Laboratory for Unconventional Conflict Analysis & Simulation

Technical Supplement for Course of Action Simulator

Luke A. Maier
Technical Paper 2014-1v2

Summary

The Course of Action Simulator (CAS) was developed as an analytical and educational model of how two states adopt courses of action in response to threats, resource constraints, and complex priorities. This analytical tool was featured in several presentations and publications, and its purpose is to help explore how agent-based modeling can inform security studies. This technical paper outlines the model's underlying calculations and algorithmic theory to make it as transparent as possible so that its limitations and strengths can be evaluated by users seeking to advance strategic modeling.

Modeling strategic decision-making as algorithmic behavior provides a disciplined, transparent, and potentially data-driven method for forecasting how threats evolve and how we can address them. Designing algorithms that articulate how states make security decisions can help U.S. strategists understand and forecast how other nations will react to changes in their security environment. Advancing algorithmic models of state behavior can provide valuable opportunities to capitalize on rapidly advancing computational power and to address the need for strategic-level simulation for national security research. The following sections: 1) introduce the model's basis in international relations scholarship, 2) explain why a state's selection of a course of action resembles algorithmic behavior, and 3) detail the calculations that CAS uses to represent states' identities, power, and priorities in an agent-based model.

Citation: Maier, L. 2014. Technical Supplement for Course of Action Model. Technical Paper 2014-1v2. Fort Bragg, NC: Laboratory for Unconventional Conflict Analysis and Simulation.

Introduction

States use formal and informal processes to make decisions regarding their national security. These processes can be grounded in law or tradition, or they can operate more subconsciously and without structure. In many states, these processes are relatively stable, though some states are more sporadic and *ad hoc*. But over long lengths of time, all states refine their decision-making processes to adapt to changing political, economic, and social circumstances and new national security threats.

International relations scholarship suggests there are three alternative decision-calculi that explain security strategy. These broad categories of prediction are conventionally grouped into the realist, liberal, and constructivist paradigms. Each offers a comprehensive set of explanations about what drives state behavior in the international politics. These explanations all treat an individual nations' behavior as a reaction to the nature and conditions of the international system, which can be considered a step-by-step process in which a state senses threats in its environment and applies a procedure (either intentionally or subconsciously) to decide how to react to those threats.

This step-by-step procedure for determining action lends itself well to description as an algorithm. An algorithm is a step-by-step set of procedures applied to change inputs into outcomes. Certainly, the decision-making processes of each state vary significantly, but the three main international relations paradigms provide very useful generalizations about themes in all states. Also, these processes operate at all three of Kenneth Waltz's "images"—or levels—of analysis: the drives of state behavior emanating human nature, individual leaders, and international system. Treating states' strategy-making as algorithms provides a rich opportunity to integrate insights from a variety of fields that have studied decision-making through, for example, computational, philosophical, cultural, psychological, and national security studies.

Algorithmic modeling of international relations has a fundamental trade-off between creating a generalizable tool and a tool that is sufficiently specified to provide specific predictions about individual states. In theory, it is possible to use algorithms to articulate the decision-making models of each state, but that would require time-consuming research and highly accurate information on the social, economic, historical, and psychological characteristics of each state's government. Until this is feasible, it is useful to attempt to create an algorithmic model that can be generalized to many states.

CAS provides a generalized model whose algorithms were derived from the three main international relations paradigms. CAS draws heavily from agent-based modeling and simulation—a broad category of computational methods that simulate the behavior of multiple actors who follow simple sets of rules. An additional advantage of agent-based models is they allow an analyst to observe "emergent outcomes"—outcomes that can only be observed at the aggregate level. The rules in an agent-based model can be articulated in the language of algorithms, and the remainder of this paper describes how CAS incorporates international relations concepts as rules into a simulation of two states and their adoption of courses of action based on simple priorities and resource constrains.

Algorithms and National Security Decision-making

An algorithm is a series of calculations that describes how an initial "system state" proceeds to future states. Designing an algorithm progresses through creation of high-level, implementation, and formal descriptions, which each represent progressive levels of specificity (Sipser 2012, p. 185). High-level descriptions provide broad, prosaic accounts of the state's decision-making model; these can be designed by international relations and subject matter experts who might have no background in computation. This requires identifying the variables that influence the decision (i.e., the "process variables") and the variables that indicate change has occurred (i.e., the "output variables"). It answers questions like "what influences the state's behavior?" The next, more specific level of description is of how the algorithm implements the process. Implementation description uses words or diagrams to portray the steps of the process in a specific order. For example, this answers "how do those influences affect state behavior?" (Often in computer algorithms, these are expressed as bulleted pseudo-code.) The most specific level is formal description, wherein the analysts writes the series of equations that specifies how the description is implemented.

