



What is a good pattern of life (PoL) model? – Guidance for simulations

Journal:	<i>Simulation: Transactions of the Society for Modeling and Simulation International</i>
Manuscript ID	Draft
Manuscript Type:	1 Applications
Date Submitted by the Author:	n/a
Complete List of Authors:	Silverman, Barry; University of Pennsylvania, Collaboratory for Advancement of Systems Approach (ACASA)
Key Words and Phrases:	Tools and Technology -> Agent-based Systems , Tools and Technology -> Human Behavior Representation , Tools and Technology -> Simulation System Architecture , Tools and Technology -> Virtual Reality/Environments , Tools and Technology -> Training Simulators
Abstract:	<p>We have been modeling an ever-increasing scale of applications with agents that simulate the Pattern of Life (PoL) and real-world human behaviors in diverse regions of the world. The goal is to support socio-cultural training and analysis. To measure progress, in Section 1 we propose the definition of a measure of goodness for such simulated agents, and review the issues and challenges associated with the first generation (1G) agents. In Section 2, we present a second generation (2G) agent hybrid approach that seeks to improve realism in terms of emergent daily activities, social awareness, and micro-decision making in simulations. Section 3 offers a PoL case study with a mix of 1G and 2G approaches that was able to replace the pucksters and avatar operators needed in largescale immersion exercises. The final section concludes by observing that a 1G PoL simulation may still be best where large-scale, pre-scripted training scenarios will suffice, while the 2G approach will be important for analysis or if it is vital to learn about adaptive opponents and/or unexpected/emergent effects of actions. Lessons are shared about ways to blend 1G and 2G approaches to get the best of each.</p>

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

SCHOLARONE™
Manuscripts

For Peer Review

What is a good pattern of life (PoL) model? – Guidance for simulations

Barry G. Silverman, Gnana Bharathy, Nathan Weyer, Qiwei Sun¹

Abstract

We have been modeling an ever-increasing scale of applications with agents that simulate the Pattern of Life (PoL) and real-world human behaviors in diverse regions of the world. The goal is to support socio-cultural training and analysis. To measure progress, in Section 1 we propose the definition of a measure of goodness for such simulated agents, and review the issues and challenges associated with the first generation (1G) agents. In Section 2, we present a second generation (2G) agent hybrid approach that seeks to improve realism in terms of emergent daily activities, social awareness, and micro-decision making in simulations. Section 3 offers a PoL case study with a mix of 1G and 2G approaches that was able to replace the pucksters and avatar operators needed in largescale immersion exercises. The final section concludes by observing that a 1G PoL simulation may still be best where large-scale, pre-scripted training scenarios will suffice, while the 2G approach will be important for analysis or if it is vital to learn about adaptive opponents and/or unexpected/emergent effects of actions. Lessons are shared about ways to blend 1G and 2G approaches to get the best of each.

Keywords:

Pattern of life, cultural simulation, agent-based models, adaptive behavior

1. Introduction and purpose

We are experimenting on a society simulation capability that is realistic enough to be useful for emulating the pattern of life (PoL) in different environments. Such a capability could be useful either for analysis of and/or for training soldiers/police/responders in a variety of community stabilization topics such as, among others: counter-insurgency, instability reduction, self-sustainment, provincial reconstruction, disaster relief, etc. While the relevant social science theory is incomplete, and the PoL training needs are not fully mapped (e.g., see [1]), we have nevertheless been tasked to construct a best practice set of tools that we describe here. With PoL simulations, artificial intelligence and human behavior modeling (HBM) research must concern itself with mind-body issues, social concerns, and reasoning about daily activities as peoples would in various cultures around the world. The more realistically the simulated agents reflect the real world, the greater will be the users' sense of place and immersion into the scenario of interest.

In the realm of national defense, there are many potential reasons to use a PoL simulation from both the training and analysis viewpoints. A commonly cited purpose is (a) to teach soldiers how to recognize and practice operating under archetypical vs anomalous patterns of behavior. One of many possible examples is a market that is too empty during business hours and thereby

¹ Electrical and Systems Engineering Department, University of Pennsylvania, Philadelphia, PA 19104
Barry G. Silverman (basil@seas.upenn.edu)

Corresponding Author:

might presage an IED explosion. This type of thinking is the goal of a program known as Combat Hunter [2]. Another very common PoL training goal is (b) how to learn local norms, go forth as “ambassadors”, and build relationships with the populace in a local culture so as to figure out what are the key issues promoting or preventing security, stabilization, transition, and reconstruction [e.g., see 3]. Yet another is (c) how to use a brigade/battalion to detect, deter, and defeat a network of (e.g., 150 insurgents) hiding amongst a region of 100,000s of civilians carrying out their PoL: e.g., see [4]. Many more training/analysis goals might exist for PoL simulations. A few other obvious military/diplomatic ones are (d) Mission Rehearsal (for example, a remotely targeted drone strike or in-person arrest) in a heavily populated area, or (e) analysis of the potential effects of a Secure, Stabilize, Transition and Recover (SSTR) campaign on the populace of a region or nation [3]. As a result, to the extent possible, PoL simulations need to be flexible, reusable, and extendable for new needs as they are enumerated.

The current practice in simulation exercises intended to help answer all these questions is largely based on human-played avatars (pucksters) who carry out the daily lives and dialogs. But this is expensive and time-consuming. Alternatively, a growing number of war games and public health/disaster rehearsal sims include hand-scripted finite state machines [5]. These are 1st generation (1G) agents that try to achieve realism via the Markovian approach, keeping agents simplistic and rigid, though computationally efficient and, hence, highly scalable in terms of numbers of agents. When simulation scenarios unfold according to plan, these PoL agents can be a very effective way to enhance the player’s immersion and sense of place.

However, simulation exercises often lead to unexpected situations that the 1G agents were never pre-programmed to handle. Since their goal is to look smart or realistic long enough to escape the player’s critical eye, they only have a small amount of problem solving or re-planning of goals under limited circumstances. The problem is that once simulation begins, the original assumptions behind the pre-composed behaviors might no longer hold, and the 1G agents will not have any way to reason about a new situation. Other common complaints about these 1G simulation entities and game-world agents (hand-scripted, finite state machine agents) are that they are shallow, narrow and brittle, have no relationship to others, and are unaware of the greater world. Also, they have no larger purpose and no ongoing daily life or aspirations.

In attempts to build more realistic agents, one prominent school of thought is focused on building more complex and inclusive finite state machines (Markov chains) that may eventually be able to respond to every possible scenario that arises in a simulation. While such efforts do often improve the behaviors of the 1G agents in the short-run (as it provides more variety of “life-like” actions for viewing), the Achilles’ heel persists: the instance space (in which each sequence of acts, involving actor agent, target, form of action and possible social relationships may serve as an instance) is so enormous, the task of *specifying the range of good instances* is intractable. Furthermore, as the actual subject of study (real-life human decision making process) is in itself not a probabilistic finite state machine (though it may often behave like one), attempts to model pattern of life with such an approach would at best lead to a simulated agent under-prepared for rare-case scenarios. In brief, while 1G agents are the most efficient, scalable approach in our toolbox, there are times when we need an alternate approach that is similar to real-life agents, rather than just a black-box surrogate.

1.1. What is a good PoL?

Ideally, we wish for 2nd generation (2G) agents that exhibit “realism” in three areas: their exhibited activities, network relations, and situated behaviors. We define these three areas further below. If all three areas can be deemed reasonable in the social environment in response to *a priori* information or events as compared to real-life human behavior, we may consider the agents to be *PoL good*. Specifically, to measure POL Goodness, we propose a scoring schema as shown in Table 1 and as described more fully below. This shows the tripartite as the rows, and three levels of capability as the columns – low, medium, and high. We give definitions in each cell. To score an agent for POL Goodness, there is 1 point per cell. A 1G agent could have only Low capability in each row, or possibly 1 or 2 of the Medium column entries. Thus any score of 3 to 4 or less might be PoL Limited. A 2G agent could have capability in all cells, though that is unlikely. It is also possible that someone working on a 2G agent ignores some of the low and/or medium capabilities, so they would still have a PoL Limited score. Anything above a score of 6 or greater is what we will call a PoL Good agent.

