

A Dynamic Social Network Software Platform for Counter-Terrorism Decision Support

Richard M. Adler, Member IEEE Computer Society

Abstract—DynNetSim is a dynamic network software platform for modeling and analyzing social and other networks as they evolve over time. DynNetSim combines stochastic modeling with a behavioral simulation framework that synthesizes system dynamics and computational agent paradigms. The resulting framework provides a holistic depiction of networks responding to influences of environmental forces, trends, and disruptive events. In social networks comprised of intentional (goal-directed) entities such as terrorist groups, such responses may encompass opportunistic and adaptive behaviors. DynNetSim also enables exploration of the likely impacts of prospective strategies to change networks, such as attacking adversaries' social networks or reducing vulnerabilities in our own critical infrastructure systems. DynNetSim can enhance diverse counter-terrorism activities including intelligence analysis; critical infrastructure protection; and preparedness planning.

Index Terms—dynamic social networks, “what-if simulation, counter-terrorism preparedness, homeland security, intelligence analysis

I. INTRODUCTION

GRAPH networks are being used increasingly to study social organizations and other complex systems [1]. Graph networks are composed of *nodes* and *arcs* (or *links*). Nodes can depict, variously, individuals, organizations, locations, material things, concepts, or other data. Arcs depict relationships or flows between nodes. Typical relationships include personal acquaintance, trust, or organizational ties; e-mail, conversations, or other interactions, or connectivity across communication networks. Flows depict transfers of information, control, funds, or materiel. Networks can be analyzed with respect to numerous statistical metrics, such as the number of direct connections per node (degree); the number of indirect (or bridging) arcs (betweenness) per node; and the shortest paths between pairs of nodes (closeness).

Social networks offer important insights into the structure of organizations and their internal patterns of communication and collaboration [2]. Network analysis highlights structural features that can signal potential performance problems, either at the individual or organizational level, or systemic vulnerabilities. The following list illustrates typical diagnostic targets for social network analysts:

- Individuals who are isolated, and thus, probably not contributing significantly to an organization
- Individuals who are over-burdened with connections, indicating potential bottlenecks to group effectiveness

- Groups within a network whose inter-connections are sparse (or “brittle”), mediated by only one or several individuals
- “Hub” individuals, whose connections (and influence) are so extensive that their departure would seriously compromise organizational capabilities or performance

From an internal perspective, these network features represent targets for remediation efforts by organizations, such as reassignments; reorganization; or training, communication and incentive programs. Resolving these flaws can improve resiliency and performance substantially. From an external perspective, network anomalies represent opportunities for adversaries to exploit and attack for strategic advantage. In both cases, network analysis focuses attention and resources on critical weaknesses to address.

Until recently, social network analysis has been confined largely to static models. Static methods entail obvious limitations in tracking changes in network structure over time. They also offer minimal utility in studying the underlying dynamics driving network evolution. Finally, they cannot support future-directed “what-if” analyses, which are critical for understanding threat consequences and network vulnerabilities. What-if analyses are also indispensable to assessing alternate interventions aimed at impairing adversaries' networks or strengthening one's own networks.

This paper describes DynNetSim, a software platform for analyzing networks from a dynamic perspective. DynNetSim combines two previously disparate approaches to modeling networks dynamically – stochastic methods and behavioral simulations using computational agents. DynNetSim also addresses another key shortcoming in current dynamic models, which is their inability to characterize the environments in which networks operate, and interactions between networks and their environments. Incorporating such context is critical to analyzing terrorist groups, which display highly opportunistic and adaptive behavior patterns in response to evolving real-world conditions and events. The next section reviews critical applications of network analysis to counter-terrorism preparedness. The following section describes the capabilities and architecture of DynNetSim. This discussion illustrates DynNetSim features using a pilot dynamic network model for counter-terrorism decision support.

II. NETWORK ANALYSIS AND COUNTER-TERRORISM

A. Relevance

Social network analysis is used increasingly by military and intelligence agencies to develop tactics and strategies against terrorist networks and insurgent groups [3]. For example, U.S. Army Intelligence officers facilitated the capture of Saddam Hussein by developing a social network that traced Saddam's political, tribal, and family linkages [4]. This analysis helped the Army focus their intelligence gathering, surveillance, and search efforts to track down Saddam on a manageable set of individuals most closely tied with him recently or in the past.