The formal algorithm comprises various functions of the processed variables which, together, represent the *process* that changes the inputs into a decision. Consider the following discussion of algorithm $\boldsymbol{g}$. The step function $\boldsymbol{g}$ is used to change the values of the output variables (i.e., foreign policies), defined as set **R**. The values in **R** change from t to t+1 such that:

$$\boldsymbol{g} : \mathbf{R_t} \rightarrow \mathbf{R_{t+1}}$$

After an initial run at time *t*, the algorithm can be applied to $\mathbf{R_{t+1}}$ to calculate $\mathbf{R_{t+2}}$ and so on until $\mathbf{R_{t+n}}$, where *t+n* is the desired end-state.

These algorithmic processes can take on two forms (Holland 1995). The first is an "internal model" that states adopt. Internal models include, among other processes, decision-making processes adopted inadvertently. This could be a subconscious process (such as the various types of fallacies or miscalculations and misperceptions that systematically influence state behavior), or a process that is deterministic. A deterministic process has no alternatives; it only has one possible outcome for each set of inputs, and thus can be thought of as an "action-reaction" model.

The second and perhaps richer form is the "overt model," by which states actively compare alternatives through a specified deliberation process to actively optimize the decision's outcome. For example, academic policy analysis often follows this model, where an analysis will identify a criteria for measuring success, identify alternative actions to meet that criteria, determine how comparatively well each alternative meets the criteria, then implement the best alternative. This second model provides many opportunities to model goal-seeking behavior, such as is envisioned in agent-based models and simulations. Both the internal and overt models can be used to represent states' heuristic mechanisms, among other things.

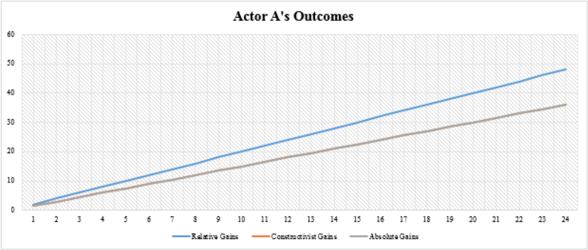
Algorithmic strategic decision-making can draw heavily from concepts in artificial intelligence planning, in which a decision-maker "is given a set of possible states and is asked for a sequence of actions that will lead it to a desirable world state" (Vidal 2007). Advanced non-deterministic or randomized methods that draw from Monte Carlos decision chains and Markov processes (e.g., see Schrodt 2000) can be used to simulate all of the possible strings of moves (where a "string" is a certain series of moves) and compare their various utilities (Vidal 2007). Other potentially valuable outputs from these methods could be Poisson Distributions of occurrence and perhaps more interestingly—in limited instances where retrodictive validation is possible—probabilistic forecasts.

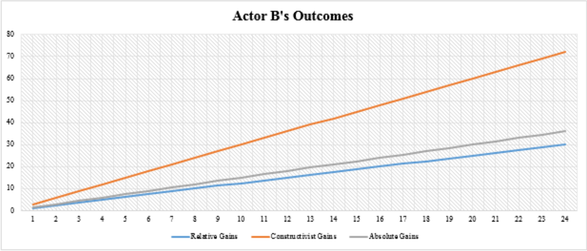
These algorithms can be incorporated into computer platforms such as game theory simulators or agent-based models (e.g., see Cederman 2003), all which have rising reliability and relevance in strategic analysis as our computational power and information accuracy improve. The process of designing such an algorithm begins by scrutinizing the relevant literatures from international relations, leadership and decision psychology, history, etc. to identify relevant processes at play in a state's decision-making. The next step would be organizing these influences into stepped or explicitly interrelated processes, and the final steps would include mathematically formalizing these steps and then translating them into a computer programming language such as Python. The next section outlines how CAS is designed following this process.