Table 1 – PoL Goodness is a Count of the Number of Cells that Describe a Given Agent

	Low	Medium	High
Activities of Daily Life (ADLs)	*Prescribed ADLs following clock *Navigates on own to destinations, avoiding obstacles.	Simple rules to handle a few common ADL issues in 1 area (eg, errands, OR combat, OR crowds)	Dynamically (re)sets priorities and (re)plans ADLs due to shifts in internal needs and external events
Nets/Social Skills	Reacts to nearest neighbors on a landscape	Connects to agents across a single-layer network	Has relationships and connections across multi-layer nets
Cognition	Can express values & reactions passed into it from a “god” source	Limited rules to react to a domain (single-layer net)	Adaptively appraises world against its own values.

To address PoL-goodness in general, researchers have fielded a growing number of cases of 2G agent-based models. However, achieving PoL-goodness is challenging for 2G agents as well. As pointed out in Van Hemel et al.[6] and Numrich and Picucci [7], the 2G approaches typically achieve adaptivity in one dimension and often rely on one or a few narrowly focused models drawn from a given discipline such as social nets, cognitive models, or system dynamics. The 2G agents often suffer from inability to integrate across disciplines and thus examples abound of problems: e.g., cognitive agents are unaware when you kill another agent standing nearby, or social agents cannot independently make decisions and form cognitive appraisals of the unexpected or nonlinear effects in the world at large. Van Hemel et al.[6], Numrich and Picucci [7] and Folsom-Kovarick et al. [8] all point out the problems of narrow disciplines leading to narrowly focused (but deep) models and the need to move beyond this.

To address these concerns, the PoL-good idea requires agents to encompass three sets of capabilities. We focus in this research on possible improvements along these dimensions.

1.1.1 Daily Life Activity Issues. The agents must be able to generate the normally expected pattern of daily life within their culture and demographic conditions. A cutting edge of PoL research is to automate the development of these PoL behaviors for a random urban environment, thereby reducing the human development time required: e.g., see [9-11], among

others. These agents are set up with day timers so that markets are busy at the right time of the day, commuters appear at rush hours, and streets get deserted when locals suspect an IED is about to be detonated. This is useful for reflecting normal vs. anomalous patterns, but it is superficial and PoL workflows are imposed on the agents rather than emerging from them. Such 1G PoL agents are easy to generate since they have no larger purpose and no ongoing daily life or aspirations. They are there to give the appearance that various scenes are populated appropriately and so collateral damage statistics will be accurately reflected if a mission is attempted. In some PoL applications, a market or street may look busy, but if you follow the agents around the corner they are going nowhere, standing still, or just waiting before they reenter the crowd again. The few cases where they do live out daily life with some sense of purpose is in commercial entertainment games (e.g., The Sims) where the agents exist in make-believe worlds. We suspect this lack of social life is because they are built for cluttering up worlds rather than for recreating societies. In fact, an early PoL app for recreating urban traffic was actually named ClutterSim, an unfortunate moniker that suggests how thoroughly developers under-estimate and objectify agent populations [12]. Agents that are just objects moving according to an external clock and workflow will fail the PoL-good test as soon as they have to interact with users or with unexpected situations. One often needs to use some degree of 2G agency to ease this dilemma (ie, Medium or High on row 1 of Table 1).

1.1.2 Network/Social Science Issues. A second dimension of PoL-goodness is for the agents to be social, to manage relations, and to interact in groups and across networks. This dimension tends to be ignored by 1G agents. Also, it's challenging for many types of 2G agents as well. As mentioned in [6, 7], 2G agents tend to specialize. They are either good at this criterion (social) or the next one (cognition), but not both. Also, where sociologic models involve the concept of networks, they usually consider a single-layered network, yet PoL simulations often require the modeling of multi-layer networks since that is how people live in reality (e.g., kinship links, economic activity, political/ego nets, and religious connections, to mention a few). Current game agents in general are poor at relationship management, have no sense of ongoing allegiance or relationship buildup, and poor understanding of cultural transgression and atonement dynamics. Fortunately, artificial society simulation has advanced to the point where complex social systems are modeled bottom up. Micro-decision making of individual agents who may influence each other is combined until macro-behaviors of the larger population emerge: e.g., see [13]. It is vital to bring such approaches to bear in PoL simulations, although to date doing that across multi-layer nets is a rarity. It is usually limited to single layer or just local neighborhood relations.

1.1.3 Situational Reactions and Cognitive Issues. Finally, it is important for agents to be situationally aware; to have the capability to reason about potential disruptions to their routines and networks; and to adapt behaviors in meaningful ways. The goal is that autonomous agents would populate a society where their micro-decision making would lead to emergent macro-level behaviors relevant to their cultural norms and value systems. This is bottom up, emergent reasoning. One way to achieve this emergence and adaptivity is to shift the paradigm so the PoL behavior and workflows emerge from the agents, rather than being inputs to drive the agents. Instead of relying on 1G scripting patterns and workflows as needed, we are interested in culling best-practice theories from the behavior literatures (e.g., psychology, sociology, political science, economics, etc.) and implementing these as parameters, metrics, and meta-models to drive the PoL agents in the virtual environment. We describe our algorithm for this in Section 2.6.

Unfortunately, many of the cognitive models used in psychology are complicated (they cannot easily scale up due to computational inefficiency) and besides are often designed for narrow or micro-cognitive foci. Those that are potentially useful for autonomous agent reasoning about ADL tasks are rarely oriented to social issues such as realizing their neighbor has been shot or that they are a member of a community that observes cultural norms and standards or that exhibits differing cultural styles of thought. While there are 2G agent engines that may satisfy this cognition sub-criterion, there are few off-the-shelf solutions that satisfy this and the other criteria.

1.2. Advances sought for a meso-level approach to agent modeling

Our goal is to improve the PoL-goodness of agents in game-worlds by adding greater depth *and* breadth via federated capabilities. Most 2G agents tend to be too deep and narrow to accomplish this, so we are interested in a hybrid meso-level, 2G capability that allows us to include aspects from many disciplines at once. As a result, we do not want a single model of behavior, but the ability to synthesize or plug together a range of relevant models. The goal is thus to federate a rich base of models (in a standards-compliant, plugin-plugout way) to replace humans having to drive the avatars and to allow the agent sims to be able to mimic social dynamics and faithfully express real-world stakeholder issues through action choices and conversational interactions.

Section 2 explains our Meso-HBM hybrid approach and how it plugs in models of diverse social science theories across disciplines. We have made the agents cognitively rich (broad and deep), socially connected on many levels, and conversational about their world and allegiances. This approach relies on a socio-cognitive agent technology known as PMFserv, along with the StateSim engine that models larger regions, higher echelons of society, 3rd world economies, and the distribution of public goods and services. On the technology side, our goal is to explore how social science theories and models could be used to drive agent behaviors so that autonomous socio-cognitive agents populate a society. Their micro-decision making leads to emergent macro-behaviors relevant to their cultural norms, value systems, and collective perceptions. We make no claims that this is the best modeling approach. We are just trying to assess if we get further than with other approaches, a question we empirically explore in the case studies.

1.3. Addressing the scale vs. behavior divide

As Folsom-Kovarik et al. [8] point out, the PoL field reflects a divide between applications of 1G agents that scale well and those with 2G agents with more adaptive behaviors. There are ever larger scale 1G agent worlds, such as, for instance, Ge et al [22] and Zhang et al [23] who study epidemics with the help of an artificial city of 19.6 million agents that each carry out 10 life activities per day. However, the agents are just pre-scripted activity patterns that are a useful device for studying spread of disease. There are no large scale demonstrations of 2G agents due to computational constraints. Using a 2G approach known as Beliefs, Desires, and Intentions (BDI) agents, Cho et al. [18] found that runtime slowed in direct proportion to the number of BDI agents in a simulated crowd. Hindriks et al [19] embedded them as gamebots in Unreal and found they could not scale beyond a few BDI agents without loss of performance. In 2016,

Adams & Gaudou [20] did an extensive survey of BDI agents in social simulations and show there are no large scale implementations of BDI agents to date.

The Folsom-Kovarik et al [8] recommended solution for reaching scale with higher fidelity behavior is to model mostly 1G agents and switch in 2G agents only when needed for “vital interactions”. The same is said in Wolfe et al [21] about judiciously adding BDI agents. While neither specifies how this might work in detail, one can imagine several use cases that might implement this idea, such as, among others: (1) using light or 1G agents in the background and heavy or 2G agents in the foreground of whatever user perspective exists; (2) using 1G agents and swapping in 2G smarts whenever a user tries to interact with a given agent; or (3) creating an agent hierarchy where 2G group leader agents have all the important decisions and also determine the behaviors of their large number of 1G followers. Each of these ideas (and others) has benefits and costs, and each needs to be tested if research is to make progress.

