Automating Forecasting and Exploration of Complex Simulation Effects (AFECSE)

Janet Wedgwood
Zachary Horiatis
Thad Konicki
Lockheed Martin Advanced Technology Laboratories
3 Executive Campus
Cherry Hill, NJ 08002
856-792-9879, 856-792-9883, 856-792-9877
jwedgwoo@atl.lmco.com, zhoriati@atl.lmco.com, tkonicki@atl.lmco.com

Keywords:
Effect-Based Operations, Course of Action Analysis, Semantic Conceptual Model, OWL, Ontology, SysML/UML

ABSTRACT: Support of military campaigns requires new approaches for effective generation of desired effects, and continuous adjustment of the actions, for the entire life of the campaign. Military planners are moving to Effects-Based Operations (EBO) [1] to achieve these desired effects for a combination of Diplomatic, Informational, Military, and Economic (DIME) actions.

As military planners move from pure military operations to Effects-Based Operations (EBO) [1], they will need tools to enhance their understanding how the desired Political, Military, Economic, Social, Infrastructure, Information (PMESII) effects can be achieved through a combination of Diplomatic, Informational, Military, and Economic (DIME) actions. Engineers at Lockheed Martin Advanced Technology Laboratories are developing the Automating Forecasting and Exploration of Complex Simulation Effects processes as part of their research into the use of Modeling and Simulation to develop and analyze campaign-level effects-based operations. It uses innovative multi-paradigm simulations of DIME actions on models to determine the probable desired effects, as well as the undesirable effects, while developing a better understanding of second and third order effects. In order for this technology to be useful to military analysts and planners, it must be made accessible to non computer scientists. Our goal is to help analysts and planners easily exploit the power of Modeling and Simulation for exploring Effects-Based Operations through automation of scenario development, model instantiation, integration and initialization and Course of Action (COA) development, simulation and analysis.

1. Goals

Support of military campaigns requires new approaches for effective generation of desired effects, and continuous adjustment of the actions, for the entire life of the campaign. Military planners are moving to Effects-Based Operations (EBO) to achieve these desired effects for a combination of diplomatic, informational, military, and economic actions.

Engineers at Lockheed Martin Advanced Technology Laboratories are developing the Automating Forecasting and Exploration of Complex Simulation Effects (AFECSE) processes as part of their research into the use of Modeling and Simulation to develop and analyze campaign-level effects-based operations. It uses innovative multi-paradigm simulations of Diplomatic, Information, Military, Economic (DIME) actions on Political, Military, Economic, Social, Infrastructure, Information (PMESII) models to determine the probable desired effects, as well as the undesirable effects, while developing a better understanding of second- and third-order effects. No single model or modeling paradigm can provide the rich set of integrated behaviors needed to adequately simulate effects-based operations. Currently, the complexity and time required to integrate a diverse set of multi-paradigm, multi-domain models is prohibitive unless automated assistance is available. Our goal is to develop a process that enables people who may not be computer scientists to more easily exploit modeling and simulation for exploring EBO through automation of scenario development, model instantiation, integration and initialization and Course of Action development, simulation and analysis.
We will define requirements, develop processes and implement wizards to rapidly integrate and configure simulation models enabling non-computer scientists to assemble composite model simulations ready for execution to explore actions and effects in regions of interest. Through automation of various aspects of modeling and simulation including scenario development, model integration, initialization, COA generation, data collection and analysis, the (re)usability of modeling and simulation (M&S) for EBO will be greatly enhanced. Scenarios, models, COAs, actions, and effects that are represented platform independently can be reused with multiple model sets on any platform by developing the applicable platform specific transformations. Applications for this technology include COA generation/strategy analysis for command and control and experiment support for modeling and simulation programs, offering substantial benefits to the various technology owners.