It is widely recognized that terrorists, particularly transnational groups such as Al Qaeda, organize in networks composed of smaller groups called cells. Networks are non-hierarchical, and often geographically dispersed, with irregular topologies. Cells are largely independent from one another. Communication outside of cells is infrequent and often indirect or Web-based, minimizing use of standard telephony and e-mail channels susceptible to monitoring. Terrorist groups are also dynamic: their structures, objectives, and operating methods mutate with changes in leadership or socio-political conditions and experiential learning.

Terrorist (and insurgent) network structures and operations are obviously designed to impede detection, infiltration, and interdiction. As a result, counter-terrorism intelligence analysts must sift through volumes of data to reconstruct movements and associations of individuals of interest and flows of information, funds, and materiel. These patterns drive further inferences about organizational structure, goals and objectives, and key relationships such as trust and control. The process is highly iterative, and requires a complex combination of intelligent search, data mining, inference, and intuition. Network analysis offers a natural framework for aggregating, filtering, visualizing, and assessing intelligence data to support this process. [Note: network analysis is not claimed to "magically" uncover previously unknown terrorists; rather it offers a framework for fusing and visualizing intelligence that facilitates (semi-automated) identification of anomalous patterns merit further investigation and analysis.]

Network analysis also provides an important defensive tool for critical infrastructure protection and domestic preparedness. The National Strategy for Homeland Security identified thirteen critical sectors: agriculture and food; water; public health; emergency services; government; the defense industrial base; information and telecommunications; energy; transportation; banking and finance; chemicals and hazardous materials; and postal and shipping [5]. Each sector can be depicted as a (non-social) network whose individual nodes and arcs and overall topology manifest vulnerabilities and entail consequences from attacks.

Potential consequences of attacks, or natural disasters, are closely related to a node's position in, and influence on its embedding network. For example, the damage caused to the Port of New Orleans from Hurricane Katrina highlighted the

importance of the Port as the gateway to the Mississippi River and the considerable commercial activity that waterway supports. Direct damages to the port, while serious, represented only a fraction of the net economic harm caused from the associated disruptions of river-based traffic.

This example illustrates the utility of network analysis in homeland and national security strategies, most notably in emerging doctrines such as network-centric operations [6] and effects-based operations (EBO) [7]. These approaches, derived from classic systems thinking concepts [8], recognize that vulnerabilities and consequences must be assessed not simply in light of direct threats to particular targets (nodes or arcs), but must encompass indirect and cascading effects that can propagate throughout networks. Thus, DARPA and other DOD agencies are funding development of network-based "system of systems" frameworks to assess the collective effects of prospective diplomatic, military, information operations and economic (DIME) actions in terms of the adversary's political, military (air, land and sea), economic, social, information and infrastructure (PMESII) systems [9].

Terrorist and insurgent adversaries clearly recognize the utility of the DIME/PMESII perspective for planning and waging asymmetric warfare. This means that Homeland Security authorities should perform network-based analyses of national vulnerabilities as a means of anticipating terrorist or insurgent targeting strategies and tactics. Such risk assessments and gap analyses serve to focus efforts to develop remediation and preparedness measures. Portfolio management techniques can then be used to develop effective investment strategies to minimize risks to critical networks.

B. Rationale for Dynamic Networks

A static network depicts an organization or system at a particular instant. A dynamic model characterizes a network's evolution over time. By analogy, consider trying to understand a complex situation by viewing one or more photographs vs. a movie. It may be possible to reconstruct relevant changes by analyzing the differences between static "snapshots" of a network at two different instants. However, a dynamic network embodies an explicit model that projects that network's evolution. That is, change is *extrinsic* to static networks whereas change is *intrinsic* to dynamic models.

This distinction is a material one. In effect, a dynamic network's intrinsic behavioral model provides an inherent *theory* for that network. First, it constitutes a framework for explaining observed modifications over time – why the network changed the way it did. Second, a dynamic model enables predictions of future states of the network – how the network can be expected to evolve forward from its current state. Third, a dynamic model provides a baseline for continuous refinement; when observations depart from projected results, the model can be reviewed and discrepancies can be diagnosed and corrected.