In our research we have investigated two use cases to date. The first one scales our 2G agents to the size of a village of about 100 agents. We reported on that use case earlier: see [14]. The second case study, reported here, implements idea (3) above and presents lessons learned on that approach to coping with the “divide”. Specifically, these two cases allow us to respectively investigate the following:

- **R1:** Can meso-level human behavior models improve the realism (PoL-goodness) of small PoL simulations?
- **R2:** Will the PoL-goodness of large-scale simulations improve due to a hierarchy of 2G leader agents in charge of the 1G followers?

Section 3 presents the second case study to explore these research questions and to see if the HBM approach is scalable, reusable, and able to make improvements relative to the PoL-good issues. We did not have funding to conduct formal experiments with controls. Hence these are research questions that we evaluated by obtaining reactions from users to what was built. Section 4 summarizes lessons learned.

2. Background on meso-HBMs: StateSim

For exploring Meso-HBM approaches, we have developed a model of models framework called StateSim. StateSim is a model of a state (or cross-state or sub-state) region and the important political groups, their ethnic (and other) conflicts, economic and security conditions, political processes, domestic practices and external influences. It is also the framework we use to assemble our 2G agents and to provide them with a social context in which they can observe and interact with other agents, the landscape, and institutions in their world. By constructing our 2G cognitive agents within StateSim, we directly address the social side of the tripartite. That is, StateSim forces the agents to confront many social life contexts (economic, political, religious, etc.) and to choose actions that influence their networks in each of these contexts. These capabilities permit StateSim agents to be used either analytically or for immersive training as we now explain.

In the analytical usage mode, StateSim agents are unconnected from physical terrain, bodies, and related artifacts. In this manner, they can run much faster-than-realtime and can be used for predicting how micro-decision making and events can lead to emergence of macro-behaviors and equilibria shifts. These portions of StateSim, described in Sections 2.1-2.3 (though also in the rest of Section 2), were built under three programs sponsored by Defense Advanced Research

Projects Agency (DARPA). In analytical mode, DARPA tested it against 100s of quarterly Events of Interest (EOI) such as coups, rebellions, and repression for each quarter from 2006 – 2008 for each of several Pacific rim nations (Bangladesh, Korea, Sri Lanka, Thailand, and Vietnam). DARPA indicated it could predict EOIs with >80% accuracy, recall 88%, and precision 64%. Aside from the DARPA backcasts (also called retrospective forecasts), in dozens of other correspondence trials, StateSim has demonstrated better than 80% correlation with the action choices of leaders and followers (insurgencies, rebellions, political repressions, and inter-group violence) for numerous countries in Asia, Africa, and the MidEast. See Bharathy and Silverman [15].

In immersive training mode, StateSim is slowed to realtime, reconnected to agent bodies, and aligned with artifacts and organizations that appear on physical landscapes. This allows StateSim to be run in the PoL style of usage that was described earlier. We typically do this by embedding StateSim behind a 3rd party game engine to drive the micro-decision making of the 3rd party's 1G agents and to give those agents the missing social context and networks for their actions. Sections 2.4 – 2.6 describe how this works (though the rest of Section 2 does also). A more complete case study of such an application of StateSim is presented in [14] which won 1st place for *Best AI/Pattern of Life* in the Federal Virtual World Competition². Finally, Section 3.2 attempts an experiment where StateSim is used in both analytical and immersive modes simultaneously. There, 2G leader agents run unembodied behind the scenes (analytically) while 1G follower agents run PoL on screen.

2.1. Profiling factions

StateSim is a model of models that facilitates the codification of alternative theories of factional interaction and the evaluation of policy alternatives. StateSim is a tool where you set up a conflict scenario in which the factional leader and follower agents all run autonomously. You are the sole human interacting and using a set of Diplomatic, Informational, Military, and Economic (DIME) actions to influence outcomes and Political, Military, Economic, Social, Information, and Infrastructure (PMESII) effects (see Figure 1). Factions are modeled as in the center of Figure 1 where each has a leader, various sub-faction leaders or “henchmen” (e.g., core, fringe, other), a set of starting resources (Economy, E, Security, S, and Politics, P), and a representative setup of 1,000s of follower agents. A leader is assumed to manage his faction's E- and S- storage tanks so as to appeal to his followers and to each of the other tribes or factions it wants in its alliance. Each of the leaders of those factions, however, will similarly manage their own E and S assets in trying to keep their sub-factions and memberships happy. Followers determine the level of the P-tank by voting their membership level (see Section 2.2). A high P-tank means that there are more members to recruit for security missions and/or to train and deploy in economic ventures. So leaders often find it difficult to move to alignments and positions that are very far from the motivations of their memberships.

StateSim runs a set of multiple games simultaneously. Within a faction one may observe games between rival leaders, between leaders and followers, and follower on follower. The across-faction games include attempts to cooperate and/or compete with other factions' leaders and followers, and/or attempts to contain factions aimed at your own downfall. For discussion's

² NonKin Village received this prize for *Best AI/Pattern of Life* in the 2011 Federal Virtual World Competition. More about NonKin is at: <http://www.acasa.upenn.edu/nonKin/nonkin-description.htm>

sake, consider these as iterated semi-cooperative games. This game formulation is the simplest game one can analyze involving conflicts between (and within) factions. Using it helps to clarify many of the key elements of these conflicts.

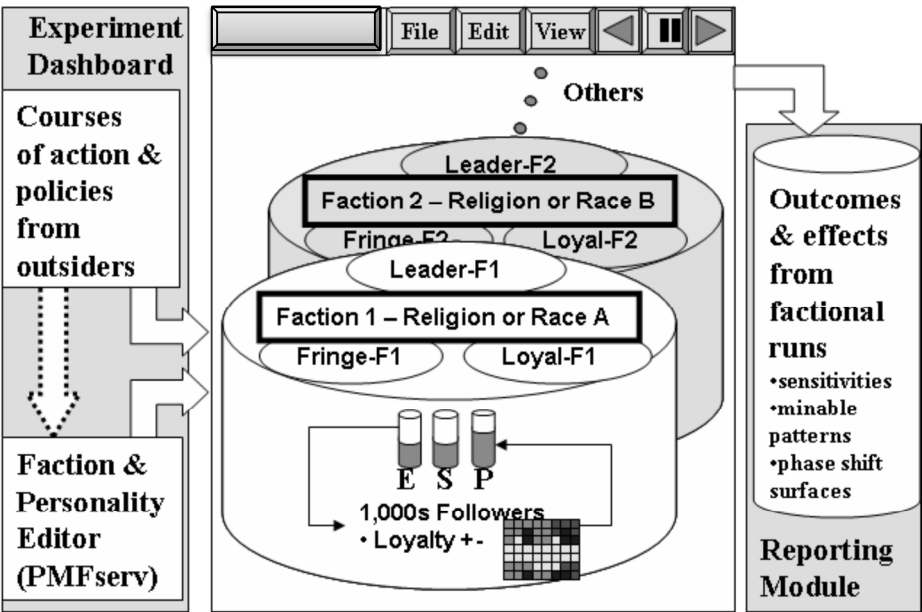


Figure 1. Models and Components Used in StateSim Scenarios
(Legend: E=economic assets, S=security assets, and P=political influence level)

2.2. Personality profiling models

One of StateSim’s strengths is that its factional leader and follower agents are cognitively detailed. The idea is to have 2G agents that utilize micro-decision making based on the profiles or tendencies of the actual people of the region being modeled. Profiling of personalities has not yet reached the stage of a mature science with first principles; however, there are best-of-breed profiling instruments with respectable field trials and high inter-rater reliability. These are useful for creating agent frameworks with greater degrees of realism.