2. Approach

2.1 Overview

We are developing user processes and libraries and user-friendly visualizations of important world phenomena using representation technologies that can be applied to exploit power of modeling and simulation while aiding in hiding complexity. We have defined two processes: a Scenario Generation Process and an Experimentation Process. These two processes are our response to the two biggest challenges to automating the instantiation of a scenario. These are to represent the platform-independent and platform-specific description of the integrated model set (nodes and relationships, models, events, and data flow, COAs and Effects) in such a way that an autocoding tool/wizard can generate the simulation platform specific model integration code and to allow the analyst to explore the solution space (change the model parameters) in such a way that they are not overwhelmed by unnecessary, irrelevant complexity. Our Scenario Generation Process starts with a platform-independent Semantic Conceptual Model (SCM) represented in the Web Ontology Language (OWL) [2] that represents the nodes (people, places, and things) and relationships of interest [3]. We are developing easy-to-use tools that: map these Semantic Conceptual Models to specific simulation execution platforms and simulation models; aid in developing and simulating COAs (taking actions of a specified intensity on nodes for specific durations) and provide analysis and visualization capabilities to observe and analyze the effects caused by the COAs. Our Experimentation Process enables users to easily inject actions and characterize the resulting landscape using DOE/Optimization methods or “What-if” analysis. This process is supported by Action and Effect entities in the SCM. In addition, the Experimentation Process has a user interface to set up the DOE/Optimization methods, modify parameters, and select which actions to take on which entities with what intensity.

The two processes are best described in the context of an example scenario. In this case, the hypothetical scenario involves a hypothetical terrorist group (Group1) performing attacks on the electrical infrastructure in Baghdad. The analyst is interested in understanding how Group1’s attacks may be affecting popular support for Group1’s leader, Leader1, and whether by improving access to electricity in a particular province, Leader1’s popular support may be reduced. The scenario-generation process shown in Figure 2.1 begins with the platform-independent semantic representation of the scenario that the analyst wants to explore, the Semantic Conceptual Model (SCM). This “virtual world” includes nodes that represent important leaders, social groups, infrastructures, etc. and relationships between them. These are represented as an ontology in OWL. In this case, the analyst would view a nodal diagram of the situation, such as the one shown in Figure 2.2, and then select models such as an electrical infrastructure model, a social model of the people in the province, a model of Group1 and Leader1, and perhaps models of local leaders (both government and non-government).

Next, the nodes and relationships in the SCM are mapped to simulation models. However, node and relationship information is not sufficient to allow an autocoding tool to generate a functional simulation. The structural and dynamic information that indicate how the models are “wired together” (data and events flow between models) is also needed, but it is better represented in UML/SysML [4] than in an ontology. This wiring can be either actual parameter passing or event generation. Data flow information allows the nodes to gather inputs from other nodes and disperse model results to other nodes after the models have run. Events allow nodes to communicate with each other. They can carry data with them and can be used to signal nodes to execute internal code or external models. This additional information is represented in platform-independent and platform-dependent sequence diagrams, activity diagrams, etc. Once the models are selected, a previously developed platform-independent structural description of how the models are “wired together” is retrieved from the library and, along with the SCM and platform specific-transformation information, is used to autocode the integration code.

Figure 2.3 shows how a portion of the scenario in Figure 2.2 (which has only platform-independent information in it) is enhanced with additional information that will allow it to be targeted to the Dynamic Integration Architecture System (DIAS) [5] from Argonne National Laboratory (ANL). The complexity quickly grows when, in addition
Figure 2.1. Our process combines a Platform-Independent Model of the scenario and the simulation models with Platform-Specific transformations to assist users in integrating simulation models for use in effects-based operations exploration.

Figure 2.2. The Semantic Conceptual Model (SCM) represents the interesting people, places, things, and relationships that can be explored using an integrated set of simulation models.
to this scenario view, one considers that there is a sequence diagram and block diagram view that also have both platform-independent and platform-dependent portions. The analyst would be able to view as much or as little of this complexity as desired. It is quite possible that an analyst could go through an entire experimentation phase only using diagrams similar to Figure 2.2 for describing the scenario.