The predictive nature of dynamic analysis has critical implications for decision support. Static network analysis offers *situational awareness*, or visibility into current

(network) status. Accurate status knowledge obviously facilitates identification of threats and vulnerabilities. As such, situational awareness constitutes a precursor to action: leaders can respond by devising strategies to change the current situation to their advantage, including defensive or remedial to improve security of our networks, or offensive measures to impair or attack adversaries' social networks. However, situational awareness fails to contribute materially to the *processes* of formulating, testing, or selecting courses of action; at best, it provides quality inputs to those processes.

In contrast, dynamic models can actively enhance or enable decision-making processes. Specifically, they enable the projection and comparative analysis of the costs, risks, and likely consequences of prospective interventions to strengthen and defend our networks or to attack our adversaries'. Comparative analysis, particularly across alternate scenarios of future conditions or attack variants, helps identify *robust* or *resilient* strategies. As such, dynamic networks offer a level of decision support that static networks cannot.

C. Network Dynamics

Two types of features may change over time within a network – structural and characteristic. Structural changes consist of modifications in a network's topology, comprised of its population of nodes and the arcs that interconnect them. New nodes may appear and join the network. Existing nodes may die or depart. Nodes may also subdivide or fuse, for example, if terrorist group splinters into factions (or one business merges with or acquires another). Similarly, a network's population of arcs typically evolves over time. New links form. Old links are severed or "switched" to new target nodes, such as when a company changes supplier partners.

Less dramatic, but equally critical, are characteristic changes. Nodes may change properties (e.g., age, size, objectives, and capabilities), while arcs may change intensity, as relationships strengthen or decay.

Stochastic models describe network dynamics from a statistical perspective [10]. Specifically, populations of nodes and arcs are generated and evolve over time according to statistical distributions and rules. DynNetSim's stochastic model will be described in the next section to illustrate this approach. Furthermore, Monte Carlo techniques can be applied to study the sensitivity of network evolution to differences in these distributions [11]. Current stochastic models focus on structural rather than characteristic network properties, although some types of characteristic dynamics are amenable to stochastic modeling.

The alternate approach to dynamics consists of explicit behavioral simulation models, such as computational agents or complex adaptive systems (CAS) [12]. Agent-based models treat nodes as largely independent and autonomous entities. Agent behaviors are often modeled in terms of conditional rules, such as stimulus-response or game theoretic decision rules: agents "sense" changes by applying their internal rules. This process starts by evaluating rule antecedent ("if") clauses. Rules whose antecedents are satisfied (true) are

triggered, by executing the consequent ("then") clauses, to bring about appropriate behavioral responses.

Agent behaviors can bring about internal (characteristic) alterations, structural network changes, or both. For example, changes in agent properties (internal state) may induce them to strengthen or weaken the intensity of existing arcs, or to create, sever, or switch arcs.

What kinds of changes can be sensed depends on the agent modeling paradigm. Cellular automata generally do not depict environments explicitly; instead automata are limited to sensing and responding to changes in their peers, which typically consist of neighboring agents in a uniform grid (That is, network arcs are defined implicitly by grid positions.) [13]. Agent models such as ant-inspired simulations depict environments explicitly, but generally as passive "terrains" to be traversed and manipulated by agents [14]. For example, ants can deposit "pheromone trails" as they move, which other ants may sense, follow, and reinforce. At the other extreme, DynNetSim treats the environment as a fully active, dynamic container, which may evolve independently of, and interact with, the agents it contains.

The behavioral dynamics of agent networks tend to be highly domain-dependent, driven by the nature (semantics) of constituent nodes and arcs. Environmental contexts – namely situational forces, trends, and disruptive events also exert significant behavioral influences on real-world networks.

This appears to particularly true in social networks. Individuals and organizations are intentional (i.e., goal directed) entities. They have beliefs and objectives, and develop courses of action to achieve those objectives. Equally important, they sense changes in their environments in addition to detecting actions of others, and, if necessary, adapt their strategies, behaviors, and possibly their goals in response to those perceptions.