Profiling Leaders – In StateSim, each leader and follower is modeled within a framework known as PMFserv [14, 16]. This adds a mind-body library of models to StateSim. For immersive applications, it allows the agents to have a physiology with some detail, while in larger scale sims, we often turn the body models off. One of the StateSim models allows leaders’ (and followers’) cultural values and personality traits to be profiled. These profiles are captured in Goal, Standards and Preference (GSP) trees [14]. These are multi-attribute value structures where each tree node is weighted with Bayesian probabilities or importance weights. A Preference Tree is one’s long term desires for world situations and relations (e.g., no weapons of mass destruction, stop global warming, etc.) that may or may not be achieved in the scope of a scenario. In StateSim agents the preference tree translates into a weighted hierarchy of territories and constituencies (e.g., no tokens of leader X in resource Y of territory Z). The Standards Tree defines the methods a leader is willing to take to attain his/her preferences, and what code that others should live by as well. We also add specific standards that capture the doctrine a leader adheres to when considering his *Economic and Security* tanks. Finally, the Goal Tree holds short-term needs the agent seeks to satisfy each turn (e.g., vulnerability avoidance, power, rest,

etc.). The GSP tree is a value model editor that allows one to (a) implement leader and follower profile instruments as nodes on the trees and (b) set the weights on the nodes which in turn implements a personality profile (an example GSP tree is shown in Section 2.6).

GSP trees are used by the agent for all decisions - e.g., selecting a next game action, determining faction alliance moves, or deciding on a speech act - thereby giving each agent a robust and individual worldview. When contemplating a decision, the agent calculates the subjective expected utility (SEU) it expects to derive from every action available to it, as constrained by perceptions, and chooses the alternative that maximizes SEU. Probabilities assess likelihood of success.

Profiling Followers – In addition to Leaders, StateSim factions also have followers who autonomously compute their strength of membership in each group, affinity for leaders, and resulting motivations toward action. This works similarly to how leaders reason and due to space limits, we will refer the reader to [14-16] for full details. We will simply state here that the followers compute their grievance state (GS) for each faction, ranging from -4 to +4 (the choice of range bears no significant meaning, as we simply normalize all the values in the computation process) and use this to determine their actions. As an example, suppose an agent identifies himself with Faction B, but lives under the rule of Faction A. The top state (GS+4) is total support of a given Group, say A. A faction getting a mid-point grievance scale (GS-0) means that agent is undecided and/or helpless to resist what A wants. At the other extreme of GS-4, the Faction B agent who lives under Faction or Leader A has already joined a resistance faction C working against A. At the extremes on either end, the agent will submit to militaristic commands of the leader of that group, while at the next lower level it will be only willing to go to protests, and verbally and economically support the activities of that group's leaders.

2.3. *The economy and institutional agencies*

Each faction or group's economic production depends on their constituency size, capital, education, health, employment level, legal protections, access to basic resources (water, etc.), and level of government repression. In the balance of this section we examine how the institutions of a single faction work and may be influenced. The write-up focuses on public institutions to keep it brief, but we also model private ones and business enterprises that the actors may manage, work at, get goods and services from, and so on. One can substitute more detailed, third party models of these institutions and enterprises without affecting the ability of our cognitive agents to interact with them. Thus the models discussed in this section are defaults and one can swap in other models without affecting how the actors think through their resource-based, ethno-cultural conflicts.

An institution's primary function is to convert funding into services for groups. Groups in turn, provide service to members. Groups, including the government, provide funding and infrastructure usage rights. In turn, each group has a level of influence over the institution which it leverages to change the service distribution to its own group and region. Influence can be used to increase favoritism (for one's own group, for example) but it can also be used to attempt to promote fairness. Institutions also are endowed with a certain level of efficiency. Efficiency is considered the fraction of each dollar that is applied to service output, as opposed to lost in administration or misuse.

The institutions currently modeled are public works, health, education, legal protections, and elections. Also, competing groups may set up competing institutional models. In this case, this

gives greater influence and control to the leaders of those groups over the institutions and their public good allocations.

2.4. Regional resource dynamics and “human terrain” reasoning

Because conflicts are invariably linked to control of resources (broadly defined), a total systems perspective dictates a model of resources arrayed in their respective territories under the control of specific leaders. Resources include people, economy and social services, media, authority, buildings and infrastructure, weapons, media channels, emissaries, military troops and so forth. Leaders manage different amounts of resources in their own and other leaders’ territories; and there is the option/risk of seizing or having one’s assets seized. As earlier membership discussion mentioned, some of a given leader’s assets (e.g., followers, workers, foot soldiers, etc.) may variously choose the level at which they support their leader, or rebel and exit the group. This strength of membership thus directly affects a leader’s authority and size of many key resources (political strength, work force, security level) which in turn constrain the actions it can take. As a result, better leaders take care with how they treat their followers and allies, and tend to allocate public goods to secure these allegiances. In this manner, the constituency’s wishes and cultural norms tend to have an influence on outcomes over time.

As a result of these StateSim resource realities, agents of all types need to be able to reason about the spatiality of a region, distribution of resources, and location of allies and opponents. For instance, leaders will need a high level understanding of the regions so they can assess threats, wage campaigns and gauge success. Similarly, henchmen will need an understanding of the overall health of an area along with potential targets, safe havens, and level of resources available. Also, followers will need to understand the proximity and type of structures and services around them in order to carry out activities of daily life, mission tasks, and social interactions, while staying out of harm’s way.

To facilitate these needs, we implemented a terrain reasoning capability within StateSim. However, because we want the agents to be reusable and embeddable behind multiple 3rd party simulators, it is more important that agents be able to rapidly assess and reason about the “human terrain” rather than the actual physical terrain. Let us explain the difference. Most military simulators have (1G) agents that navigate a physical terrain from point A to B based on some coordinate system (lat-lon, x-y, etc.) but they have only surface (or no) understanding of what exists at those locations. There is no need to replicate this geo-spatial detail in StateSim. By contrast, StateSim agents are unaware of maps and geo-spatial locations. Leader agents are aware of regions and can tally up the power and vulnerabilities of other leaders in those regions, plus summary stats for # followers, # targets, # resources, etc. They use that information to plan out workflows they want their followers to execute. Followers, in turn, are aware of nodes in an abstracted regional or local network and meta-data about those nodes (e.g., type of building, level of security, services or goods it can provide, etc.). For instance, by using their workflows and choosing courses of action (COAs), follower agents migrate across a node network to go to work, eat meals, buy goods, pray, socialize, go home, etc. When they plug into 3rd party “proxy” agents (e.g., OneSAF, VBS2, Unreal, etc), StateSim agents can direct those proxies to where they want them to go and when. StateSim agents sleep while the 3rd party proxies navigate to the actual geo-location of the relevant node, and then awaken and decide what their proxy should do

next based on completing current workflows, carrying out new orders from leader agents, or simply responding to internal needs and starting other workflows.

We have designed a taxonomy to categorize and describe complicated objects that would be needed in a virtual environment with sophisticated agents and potential human player interaction capability. Considering our desire to expose culture in our models, we embed culturally-affected “collective perceptions.” Specifically, we use a culture’s systems of regulation and core beliefs to influence how and when structures and environmental features are perceived. For example, behaviors related to food purchasing may be driven by norms like market days or religious observances. Similarly, gender differences may affect accessible locations and interactions between agents.

2.5. *Towards a taxonomy of actions and workflows*

Our research to date on PoL applications has left us with a large taxonomy of PoL actions in a hierarchy that can be re-used as building blocks for next-generation agents. This can promote the development of a taxonomy of common action sequences useful across scenarios. That in turn, might enable faster construction of more complicated plans by hiding lower-level details and reusing lower-level action modules. Each category of agent in this taxonomy has its unique set of available actions and workflows. For example, group leaders have the highest level set (list of DIME actions), US/Coalition forces and Insurgents have mostly civil-military-related actions and workflows, and civilians have mostly non-kinetic related ones. These are packages of hierarchical action sets/workflows which may be expanded to a sequence of sub-level actions/workflows. As happens in PoL simulations, this expands to many 100s of pages of diagrams that must be coded, tested, verified, and maintained.

As mentioned in Section 2.3, in StateSim the temporal and spatial dimensions of the actions are handled abstractly so that they are reusable from scenario to scenario and across diverse 3rd party simulators. Temporally, actions are tied to StateSim tick rates which can be defined at the modeler’s discretion (e.g., 15 minutes is often chosen for immersive sims, while 1 or 2 weeks is often the tick interval for strategic sims). As described in the 2 case studies in Section 3, we often embed StateSim behind 3rd party simulators via some peer-to-peer protocol to drive the behavior of their actors. In these cases, StateSim agents are not tied to real-time, so they often hibernate while awaiting a 3rd party sim to carry out their instructions, awaking in time to check what they want done next. Likewise our terrain reasoning is similarly abstracted and allows StateSim agents to be highly portable.