Unlike many approaches, the semantics of each simulation model is not required to match the common semantics of an overall model. Our approach is exploring methods that allow the semantics of the nodes to vary based on the simulation model that is used to implement it. This way, any new functionality in the model will be available to any other node in the integrated system that can take advantage of it. These methods include a flexible ontology mapping capability, allowing each simulation model to be used to the greatest extent possible. We are also investigating how far up the user chain we can place such a tool.

Continuing with our processes, after the analyst has set up the scenario and instantiated the integrated simulation, we move to the experimentation process, as shown in Figure 2.4. The first step of parameterization of the SCM is important in that this gives the analyst the ability to change any parameter values in the SCM so they apply to the particular region and purpose of the analysis. For example, the analyst might feel that the “military aggressiveness” parameter of Leader1 should be higher or lower based on current events. Next, the analyst will define and analyze candidate COAs by observing the effects produced by introducing actions on nodes during the simulation. For example, the analyst might want to observe the effects of increasing the number of hours electricity is available in a particular province. The goal might be to see if the people (the social model) will lower
their support for the Leader1 if Group1 continually knocks the power out after it has been restored by the coalition. The analyst uses a simple timeline tool to say when to take the action (increase the electricity availability by some percentage) and for how long. For COA development, multiple actions can be “active” at any time. It is important to note that the analyst only interacts directly with the model through intuitive graphical user interfaces—never through the code.

Three methods are available to explore the solution space. A design of Experiments [6] or optimization engine can be integrated with the modeling and simulation platform, or the user can choose individual “What-if” type explorations. This is an interesting area but is beyond the scope of this paper.

Figure 2.5 shows more detail of how actions might be represented in the SCM. This data can be used to populate a user-oriented graphical interface that shows the actions that might be taken on particular nodes. The accompanying model information at the bottom of Figure 2.5 helps the model wizard decide which models are the best set for the scenario based on the actions that they support.

Our scenario generation and experimentation processes are promising approaches to making modeling and simulation friendlier, ever more automated, and therefore, more available to non-computer scientists. We are using platform-independent descriptions of the problem combined with libraries of platform-specific transformation information to hide the complexity of modeling and simulation while increasing its usability for EBO. We are continuing our work in this area on internal research and development and believe it has many promising application areas outside of EBO.

3. References


Figure 2.5. Additional information needed to support COA generation and, in the future, a model recommendation tool show the user actions that might be taken on certain nodes.
Authors Biography

JANET WEDGWOOD is a Principle Member of the Engineering Staff at Lockheed Martin Advanced Technology Laboratories. Ms. Wedgwood was Principal Investigator on the LM team for the DARPA IBC program. In this role, she not only coordinated all technical aspects of the program, but was highly involved in the model integration activities. Her current research in developing the concepts and processes behind semantically-driven model integration automation is highly informed by her IBC experience. On IBC, multiple models written in different paradigms were integrated using the DIAS framework. Ms. Wedgwood holds an BSEE from RPI and MSEE from Stanford University.

ZACH HORIATIS is a Senior Member of the Engineering Staff at Lockheed Martin Advanced Technology Laboratories. Mr. Horiatis’s work includes systems architecture, design, integration and development of intelligent complexity science (agent based systems), system dynamics and collaborative engineering disciplines as well as configuration management processes. Mr. Horiatis is developing automation mechanisms of various aspects of modeling and simulation including model integration, initialization, and COA generation.

THADDEUS KONICKI is a Senior Member of the Engineering Staff at Lockheed Martin Advanced Technology Laboratories. Mr. Konicki is currently involved in the system design, tool evaluation, and pilot implementation for the platform independent modeling component of the AFECSE demonstration. Mr. Konicki has experience in enterprise system development in the areas of experiment design systems and technology refresh management. Mr. Konicki holds an MSCS from St. Joseph’s University, an MSECE from Drexel University and a BSEE from Drexel University.