A simple example illustrates the power of environmental influences in modeling terrorism. Analysis of terrorist perpetrators has shown that moderate Muslim individuals can grow sufficiently angry at actions taken by Western nations that they adopt more radical views and become susceptible to recruitment by terrorist groups.

Both stochastic and computational agent approaches try to model how networks change over time. We believe that agent-based approaches, particularly extended agent models that enable explicit modeling of environmental factors and interactions, hold considerably more promise in explaining – and predicting – why those changes occur in real-world social networks. Given the demonstrated opportunism, adaptiveness, and lethality of terrorists and insurgents, the need to integrate dynamic network modeling into counter-terrorism analysis and preparedness planning is urgent.

III. DYNAMIC NETWORKING FRAMEWORK (DYNNETSIM)

A. DynNetSim's Static Network Model

DynNetSim is a dynamic (social) networking framework being developed for counter-terrorism analysis and decision

support. DynNetSim provides a rich set of network modeling, behavioral simulation, visualization, and analysis capabilities.

DynNetSim's (static) network ontology consists of a hierarchy of object-oriented agents. Each class of agents, or entity type, encapsulates descriptive attributes and behaviors. The hierarchy serves to organize entity types via an intuitive containment metaphor: parent agents types provide an embedding context (or are composed of) child types. The hierarchy also defines an ordering that drives the simulator's sequence for invoking agent behaviors during each cycle.

DynNetSim's fundamental representational unit is a *Scenario*. Scenarios can depict actual or hypothetical situations (e.g., a specific emerging threat vs. an archetypal biochemical attack). A Scenario contains one or more Environments, which contain one or more Networks, and zero or more Events. An Environment for counter-terrorism analysis might include attributes such as global oil production and consumption, the size of immigrant populations, and trends such as annual rates of oil production capacity, economic growth, and immigration. Environments contain Forces, which track phenomena such as anti-American sentiment in the Muslim world, international cooperation on counter-terrorism initiatives such as intelligence sharing and transportation security. Events depict disruptive occurrences, such as a change in political leadership or a natural disaster.

A DynNetSim Network is composed of Nodes and Links (arcs). A simple social network might simply be composed of the single node type Person. More elaborate models might define heterogeneous Node types, such as Persons and Groups. The pilot DynNetSim counter-terrorism network model encompasses four distinct Node types: Person, Group, ThreatActivity, and Target. Groups depict terrorist cells, which are essentially sub-networks of Persons. The ThreatActivity node type models terrorist or insurgent actions, such as planning, training, or weapon construction, which contribute to attack threats. This ontology is depicted in Figure 1.

Links in a DynNetSim Network can connect Nodes within or across Types. Thus, Person nodes can be linked to one another, or to Groups, Activities, or Targets. A Link is directed; it has a unique source and target Node.

The semantics of Links in a heterogeneous network pose a major design challenge. Intuitively, Links should support semantics corresponding to different kinds of relationships between pairs of nodes, particularly if those nodes are of different types. For example, inter-personal links reflect frequent contact or control, whereas inter-group links depict collaboration and resource sharing and person-group links represent membership. DynNetSim supports this requirement by defining a single type of Link that transparently manages multiple, independent relationships or flows (and their intensities) using internal tables. In short, DynNetSim trades off the design complexity of managing diverse arcs between pairs of nodes in favor of a single composite Link that encapsulates relationship maintenance. Links also contain an Age attribute that tracks the duration of the connection

between its associated pair of Nodes. DynNetSim allows cyclic graphs; whether they occur or not depends on the domains-specific semantics for Links in a given network.