2.6. *Algorithm for utility-based decision making to generate workflows*

We have stated in this paper that it is possible for 2G agents to generate their own workflows. Thus workflows are outputs that emerge from the agents, rather than inputs that guide the agents. Here is how this works. When a 2G agent carries out ‘daily life’, this does not specify what actions to take. Rather it relies heavily on the GSP value tree described in earlier Section 2.2. It works as follows (a more complete description is in [14, 16]). When authorized to conduct daily life activities, the agent continually performs the following “cognitive appraisal” decision loop on each tick of the simulation:

- 1) Observe the world – use perception (sensors) to determine what has happened to oneself, to others, and to the world at large since the previous tick of the simulation. Was the agent’s previous decision carried out successfully (or not), and what effects did it have in the world?
- 2) Orient internal parameters – Update all internal parameters to reflect the changes observed since the last iteration. Thus, update hunger (stomach tank), fatigue (need for sleep tank), stress levels, GSP tree leaf nodes, relationship strengths and alignments, and so on. Updating the GSP tree leaf nodes causes emotions to further activate or decay, thereby leading to overall sense of utility of the current state of the world.
- 3) Decide what to do next – This is done by re-invoking the GSP tree and emotion module to appraise the value of each of the available set of actions, one at a time. The overall utility of the world is computed if that action is taken (and its likely consequences occurred). These choices are ordered from highest to lowest utility, and the best response algorithm selects the highest utility action choice.
- 4) Act – The highest utility action choice is taken, with a random probability of it actually getting executed as desired.

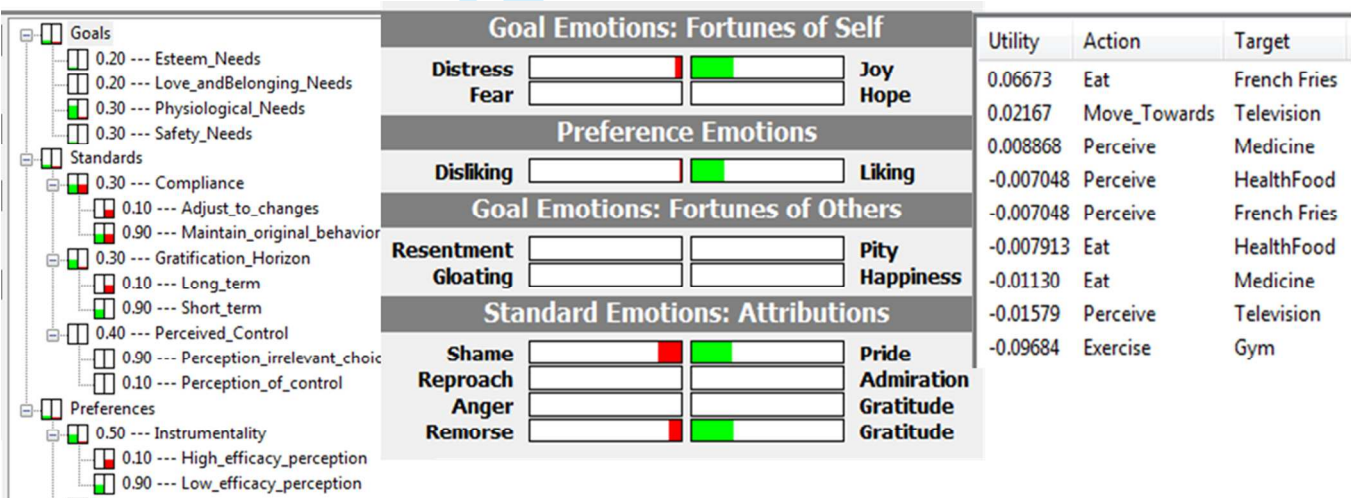


Figure 2. An Agent’s GSP Tree Activations, Emotions, and Utilities

As an example of this algorithm, let us hypothesize that there is an agent with GSPs as shown on the left of Figure 2. We can see this agent likes to maintain current behavior, has a short term gratification horizon, and places low efficacy on instrumentality (i.e., doesn’t believe his actions can change outcomes). In this example, this agent likes to eat junk food and watch TV, and eschews the Gym and healthfood. As time goes on in the simulation, his energy level drops and it observes its own physiology (Step 1) and sees it is hungry (Step 2). Appraising all its choices such as going to the Gym, taking Medicine, etc., it estimates Eating French Fries as the highest utility choice through the following process (Step 3). The GSP activations for any given choice are displayed on the leaf node branches of the GSP tree on the left side of Figure 2. There is a succeed (left) and fail (right) tank on each GSP node. The Figure currently shows the node success and failure levels for eating French fries. The resulting emotions about eating the fries are shown in the middle histograms. Summing up these positive and negative emotions gives the

utility score for French Fries. The utility of all possible actions are compared on the right side. Its top choice is eating the fries, while its second choice is to move toward the TV. It takes that action and then starts the algorithm again. Over time, the agent builds its own workflow as an output of its sequence of choices.

It should be noted that the StateSim engine can run the minds of 2G or 1G agents. This is done via a toggle switch that removes the cognitive model from a given agent's model library and replaces it with an FSM engine. In the latter case, the 1G agent will run pre-stored workflows. Thus, the 1G agents in Case 2 below carry out daily life and other missions according to pre-canned routines. It is possible for 2G and 1G agents to be mixed together. For instance, if the leader of a group of 1G agents is a 2G agent, it can use his cognition to determine what actions it wants its followers to carry out. For instance, an insurgent leader might send some agents out on the daily life mission and others will carry out a specific IED mission against a target of the 2G leader's choosing. The 2G agent uses his cognitive appraisal algorithm to determine who is an enemy to be attacked, and indicates how many followers should go on the attack (strength of the attack). If there are any 2G followers on the attack mission, they will use their own cognitive appraisal loop and determine how and when they want to do the attack (they might even abandon the attack if their GSPs lead them that way). The followers that are 1G have no such choice mechanisms. They simply activate on the IED mission workflows and carry them out against the specified target.

3. Results to date and a scale-up case study

Our primary research question posed at the outset is whether meso-level human behavior models (2G agents) can be PoL-good when used in tandem with 1G agents in large-scale simulations. It has been demonstrated countless times that 1G agents work well in large scale immersive simulations. Trainees learn the lessons, and users tend to be satisfied with the experience. However, as was discussed in Sect 1, these exercises are expensive, require a large number of pucksters to guide all the avatars, and take a lot of time and effort to set up and run. Pucksters are required since the 1G agents cannot be expected to fully and properly play the roles, carry out the missions, convey the cultural nuances, illustrate emergence of secondary and tertiary effects, or convey the scenario lessons. Based on earlier Table 1, it would seem in theory that 2G agents have the PoL Goodness to be able to satisfy such concerns. If this is also true in practice, it would represent a breakthrough since one could readily conduct/repeat the simulation exercises without all the lead time, setup effort, and expense of the pucksters for each repetition. In Section 1.3, we posed this question as R2, and in this section we explore the answer.

In our earlier research (R1) we first sought to answer whether 2G agents can improve realism and adaptivity in small immersive worlds. According to the sponsors, we were successful in demonstrating that and the software won an open competition [14]. In the current paper, we are trying to scale up from an immersive world of 100 agents (earlier case #1) to a strategic planning world of 100,000 agents (current case #2). In both cases, we are reusing the same HBMs to drive behaviors of up to 100 agents of the 3rd party PoL software.

In case #2, the case being reported in this article, we still have up to about 100 HBM agents, but this is a strategy style game and trainee interaction is constrained to defining the patrols/missions for 10,000 blue force 1G agents who carry these out to implement the players' plan. The HBM capability is reserved for the population (white) and insurgent (red) leader agents

and henchman, since these are the significant autonomous actors that we wish to make as realistic as possible. As opposed to having 100% of the agents following pre-scripted workflows (guided by pucksters), we will see whether the leaders being smart HBMs leads to acceptable outcomes. We will explore acceptability in terms of (1) acceptance by the customer that the intended lessons are able to be taught; (2) that the simulation can be set up and run as desired at different sites; and (3) the training instructors can run the simulation exercises themselves with no outside support (eg, computer operators, puckster players, and other support staff).

3.1. Case study - attack the network (AtN): scale-up to province

3.1.1 Purpose. We begin this case by describing a serious game, called Metis, whose goal is to facilitate the training of how to detect, deter, detain, and destroy counter-insurgent networks (consisting of 100-200 members) operating in a population of up to 100,000 civilians. Specifically, Metis is a cultural agent-driven simulation to support Army staff training in Attack-the-Network and Counterinsurgency operations: JWC (2011). As a result, Metis allows the staff to train real world decision-making and discover the impact of their decisions without real world risks.