The Mechanics of CAS

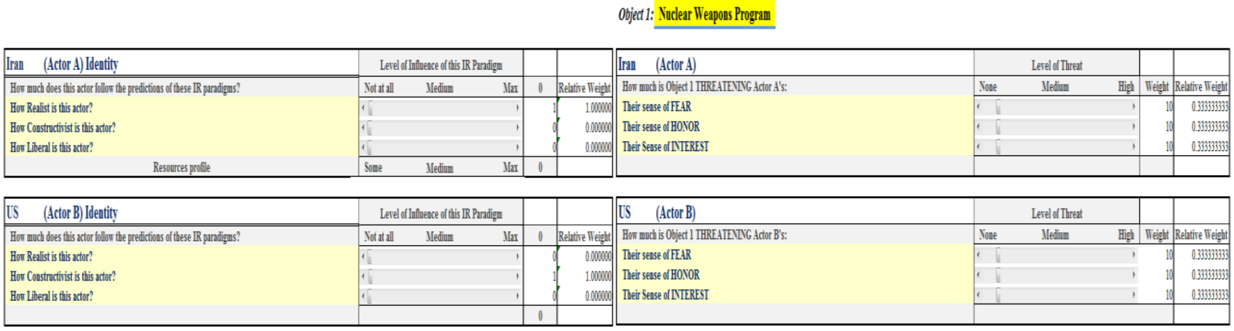
CAS aims to demonstrate how international security studies can apply agent-based models to simulate how states develop courses of action based on resource constraints and broader decision-making goals. Their decisions influence each other, and their courses of action are adopted to achieve specified goals. The goals are dually defined by the actor's identity as a realist, liberal, or constructivist state, as well as their ideal deployment of eight elements of national power in each of the issues. Following modeling convention, these security issues are referred to as "objects." The objects have a location in a three-dimensional space whose axes (in set $\boldsymbol{v_t}$) represent the level of three kinds of threats: threats to fear, threats to honor, and threats which comprise set $\boldsymbol{v_t} = \{f_t \ , h_t \ , i_t \}$. These dimensions of threats are differentially emphasized by the three main international relations: realists emphasize states make international relations decisions based on threats to fear, liberals emphasize threats to interest, and constructivists emphasize threats to honor, among other considerations broadly defined.

Each object has a "location" ($\boldsymbol{l_t} = [f_t, h_t, i_t]$), with lower values on an axis representing low threat levels in that threat category for that object. High values represent the actor is experiencing high levels of threat on that issue. The actor's goals are to reduce threats to their priorities. Realists prioritize reducing threats to fear, which relate to hard power resources including military, economic, and technological resources. Liberals seek to reduce threats to interests, which include economic, diplomatic, human, technology, and informational resources. While realists seek to increase their *relative* power, liberals seek to increase their *absolute* power. Constructivists seek to reduce threats to honor, and they also seek to fulfill certain ideal types which are defined as the possession of definable levels of each power resources, which we refer to as "power configurations." The model measures each actor's performance by these metrics in graphs like the following:





Few threats fit entirely into a single category; most threats have qualities of all three. Similarly, few states entirely follow one of the three IR theories. The initial threat levels for each object is set via sliders like those pictured below to the right. Each IR theory explains some state behavior, and certain states fit some theories better than others. To account for this, any particular state's behavior is influenced by complex combinations of these international relations theories, as described as list $w = [r, c, l]$ where r, c, and l are the relative weights of each of the IR theories. Using sliders like those below to the left, the model asks the analyst to estimate the degree (on a scale of 0-100) to which each state's decisions match each of the three theories' predictions. For example, while some states prioritize the reduction of military threats, others prioritize the reduction of economic threats.



Object 1: Nuclear Weapons Program

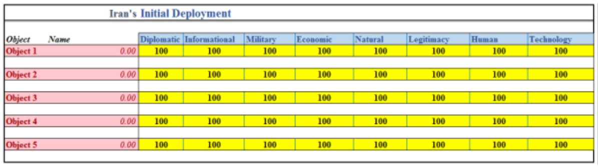
Iran (Actor A) Identity	Level of Influence of this IR Paradigm				
How much does this actor follow the predictions of these IR paradigms?	Not at all	Medium	Max	0	Relative Weight
How Realist is this actor?				1	1.000000
How Constructivist is this actor?				0	0.000000
How Liberal is this actor?				0	0.000000
Resources profile	Some	Medium	Max	0	