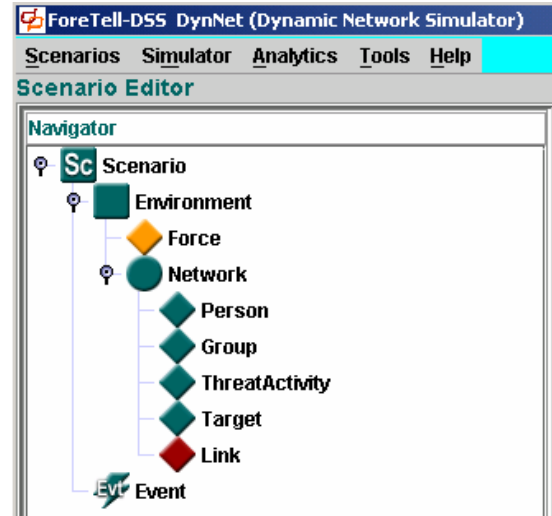


Fig. 1. Ontology for Pilot DynNetSim Network Model

B. Building Networks in DynNetSim

Networks in DynNetSim can be populated via three methods. Populations of Nodes with a Network can be generated statistically; created individually (as “named” entities), or generated and created. Links can be defined to connect Network Nodes in three corresponding modes: statistical generation; explicit creation, or by combining both methods.

DynNetSim provides a graphical user interface (GUI) called the Scenario Editor to support the definition of Scenarios. Users create instances of Scenario, Environment, Network, etc, and then populate them using attribute editors.

Individual “named” Nodes and Links making up a Network can be defined from scratch, imported from external files or databases, or copied (and reused) from DynNetSim's library. For example, users can create a Link instance and then manually specify its source and target Nodes and any extant relationships using standard GUI editor controls.

Alternatively, Nodes and Links can be generated automatically, driven by from attribute values defined in the Network entity. In this case, Nodes and Links are created over time, when the DynNetSim simulation engine executes the Scenario. DynNetSim employs declarative specifications for a stochastic network adapted from [10]. Users specify rates of Node creation via standard statistical distributions. For example, Nodes might be created at a rate based on a normal distribution with a mean value of six and a standard deviation of 2. Given this specification, the DynNetSim simulator takes a sample (random deviate) from the distribution, creates that number of Nodes, and adds them to the network each cycle.

Similar stochastic specifications drive the management of Link populations, in terms of distributions for rates of Link formation, decay, and strengthening. Other Network entity

attributes specify link “rules” that dictate how Nodes links to one another, and how those rules are assigned to new Nodes. Rules can be assigned uniformly, randomly, or by a hybrid distribution based on percentage quotas.

Examples of link rules include connecting with Nodes randomly; to Nodes that are the oldest (or youngest); or to Nodes with maximal or minimal Degree (number of arcs). The maximal degree rule corresponds to a dynamic in which the rich get richer – i.e. the most “popular” Nodes grow connections faster than any others. Given a source Node and link rule assignment, DynNetSim sorts the pool of available target Nodes into equivalence classes, and randomly samples from those sets until the quota is reached. Custom link rules can be created for domain-specific Node attributes.

These specifications dictate how Nodes and Links are created over time. Other rules specify further forms of network dynamism. Are Nodes allowed to die, and if so on what basis – random selection, old age, or minimal degree? Are Links static or dynamic? DynNetSim supports several forms of Link mutability. Links may decay spontaneously. Nodes can create new Links over time, or can “rewire” by dropping Links to some Nodes and forming Links to others. Node link assignment rules can change. Finally, Nodes can merge, resulting in a union of their Links

Stochastic models are valuable because they specify interesting network dynamics quickly and easily. The drawback is that these dynamics are relatively coarse, and driven by a restricted set of (domain-independent) structural attributes, such as age, degree, and fitness. As such, stochastic approaches are typically more useful for modeling non-intentional systems, such as critical infrastructure networks, Internet servers or content links (URLs); academic citations; biological systems; and the like.

C. *DynNetSim's Behavioral Dynamics*

DynNetSim supports a second form of network dynamics. This behavioral framework explicitly incorporates networks' environmental contexts as well as goal-directed and adaptive behaviors of terrorists, insurgents (and national and homeland security forces). As such, this behavioral model has substantially more explanatory and predictive potential than stochastic models. The downside is that specifying and validating semantically rich behavioral dynamics for complex real-world social networks is obviously a difficult and extended task. However, such efforts are necessary because of the unacceptable costs of failing to address threats adequately and effectively.