The US Army sponsored Metis as a way of training the Army Decision Making Process applied to Attack the Network (AtN) doctrine [eg., see 4]. They specified a number of parameters such as size of the population, fixing the size of the insurgent network, and where to situate the game. Specifically, Metis is set in a northern province of Afghanistan covering a region with 5 districts, 2 of which are urban and 3 are rural. Aside from urban and rural groups, there are 3 Sunni cultural clans, the Taliban, government, drug lord families, and ISAF (US led coalition or blue force). Both the government and Taliban have competing institutional services and goods to distribute, though the Taliban often carry out night missions, illegal checkpoints (taxing those who wish to pass), kidnappings and ransoms, and other means of fund raising. Young male adult “opportunists” find work with the Taliban and the drug lords. The black market (opium fields) supports drug lords, but also migrant workers, opportunists, and the Taliban. Insurgent cells have leaders, financiers, bomb-makers (buy parts, carry illegal materials, assemble IEDs), and foot soldiers of varying loyalty. They are organized into independent cells, and if any are caught, their cell phones may be stripped to find their calling network and names of other agents to detain and question. Foot soldiers conduct reconnaissance, plant/detonate IEDs and suicide attacks, and carry out indirect and direct fire missions.

3.1.2 Pattern of Life (PoL) . All of the ethno-political factions and actors in Metis are driven by StateSim agents including faction leaders and followers. Up to about 100 factional leader and key follower agents are fully developed StateSim socio-cognitive models (PoL Goodness = 5 covering most of the right 2 columns of Table 1). These agents profile real world actors with similar political, economic, ethnic, and military goals. The leader agents make DIME decisions that they expect to be carried out by their followers. Key henchmen (full HBM followers) react to these decisions and outcomes, forming opinions about what leaders they support or not, mobilization levels of their group, and strength of membership. The tens of thousands of factional followers, foot soldiers, and workers are modeled as 1G agents in StateSim (PoL Goodness = 3 covering the left side of Table 1). These latter agents follow pre-scripted workflows, rules, and state transition graphs to simulate human activity across multiple

1
2
3 networks: criminal, government, coalition, insurgent, etc. As described at the end of Section 2.5,
4 they gladly carry out the orders of the factional leaders, provided the full-HBM followers are so
5 motivated. However, if the full-HBM followers become disenchanted with their leadership,
6 orders will be carried out half-heartedly or not at all. In this fashion leaders begin to lose power
7 and if they do not appease their followership, members might instead begin to defect and join
8 other factions, subject to saliences of group exit and entry.
9

10 These 1G agents continually update their internal state with input from the 2G agents to
11 reflect satisfaction with their situation, internal needs, alignments, and daily activities they are
12 carrying out (and where they are on the node network of facilities, infrastructure, and buildings).
13 In Metis, there are no 3rd party agents displayed continuously moving in real-time across the
14 landscape. Instead, the Metis screens require discrete visual feedback from StateSim, such as
15 showing the routes that blue agents take on patrols, the Significant Actions (SigActs) that happen
16 at various nodes, and casualty statistics that might result based on how occupied those nodes
17 were. So, a market node with an IED going off in daytime would have more casualties than in
18 the evening.
19
20

21 *3.1.3 Training Goals.* Metis is designed to run on a desktop computer, but allows whole staff
22 interaction from individual workstations through a client-server approach. The user interfaces
23 replicate the look and feel of Military Decision Making Process workflows that trainees would
24 use on a deployment in a Tactical Operations Center. As can be seen in Figure 3, a
25 Brigade/Battalion staff would figure out mission tasks, schedules, and routes for all the platoons
26 and squads (10,000 soldier agents), including what assets they require to safely accomplish those
27 missions. These missions implement the Commander's course of action (COA) and can cover the
28 gamut of the 100s of actions possible in StateSim.
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

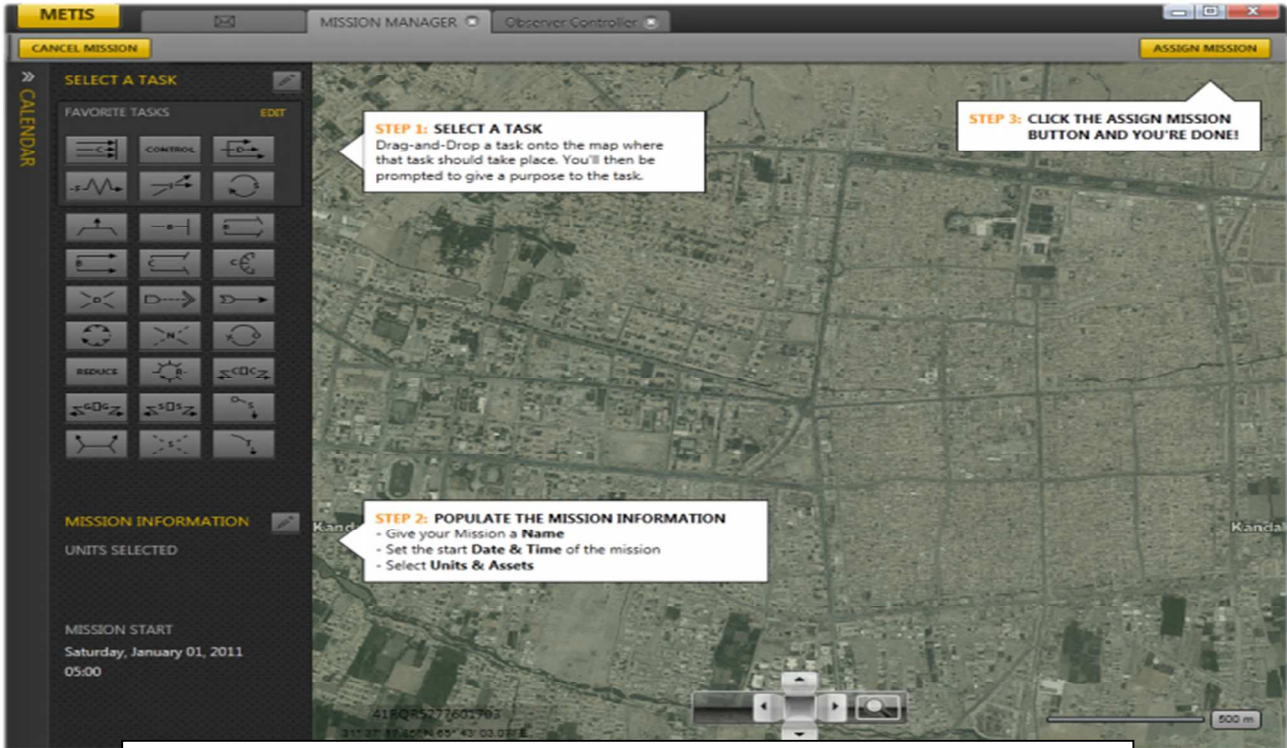


Figure 3. Edit the Blue Force Courses of Action

3.1.4 *Effects and Outcomes*. When the simulation runs, autonomous white and red leaders (and key henchmen) make decisions. The collection of up to 100,000 white agents carry out their daily PoL activities; and the red agents conduct insurgent activities, workflows, and missions. The blue forces interact with all of them as they “move across” the landscape on patrols, at security checkpoints, etc. At the end of each blue unit’s patrol, those units provide textual reports of what happened, plus a SigActs log is recorded and information extracted from interviews or detainee cell phone extractions (i.e., cellex) are likewise shared. The SigAct log for one 6 week run is summarized in a pie chart in Figure 4(a). By modeling patrol reporting, SigActs, and the system interfaces, Metis allows the staff to use their existing processes and tools to create and share products to facilitate their situational understanding of what is happening in the sim and to foster their decision making process. It is possible for blue players to read the Patrol Reports, SigAct Logs, and other extracted information to make forecasts of what red is trying to do. Red actions follow typical workflow steps. Each step has certain resource needs and timing. If a red agent is apprehended performing a step, one can infer what other red agents might be about to do, when it might happen, and what other SigActs and roles to be on the lookout for. At the end of every interval, blue force courses of action are summarized into overall simulation metrics as shown in Figure 4(b). Here we see a 6 week example of how the COAs led to slightly better governance (2nd line from top) and slightly worse security (4th line from the top). All other metrics appear flat, so there is little improvement in this short interval.