Iran (Actor A)	Level of Threat				
How much is Object 1 THREATENING Actor A's:	None	Medium	High	Weight	Relative Weight
Their sense of FEAR				10	0.333333333
Their sense of HONOR				10	0.333333333
Their Sense of INTEREST				10	0.333333333

US (Actor B) Identity	Level of Influence of this IR Paradigm				
How much does this actor follow the predictions of these IR paradigms?	Not at all	Medium	Max	0	Relative Weight
How Realist is this actor?				0	0.000000
How Constructivist is this actor?				1	1.000000
How Liberal is this actor?				0	0.000000
				0	

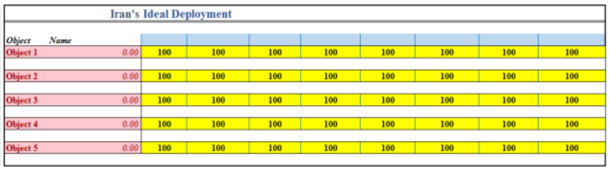
US (Actor B)	Level of Threat				
How much is Object 1 THREATENING Actor B's:	None	Medium	High	Weight	Relative Weight
Their sense of FEAR				10	0.333333333
Their sense of HONOR				10	0.333333333
Their Sense of INTEREST				10	0.333333333

In addition to simulating a state's complex threat calculus, the model also considers how particular actions (referred to as "moves") deploy elements of national power, which we refer to as "resources." The initial and ideal resource distributions are input into these tables:



Iran's Initial Deployment

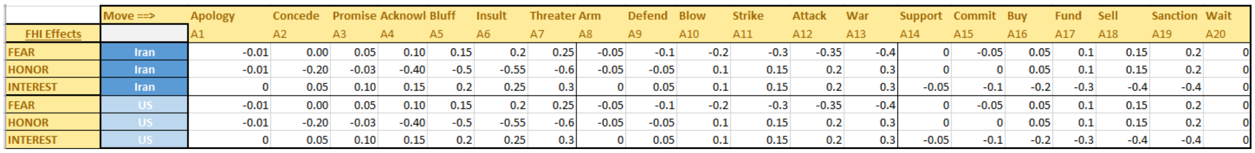
Object	Name	Diplomatic	Informational	Military	Economic	Natural	Legitimacy	Human	Technology
Object 1	0.00	100	100	100	100	100	100	100	100
Object 2	0.00	100	100	100	100	100	100	100	100
Object 3	0.00	100	100	100	100	100	100	100	100
Object 4	0.00	100	100	100	100	100	100	100	100
Object 5	0.00	100	100	100	100	100	100	100	100



Iran's Ideal Deployment

Object	Name								
Object 1	0.00	100	100	100	100	100	100	100	100
Object 2	0.00	100	100	100	100	100	100	100	100
Object 3	0.00	100	100	100	100	100	100	100	100
Object 4	0.00	100	100	100	100	100	100	100	100
Object 5	0.00	100	100	100	100	100	100	100	100

A highly realist actor, for instance, would not take moves that deplete their hard-power resources. A highly honor-driven state would not adopt courses of action that deplete their information, human capital, and legitimacy. Each type of move has trade-offs. Some are good at achieving realist goals, while others only achieve liberal or constructive goals. Some moves are more balanced, and achieve limited progress for each axis. The initial settings are as follows, but the analyst can open the background pages and edit these move effects:



	Move ==>	Apology	Concede	Promise	Acknowl	Bluff	Insult	Threater	Arm	Defend	Blow	Strike	Attack	War	Support	Commit	Buy	Fund	Sell	Sanction	Wait
FHI Effects		A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13	A14	A15	A16	A17	A18	A19	A20
FEAR	Iran	-0.01	0.00	0.05	0.10	0.15	0.2	0.25	-0.05	-0.1	-0.2	-0.3	-0.35	-0.4	0	-0.05	0.05	0.1	0.15	0.2	0
HONOR	Iran	-0.01	-0.20	-0.03	-0.40	-0.5	-0.55	-0.6	-0.05	-0.05	0.1	0.15	0.2	0.3	0	0	0.05	0.1	0.15	0.2	0
INTEREST	Iran	0	0.05	0.10	0.15	0.2	0.25	0.3	0	0.05	0.1	0.15	0.2	0.3	-0.05	-0.1	-0.2	-0.3	-0.4	-0.4	0
FEAR	US	-0.01	0.00	0.05	0.10	0.15	0.2	0.25	-0.05	-0.1	-0.2	-0.3	-0.35	-0.4	0	-0.05	0.05	0.1	0.15	0.2	0
HONOR	US	-0.01	-0.20	-0.03	-0.40	-0.5	-0.55	-0.6	-0.05	-0.05	0.1	0.15	0.2	0.3	0	0	0.05	0.1	0.15	0.2	0
INTEREST	US	0	0.05	0.10	0.15	0.2	0.25	0.3	0	0.05	0.1	0.15	0.2	0.3	-0.05	-0.1	-0.2	-0.3	-0.4	-0.4	0