Numerous approaches (and supporting software tools) exist for behavioral modeling, including Monte Carlo, system dynamics, CAS, rule- and event-based techniques. Broadly speaking, each method focuses on a particular dynamic driving change in the system: statistical variation; causal interactions; adaptive behaviors, etc. These methods have proven their effectiveness for their target focus. However, most real-world systems manifest multiple dynamic forces, often interacting with one another. “Uni-modal” approaches

are not as efficacious at depicting multiple dynamics. For example, system dynamics is adept at depicting situational forces via networks of interconnected “stocks” and “flows”, including feedback loops, latencies, and other mechanisms. This approach is less apt for variable numbers of entities displaying autonomous, intentional behaviors. CAS models excel at modeling populations of autonomous agents, but are cumbersome for depicting complex interactions among environmental forces. Consider, by analogy, taking only a hammer or a saw to a construction project, or taking pictures using a film that only responds to blue or orange light: uni-modal models can manifest serious omissions and distortions that expose their interpreters to unacceptable risks. DynNetSim addresses this problem by integrating complementary, previously standalone techniques into a unified framework: each dynamic that acts within (or on) a network is depicted via an appropriate technique, but shares common model state.

DynNetSim's “multi-modal” framework currently models the following situational dynamics:

- Trends: slowly varying, predictable changes in entity attributes, such as rates of population or economic growth.
- Events: disruptive changes tied to specific points in (simulated) time. Events can alter one or more attribute values for any entity type, reflecting spontaneous, exogenous changes. For example, Israel launches attacks on Palestinian terrorists in Gaza in Month 7.
- Causality (hard-coded): system dynamics productions [8]. This type of causal rule transparently propagates effects from changes in entity attributes. For example, increases in demand for oil increases its price, which increases income for oil-producing nations.
- Causality (soft-coded): causal rules defined on entity types such as Forces. These rules expose causal parameters, which users can tailor to fit their situation (e.g., latency, duration, feedback, and magnitudes of effects). For example, an increase in anti-Western sentiment (Force) causes increases in recruitment rates by terrorist group.
- Agent behaviors: process-oriented activities such as terrorist groups planning, funding, and preparing to stage attacks. Activities typically involve creating schedules, procuring and moving resources, and constraints.
- Agent decision-rules: CAS productions which, when triggered, can modify Agent behaviors as well as attributes. Example “meta-behaviors” include switching current tactics or targeting priorities in response to changing conditions.
- Statistical variation: DynNetSim provides a stochastic dynamic model internal to the Network entity type. Users can also perform Monte Carlo simulations on entire Scenarios. A Monte Carlo utility allows users to specify value distributions for selected entity attributes (inputs) and then run a specified number of Scenario “trials” in a batch simulation mode.

This “inventory” of behavioral building blocks can be exploited to model: how terrorists organize, communicate,

plan, fund, and execute operations; how terrorists recruit and exploit media attention; what kinds of patterns to look for in intelligence data to indicate these activities; and how strategies to capture “key” individuals, impede operations, or interdict attacks are likely to perform (or be countered by terrorists).

D. DynNetSim Architecture

DynNetSim was implemented using ForeTell, a generalized software platform for building decision support systems [15]. DynNetSim leverages ForeTell’s core modeling, behavioral simulation, and analysis capabilities. DynNetSim extends ForeTell with a domain-specific ontology tailored to dynamic network analysis: Environment, Force, Network, Link and NodeInterface. These entity types, their attributes, and dynamic behaviors all constitute domain-specific content that ForeTell manages and applies for DynNetSim.

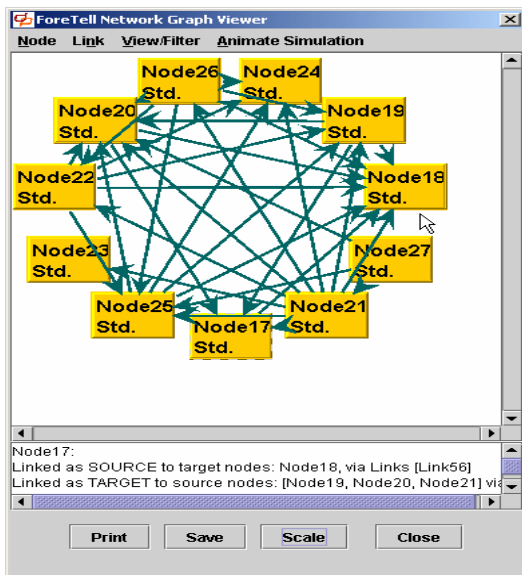


Fig 2. DynNetSim Network Viewer

For example, the Network entity type implements the stochastic network dynamics, while Forces implement the causality behavior pattern described above. The sole functional extension required for DynNetSim was to implement a custom viewer for visualizing network topology and Node and Link contents (cf. Figure 2), and integrate that display into ForeTell’s simulator user interface.