a) Significant Actions (SigActs) Occurring in the Various Regions

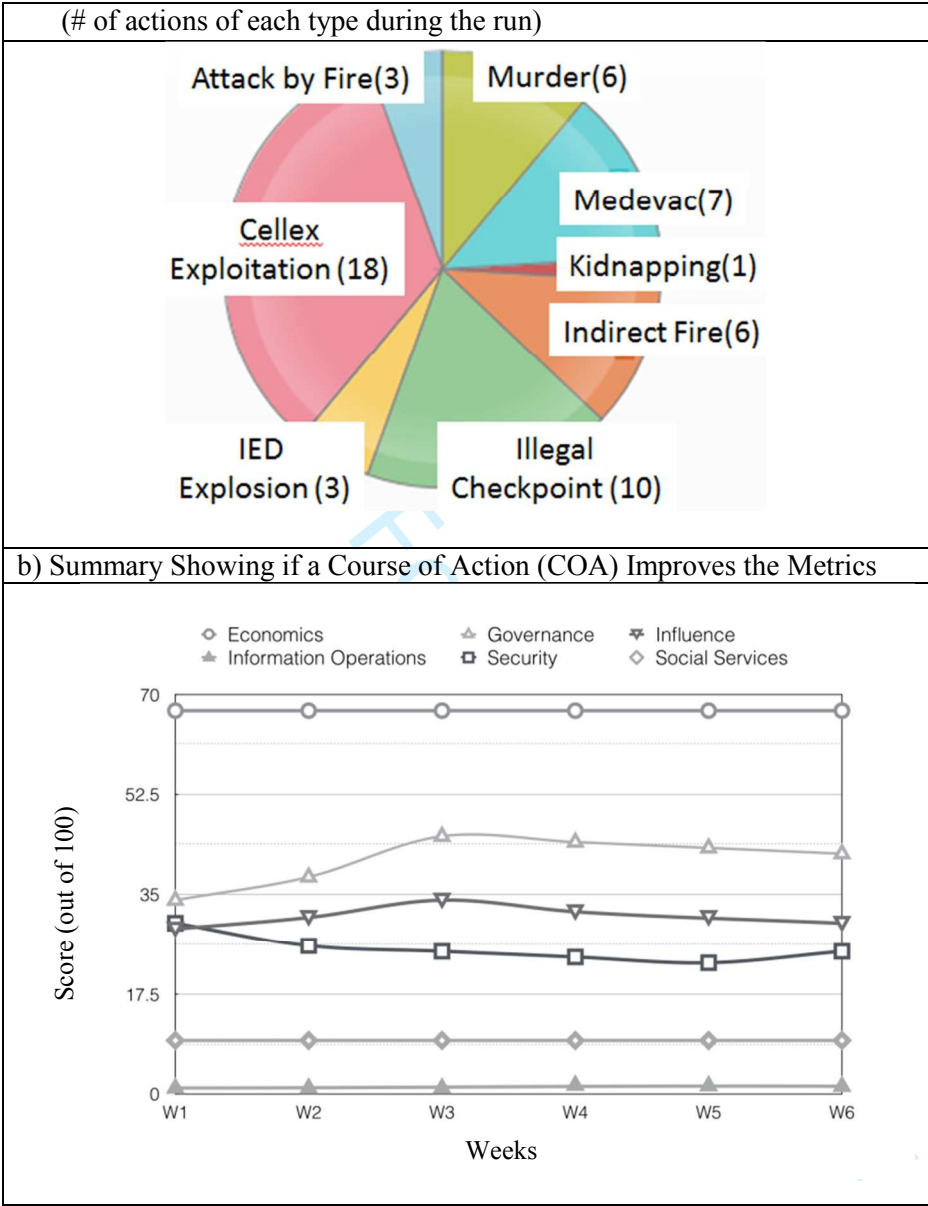


Figure 4. Some of the Outcomes of a Six Week Run.

3.1.5 Agent Adaptivity and Sensitivity Analysis..

Many parameters in Metis are fixed to accommodate the specs of the customer. This includes things like population size, coalition memberships, insurgent force size, and so on. StateSim is far more adaptive than this reflects and it permits agents to shift allegiances, leaders to change sides, and followers to join other groups. However, in Metis, the StateSim group size had to be set to block member entry/exit and thus there are only a fixed number of hostile agents to catch. In real life, rebel groups are often seen to shift in size based on government policies, perceived inequities, collateral damage affecting neutral parties, and so on. To directly test the effectiveness of this adaptive feature, we ran a controlled trial in which the government leader’s personality (see GSP tree weights in earlier Figure 2) was varied. Recall that leaders will make action choices based on their profiled personality. We are often asked by diplomats to vary a given leader’s personality profile as if he received a call from our President asking

(incentivizing) him to be more moderate. In this case we had government Leader A who is ingroup biased (i.e., he favors urban elites vs. rural poor), focuses on narrow interests, and is controlling. We also created an alternative government Leader B, profiled to be outgroup inclusive, focused on greater good, and open-minded. Otherwise the two leaders were identical. The scenario omitted blue forces and involved the host nation government and their security forces (green) arrayed against a small rebel group (red) with various civilian groups (white) caught in the middle including urban elites and rural poor. The rebel group wants a form of sharia law with them in charge of the region. They mostly operate in the area of the rural poor, come from that region, and have many kinship links straddling those two groups. We removed actions of rebels to intimidate civilians, so the scenario would depend entirely on civilians' own motivations. We then ran the scenario forward under each government leader to see actions they would choose and what population reactions would emerge.

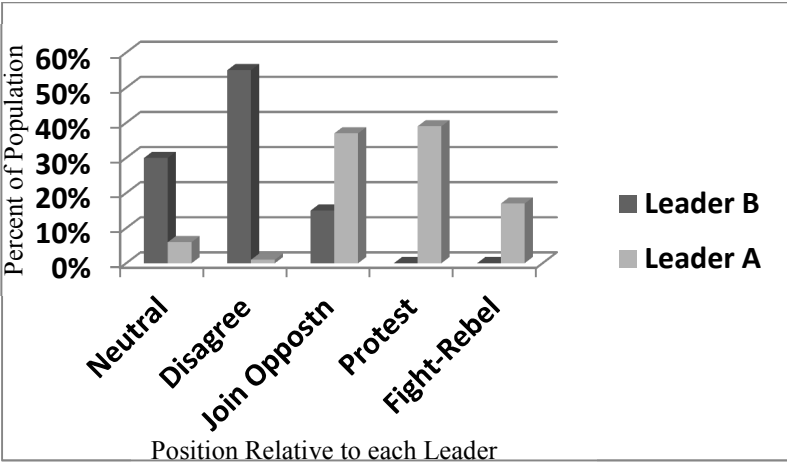


Figure 5. Impact of Leader Personality on Rural Poor
(A: Ingroup Biased, Narrow Focus, Controlling.
B: Outgroup Inclusive, Larger Good Focused, Open.)

The resulting civilian reactions are portrayed in Figure 5. The bars correspond to the sentiment or membership position of the rural poor relative to the government. The reader should recognize the x-axis in Figure 5 represents 5 of the grievance positions of the membership scale in Section 2.2. The height of the bars represents the percent of the rural poor that are choosing that grievance level. These are the average of the 2 weeks at the end of the year-long run. Specifically, in Figure 5 we can observe that under Leader A the rural populace grievances are on the right side, and they are heavily choosing the more extreme action sets. This is because Leader A is causing them to be oppressed, have lower quality of life and services, and suffer collateral damage of governmental attacks on rebels. Under Leader B, the rural poor are not happy with their treatment by the government (especially relative to the urban elites' treatment) but the majority go about their daily lives doing nothing about it except disagreeing. Only about 10% are opting to join the opposition (rebels). The result is more than sentiment, as it leads them to alternative daily action sets related to their choice of membership position.

These type of results reflect 2G agent micro-decisionmaking leading to emergent outcomes. It is important to be able to model such effects if one is to fully analyze the likely impacts of DIME actions. In the case of Metis, however, there is sufficient challenge to learning

how to use AtN doctrine and just solving the case of fixed size groups. The dynamics are, however, available for a later stage of AtN training.

