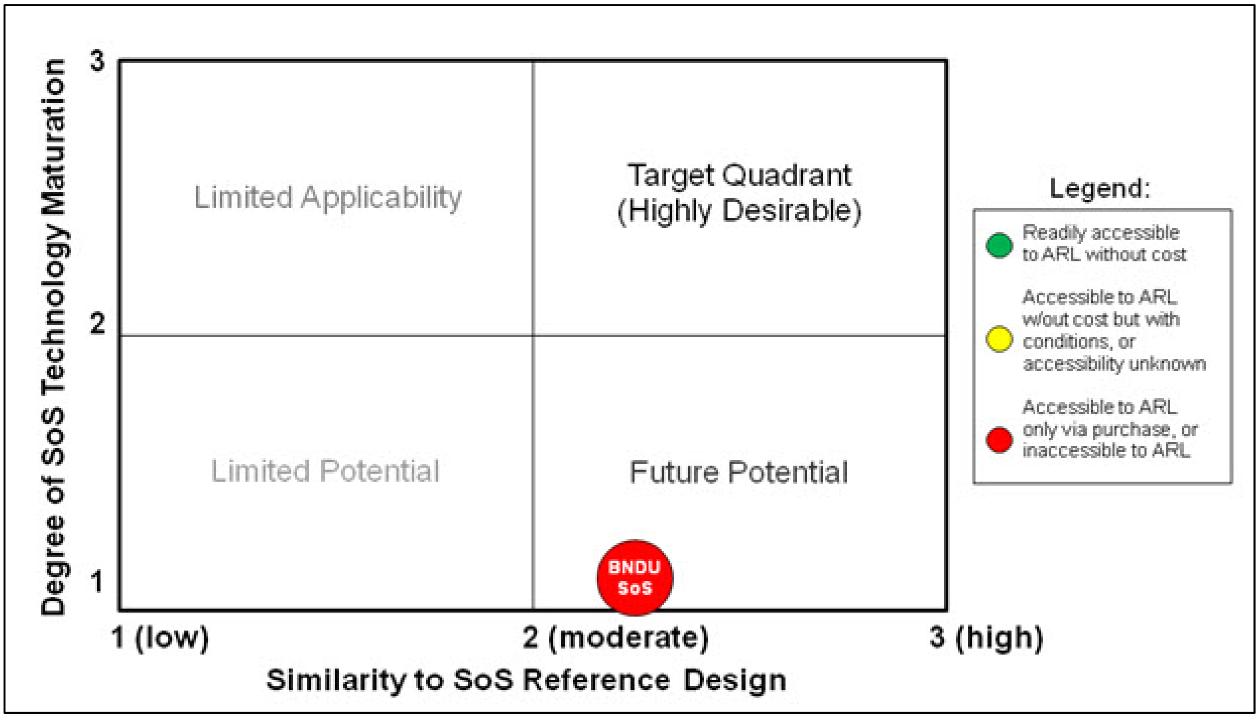

Figure J-4. Quadrant chart addressing SoS design for the BNDU agent-based model.

List of Symbols, Abbreviations, and Acronyms

ABIR	Agent-Based Identity Repertoire
ABM	agent-based model
ADA	Advanced Decision Architectures
AFEX	U.S. Air Force Exercise
ARL	U.S. Army Research Laboratory
ATEC	U.S. Army Test and Evaluation Command
BDI	Beliefs, Desires, and Intentions
BNDU	Beijing National Defense University
CA	Cellular Automata
CAS	Complex Adaptive System
CityDev	City Development Multi-Agent Simulation
CMMS	Conceptual Models of the Mission Space
COA	course(s) of action
COIN	counter-insurgency
C2	command-and-control
CTA	Collaborative Technology Alliance
DMP	decision-making process
DSTL	UK Defence Science and Technology Laboratory
DTA	Defence Technology Agency
EINSTein	Enhanced ISAAC Neural Simulation Toolkit
FSM	finite state machine
GA	genetic algorithm
GUI	Graphical User Interface
HLA	High Level Architecture

HN	host nation
HUMINT	Human Intelligence
IBCT	Infantry Brigade Combat Team
IO	Information Operations
IO	international organizations
ISAAC	Irreducible Semi-Autonomous Adaptive Combat
LAIR	Laboratory for Artificial Intelligence Research
MANA	Map Aware Nonuniform Automata
MAS	multi-agent system
MCCDC	U.S. Marine Corps Combat Development Command
MOD	Ministry of Defence
MOOTW	Military Operations Other Than War
MTPRPD	Measures of Tipping Points, Robustness, and Path Dependence
NATO	North Atlantic Treaty Organization
NCW	network-centric warfare
NGO	non-governmental organizations
NPS	Naval Postgraduate School
OG	Operational Game
OGD	other government departments
OneSAF	One Semi-Automated Forces
OOS	OneSAF Objective System
OTB	OneSAF Test Bed
PM	program manager
PS-I	Political Science-Identity
PSO	Peace Support Operations
PSOM	Peace Support Operations Model
PSYOP	Psychological Operations

SA	situational awareness
S-F-V	Seeker-Filter-Viewer
SIP	Strategic Interaction Process
SLAD	Survivability/Lethality and Analysis Directorate
SLV	survivability, lethality, and vulnerability
SLVA	survivability, lethality, and vulnerability analyses
SOA	Service Oriented Architecture
SoS	system of systems
SoSA	system of systems analysis
STOAT	Stabilization Operations Analysis Tool
TRADOC	U.S. Army Training and Doctrine Command
TTCP	The Technical Cooperation Program
UK	United Kingdom
Warcon	Wargame Construction Toolset

NO. OF COPIES	ORGANIZATION
1 (PDF)	DEFENSE TECHNICAL INFORMATION CTR DTIC OCA
2 (PDFS)	DIRECTOR US ARMY RESEARCH LAB RDRL CIO LL RDRL IMAL HRA RECORDS MGMT
1 (PDF)	GOVT PRINTG OFC A MALHOTRA
1 (PDF)	USARL RDRL SLE R FLORES
2 (PDFS)	RDRL SLE G J SMITH RDRL SLE M P SIMPSON
1 (PDF)	PHYSICAL SCIENCE LABORATORY NEW MEXICO STATE UNIVERSITY R BERNSTEIN JR

ABERDEEN PROVING GROUND

1 (PDF)	DIR US ARMY EVALUATION CTR HQ TEAE SV P A THOMPSON 2202 ABERDEEN BLVD 2ND FL APG MD 21005-5001
5 (PDFS)	DIR USARL RDRL SL J BEILFUSS M STARKS P TANENBAUM RDRL SLB A M PERRY RDRL SLE M B RUTH