DynNetSim in turn, be extended to model diverse types of networks, including terrorist networks, critical infrastructure systems, and populations experiencing pandemic outbreaks.

Tailoring DynNetSim involves adding new entity types, attributes, and supporting behaviors to the core domain model. For example, an intelligence analysis solution adds an IntelData entity type to integrate raw intelligence data, and a TaskingOrder entity to manage requests for additional intelligence. The Link entity type is extended to maintain hypotheses, pointers to supporting IntelData items, and pointers to TaskingOrders for intelligence gathering aimed at corroborating assumptions or hypotheses or completing an

emerging intelligence picture. A DynNetSim variant to support preparedness planning adds Strategy and Activity entity types, to model prospective interventions (e.g., to defend against network threats). Supporting attributes and behaviors track activity costs, resources, schedules, and their projected impacts on Network Nodes and Links. Figure 3 summarizes DynNetSim’s layered architectural.

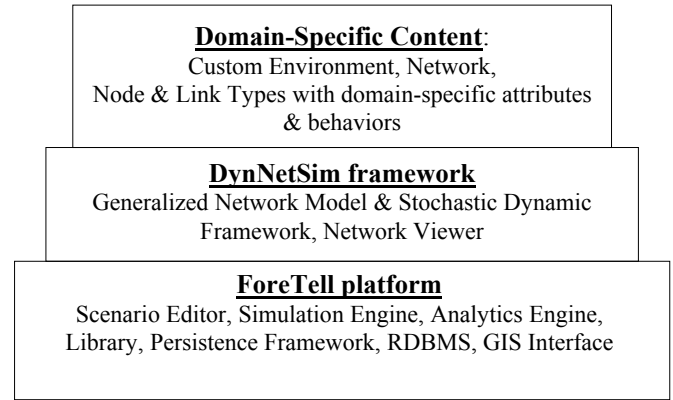


Fig. 3. DynNetSim’s Layered Architecture

DynNetSim’s functionality is driven by ForeTell’s underlying modeling, simulation, and analysis frameworks and supporting Graphical User Interfaces (GUIs):

Scenario Editor. ForeTell’s Scenario Editor provides a GUI for creating, editing, browsing and maintaining DynNetSim Scenarios. Users create Scenarios by “snapping together” and editing instances of desired entity types. Entities can be created from scratch, copied from a library of pre-defined components, or imported from data files or external databases. Creating, validating, and maintaining a suitable library of reusable Forces, Persons, Groups, etc., is a key prerequisite to ease-of-use, productivity, uniformity, and large-scale deployment. Users inspect and edit attribute values via standard GUI controls such as text boxes, sliders, lists, and tables. Users can also edit individual entity attribute values with metadata, to facilitate Scenario maintenance, sharing, and “sense-making”. Metadata includes comments, sources, and tags indicating fact vs. assumption and certainty level. To facilitate “what-if” analyses, Scenarios can be copied in their entirety, and then edited selectively to quickly define variant strategies or alternate assumptions about forces, trends, events, and entity. Such variants support situational extrapolation for hypothesis testing and outcomes analysis for validating intervention strategies. DynNetSim uses ForeTell’s Relational Database Management System (RDBMS) to store Scenarios persistently.

Simulation Engine. ForeTell’s Simulation Engine executes DynNetSim Scenarios, projecting the evolution of networks over time. ForeTell employs a hybrid discrete event agent-oriented simulator, which is extended with Event, system dynamics, and Monte Carlo overlays. At each simulated interval, the engine invokes the model’s entities in a uniform order (cf. Figure 4): any “timely” Events, the Environment, Forces, Network, and constituent Nodes. Each such entity

runs its type-specific behaviors, which include updating trends, carrying out processes, and executing decision rules. These actions potentially involve sensing internal and external state (Environment, peer Nodes, and Links) and responding according to their (intentional) patterns. The engine then propagates causal influences from productions triggered in the current cycle. As noted earlier, rules are found in domain-specific hard-coded rule sets (influence diagrams) or Forces, and are triggered by relevant entity attribute changes.