3.1.6 Acceptance Test Results. At the end of this project, a company we teamed with was tasked to field and deploy Metis in 10 Army locations that train command staffs. This is equivalent to repeating an experiment 10 times. We were not allowed to visit those sites or participate in those deployments due to their classified status. The feedback we were allowed to collect was that deployments were successful, participants are satisfied with this tool, and the Army considers it a success. In essence, these are the results we were seeking to find out in the acceptance testing. Specifically, we wanted to explore acceptability in terms of:

(1) acceptance by the customer that the intended lessons are able to be taught – Indeed the Army accepted Metis and approved its distribution for AtN training of brigade/battalion staffs.

(2) that the simulation can be set up and run as desired at different sites – This was proven since it was done 10 times, once at each of the School houses doing command staff training.

(3) the training instructors can run the simulation exercises themselves with no outside support (eg, computer operators, puckster players, and other support staff) – In fact, that is what was stated when each site approved the deployment and adopted it for instructional use.

4. Conclusions and lessons learned

The first decade of PoL simulations saw the most popular approach to be human pucksters playing all the roles. The second most common approach is what we labeled as 1st generation (1G) agents consisting of Markov chains or FSMs with pre-scripted rule sets. This approach scales to 100,000s or higher numbers of agents and much research is ongoing to speed up and automate the generation of the rules and workflows/chains. For training of fixed scenarios requiring very large civilian populations with entirely pre-scripted behavior branches and no need for agent decision autonomy or connectivity, the 1G approach may very well be the best PoL-good choice. Often, however, in practice, military training simulations of this scale still require lengthy setup times and large staffs of pucksters guiding the avatars, computer operators, scorers and observers, and so on.

By contrast, in some training scenarios (and in all analytic ones), it is important that the agent activities are not rigidly pre-set. Instead, workflows and activities are expected to adapt dynamically, as a function of agent micro-decision-making as they perceive, react to, and try to influence others in the multi-layer network space. In such cases, it would be ideal if 2G agents or a combination of 1G and 2G were realistic enough to accomplish the training objectives and thereby save costs and provide self-service training without the need for all the operators, pucksters, and so on. To that purpose, Section 2 has explored the idea of a 2G agent framework fueled by a meso-level model of HBM models (StateSim) that cuts across and synthesizes many disciplines (economic, political, cultural, cognitive, civil, etc.). We then applied this 2G approach to a case study to test it.

Specifically, this test was to scale to 100,000 agents using a hybrid approach. In this case, 99% of the agents are 1G (PoL Goodness = 3) and carry out many dozens of pre-scripted, street-level workflows of daily life (10,000s of citizens), insurgent activities (the 150 red agents), and courses of action for security, stabilization, transition, and recovery (10,000 blue agents). But civilians and insurgents are guided by approximately one hundred 2G agents (PoL Goodness = 5)

that fill all the factional leader and key henchmen roles. These 2G agents autonomously make all the red and white factional decisions, mimic the behaviors of their real world counterparts, and react to situations emotively and moralistically. The latter sets the attitude, mobilization, and level of support of all the 1G agents for/against the various leaders. Case Study 2 demonstrated these dynamics and the PoL-goodness of their micro-decision making leading to emergent macro-behavior of the larger collective depending on which leader personality ran the government. This type of adaptive behavior is hard to accomplish with 1G agents alone.

In attempts to improve PoL-goodness, we must realize the limitations of the existing Markovian approach which only builds unthinking automatons – such agents only mimic their real-life counterparts in output actions while bearing little resemblances in underlying mechanisms (human reasoning, social relationships and so forth). Looking ahead, as demonstrated in our case studies, the HBM-driven agents with actual decision-making process and other human-like properties offer adaptive performance in simulations. In conclusion, we believe (1) though they are far more scalable than 2G agents, there are strict limits to PoL-goodness with 1G Markovian agents, and (2) scaling can be achieved and improved with judicious mix of HBM-driven 2G agents. This should provide an improved degree of PoL-goodness in simulations under scenarios requiring adaptive and emergent behavior.

Acknowledgement

Various parts of this research have been supported by the United States Marine Corps, Office of Naval Research, Pentagon/ Combating Terrorism Technology Support Office, Army/ Program Executive Office for Simulation, Training and Instrumentation, the Agency for International Development, and Pentagon/D9. Also, there have been many contributors to StateSim over the years. We are grateful to all of the above, but any errors are those of the authors alone.

References

1. Benoit, TS and Graffeo C. Beyond socio-cultural sensemaking. in *IITSEC Conference Proc.*, Arlington, VA: NTSA, 2014, No.14101
2. United States Marine Corps. *Introduction to combat hunter B1E0796 student handout*, Quantico, VA, United States Marine Corps, 2006
3. US Agency for International Development. *Shared progress, shared future*, Washington DC: US Agency for International Development, 2016
4. Joint Warfighting Center. *Attack the network*, Suffolk, Virginia: Joint Warfighting Center, US Army, 2011
5. Buckland M. *Programming game AI by example*, Plano, Tx: Woodward Publishing, 2014
6. Van Hemel, SB, MacMillan J, and Zacharias GL. (Eds.). *Behavioral Modeling and Simulation: From Individuals to Societies*, National Academies Press, 2008.
7. Numrich SK and Picucci PM. New challenges: human, social, cultural, and behavioral modeling. Ch. 26 in A. Tolk, *Engineering principles of combat modeling and distributed simulation*, 2012; pp. 641-667.
8. Folsom-Kovarik JT and Schatz S. AI challenge problem: scalable models for patterns of life. *The AI magazine* , 2014; 35.1: 10-14.

9. Hershey DJ and McKeown DM. Automatic annotation of urban environments to support pattern of life behaviors. *IMAGE 2012 Conference Proc.*, Scottsdale: The IMAGE Society, 2012.
10. Blank W and Crane A. Smartobjects and agents for autonomous high population simulated environments. *IITSEC Conference Proc.*, Arlington, VA: NTSA, 2011, No.11098
11. Watson S, Banzhaf W and Vardy A. Automated Design for Playability in Computer Game Agents, in *IEEE Conference on Computational Intelligence and Games (CIG)*, IEEE, 2014.
12. Speicher D and Wilbert, D. Simulating urban traffic in support of the Joint Urban Operations experiment. In *The Interservice/industry training, simulation and education conference(IITSEC)*, 2004.
13. Epstein JM and Axtell R. *Growing artificial societies: social science from the bottom up*. Wash. DC: Brookings Institution Press, 1996.
14. Silverman BG, Pietrocola D and Nye B. Rich socio-cognitive agents for immersive training environments: case of nonkin village, *JAAMAS*, 2012; 24.2: 312-343.
15. Bharathy GK and Silverman BG. Applications of social systems modeling to political risk management. *Handbook on decision making*. Berlin: Springer, 2012; 331-371.
16. Silverman BG, Sun D, Bharathy G and Weyer N. Speeding model creation through reuse: case of the StateSim generator, J. Cohn, S. Schatz (eds), *Modeling socio-cultural influences on decision making*, Boca Raton: CRC Press, 2016.
17. Ranis G and Fei J. A theory of economic development, *American Economic Review*; 1961; 51: 533-565.
18. Cho, K., Iketani, N., Kikuchi, M., Nishimura, K., Hayashi, H. & Hattori, M. BDI model-based crowd simulation. In *Intelligent Virtual Agents*, Springer.2008, pp. 364–371.
19. Hindriks, K. V., van Riemsdijk, M. B., Behrens, T., Korstanje, R., Kraaijenbrink, N., Pasma, W., de Rijk, L. Unreal goal bots: connecting agents to complex dynamic environments. In *Proc. Of AGS '10*. 2010.
20. Adams, C, Gaudou, B, "Bdi agents in social simulations: a survey," *Knowledge Engineering Review*, 31:3, pp. 207-238, June 2016.
21. Wolfe, SR, Sierhuis, M, Jarvis, P A, To BDI, or not to BDI: design choices in an agent-based traffic flow management simulation. In *Spring Simulation Multiconference*, 2008, 63–70. International Society for Computer Simulation.
22. GE, Y., Meng, R., Cao, Z., Qiu, X, Huang, K, Virtual city: An individual-based digital environment for human mobility and interactive behavior. *Simulation*, 2014, 90(8), 917–935.
23. Zhang M, Verbraeck A, Meng R, Chen B, Qiu, X,. Modeling Spatial Contacts for Epidemic Prediction in a Large-Scale Artificial City, *Journal of Artificial Societies and Social Simulation (JASSS)*, 2016, vol. 19(4), pages 1-3.