In the algorithm, the analyst assigns the effect of actions $\boldsymbol{a_1^v}$ through $\boldsymbol{a_n^v}$ in set of actions $\boldsymbol{A_i^v}$on each component of $\boldsymbol{v}$ as such:

$$A_i^v = [A^f, A^h, A^i] = \left[[a_1^f \quad a_2^f \quad \cdots \quad a_n^f], [a_1^h \quad a_2^h \quad \cdots \quad a_n^h], [a_1^i \quad a_2^i \quad \cdots \quad a_n^i]\right]$$

Each move also has an effect on Actor A's resource deployment for each object. For example, if Actor B takes military action, this will deplete Actor A's military and economic resources. Each move is given a utility rating that represents how well the move achieves the actor's goals considering its identities and priorities. The utilities for the individual components (which correspond to the f, h, and i axes) are summed together to yield overall utility (U_t) for each move. Each utility component is an inverse function of the object's distance from zero along the related axis; this means higher threat levels have lower utility. (.01 is added to the threat level so that its minimum is insignificant, but non-zero.) The inverse of the threat level is multiplied by the object weight (r, c, or l) so that the utility components for the prioritized axes will have more bearing on the move's overall utility U_t :

$$U_t = \left(\frac{r}{|f_t| + .01}\right) + \left(\frac{c}{|h_t| + .01}\right) + \left(\frac{l}{|i_t| + .01}\right)$$

However, the model checks to ensure that the move selected does not cause more than a specified decrease in the deployment of resources prioritized by the actor's identity. The permissible degree of decrease is calculated as the standard of deviation of all the move outcomes for the axis of interest multiplied by the weight (w) of the associated identity subtracted from 1 as such:

$$[\mathrm{v}_t + \alpha_t^x < \bar{v}_t - \sigma_{v_t}^2(1 - w)]$$

where α_t^x is the effect of the move, v_t is the object position on the relevant axis at time t, and $\bar{v}_t$ is the average object position on the relevant axis after all of the moves. So, for example, high weights for the realist identity will cause the actor to have little tolerance for losses along the fear axis. A moderately liberal actor will tolerate moderate losses along the interest axis. If the above expression evaluates as false, then the model selects the best move for which this expression evaluates as true.

When each actor has selected their best move, the object's new location for axis v at time t+1 is determined by summing the effects on v of the selected moves and adding that sum to the object's old location at time t such that:

$$v_{t+1} = v_{t+1}^{actor\_A} + v_{t+1}^{actor\_B} - v_t$$

where $v_{t+1}^{actor\_A}$ is the object's location only after Actor A's move, $v_{t+1}^{actor\_B}$ is the object's location only after Actor B's move, and v_{t+1} is the location after both moves (i.e., after their moves are "resolved").

Altogether, the move selection algorithm is described below. It is written like a Python function, but the variable names are described in mathematical notation and indexes so the reader can follow the variable paths throughout the loops more easily. The model iterates this function 24 times for each object to shift each location from $\boldsymbol{l_0}$ to $\boldsymbol{l_{t+24}}$ at the end of the simulation.

```
actors = {Actor A, Actor B}
$\boldsymbol{w}^{actor\_A} = [r, c, l]$
$\boldsymbol{v_t} = \{f_t, h_t, i_t\}$
$\boldsymbol{l_t} = [f_t, h_t, i_t]$
guardrails=“on”        #this can be turned “off” to allow the actors to
                       select moves without considering their effect
                       Actor A’s resources.

for $\mathrm{v}_t$ in set $v_t$ :
    if $[\mathrm{v}_t + \alpha_t^x < \bar{v}_t - \sigma_{v_t}^2(1-w)]$ == TRUE:
        if guardrails==“on”:
            $g = 1$
        else:
            $g = .001$
        $\varepsilon_v = 100g$
    else:
         $\varepsilon_v = 1$
$A_i^v = [A^f, A^h, A^i] = [[a_1^f \quad a_2^f \quad \cdots \quad a_n^f], [a_1^h \quad a_2^h \quad \cdots \quad a_n^h], [a_1^i \quad a_2^i \quad \cdots \quad a_n^i]]$
moves = [ ]
for actor_x in actors:
    $\mathrm{V}_t^{actor\_x} = [\ ]$
    for v in range(len($v_t$)):
        $A^{\mathrm{v}} = A[\mathrm{v}]$
        $\mathrm{V}_t^{\mathrm{v}} = [\ ]$
        for $a_t^{\mathrm{v}}$ in $A^{\mathrm{v}}$ :
                $\mathrm{V}_t^{\mathrm{v}}.\mathrm{append}(\mathrm{v}_t \ + a_t^{\mathrm{v}})$  #adds $\mathrm{v}_t$ to set $\mathrm{V}_t^{\mathrm{v}}$
        $\mathrm{V}_t^{actor\_x}.\mathrm{append}(\mathrm{V}_t^{\mathrm{v}})$   #adds $\mathrm{V}_t^{\mathrm{v}}$ to $\mathrm{V}_t^{actor\_x}$
    $U_t \ = [\ \ ]$
    for $n$ in range(len($A$ [1]):
        $U_t^n = \left(\frac{r}{|\mathrm{V}_t^{actor\_x}[0][n]|+.01}\right) + \left(\frac{c}{|\mathrm{V}_t^{actor\_x}[1][n]|+.01}\right) + \left(\frac{l}{|\mathrm{V}_t^{actor\_x}[2][n]|+.01}\right) * \varepsilon$
        $U_t \ .append(U_t^n)$
    $\alpha_t^{best\_x} = U_t \ .\mathrm{index}\left(min(U_t \ )\right)$   #finds the index of the best move
    $V_{t+1}^{actor\_x} = \left[\left[V_t^{actor\_x}[0][\alpha_t^{best\_x}]\right], \left[V_t^{actor\_x}[1][\alpha_t^{best\_x}]\right], \left[V_t^{actor\_x}[2][\alpha_t^{best\_x}]\right]\right]$
    $\mathrm{moves.append}(V_{t+1}^{actor\_x})$
$l_{t+1} = [\ \ ]$
for v in $v_{t+1}$:
    $\mathrm{v}_{t+1} = \mathrm{moves}[0][0] + \mathrm{moves}[1][0] - l_t$
    $l_{t+1}.\mathrm{append}(\mathrm{v}_{t+1})$   #this makes $l_{t+1} = [f_{t+1}, h_{t+1}, i_{t+1}]$
return $l_{t+1}$
```

In the Excel model, this algorithm is written in matrix form and repeated for each time *t* from *t=0* until *t=24*. This allows the actors to select 24 independent moves for each object. Together, these moves are considered the actors' courses of action, which are the model's fundamental outputs. They are listed in the outputs page, which also contains graphs illustrating how Actor A's resource deployment changed over time.

Future Research

CAS represents an early attempt to integrate complexity, algorithmic agent-based models, and international relations theory into an analytical tool. Complexity theory and its expanding array of statistical and computational methods provides rich accounts the world's interactive and emergent relationships, such as networks (e.g., see Slaughter 2009), political ecologies, and dynamic systems. Complex adaptive systems and insights from evolution and genetic adaptation could empower analysts to study how states adapt their processes to accommodate new circumstances, such as changes in national power or new domestically-driven political mandates. Moreover, when one considers that states often act and react to one another's decisions in ways that are intersubjective, it seems quite beneficial to simulate that states autonomously adapt their specifications over time to optimize their security outcomes in the context of a broader security environment. The lab will continue to refine and expand the model as new data and theories emerge. However, states' iterative and relative decision-making models require further theoretical description and data collection for validation, but deriving a generalized model such as CAS is an important interim step to advance the strategic modeling enterprise.