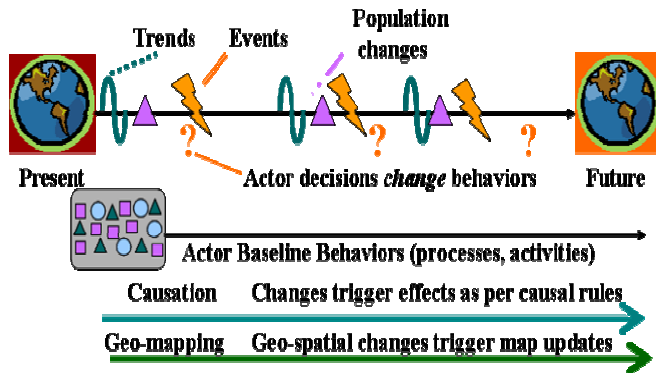


Fig. 4. ForeTell's hybrid behavioral simulation schedule

DynNetSim users can monitor and control executing Scenarios through ForeTell's "dashboard" style GUI, made up of controls, gauges and time series graphs. Users can suspend simulations to monitor situational metrics (e.g., cumulative injuries and costs) and to inspect specific entities. DynNetSim integrated network viewer displays the current network state, and allows users to inspect Nodes or Links.

Analytics Engine. As the Simulation Engine runs, it logs all changes in DynNetSim Scenario entity state to a database. ForeTell's Analytics Engine helps users access and reduce this mass of data to explore projected outcomes of individual Scenario. More importantly, users can compare outcomes across Scenarios involving competing strategies and/or alternate assumptions. Such differential analyses are vital for isolating relative strengths and weaknesses; uncovering unintended consequences; refining analyses or strategies, and identifying resilient strategies. Users can quickly generate summary analytics via a menu-driven, including tabular reports, time series and radar plots, and frequency histograms. ForeTell's analytics engine embeds open source math, graphics, and statistics libraries, allowing rapid extension to satisfy new analytic requirements.

Other Facilities. ForeTell provides further tools to support DynNetSim. An on-line help facility presents documentation on DynNetSim entity types and attributes. Scenarios and simulation logs can be imported and exported via open systems data exchange formats such as XML, CSV, and SQL. Curve fitting and liner interpolation utilities convert observational data into executable specifications for entity behaviors. ForeTell also describes DynNetSim's dynamics to users on demand, via behavior viewers and influence diagrams: given the number of "moving parts" such "transparency" of behavioral logic is critical for user

acceptance., particular when complex dynamics are

DynNetSim's layered architecture provides functionality and extensibility that would be difficult to develop from scratch in a dedicated dynamic network tool. The DynNetSim architecture also provides a supporting scenario-based methodology to guide network modeling and analysis.

The most notable weakness of DynNetSim's architecture is that domain-specific network models are not purely declarative. DynNetSim requires explicit programming and compilation to specify custom Node types and attributes, and domain-specific behaviors (e.g., causal productions for Environment-Node interactions and Node-specific decision rules). That said, domain-specific extensions typically require no more than a few person days of effort, because of ForeTell's rapid development tools and pattern-based component frameworks.

IV. RELATED WORK

The literature on social networks analysis is extensive, but focuses predominantly on static models [2, 16]. Also, business applications far outnumber defense and security studies such as [17, 18]. Research on dynamic social networks is more recent and generally less advanced [19]. Key exceptions are dynamic multi-agent network models developed at CMU's Computational Analysis of Social and Organizational Systems (CASOS). CASOS researchers have used dynamic behavioral agent networks to study the resilience of terrorist networks to disruption by security forces and to predict growth of insurgencies and their likelihood of destabilizing national governments [20, 21, 22]. CASOS models focus primarily on agents and their interactions, paying less attention to environmental dynamics and interactions with agents than does DynNetSim. Social scientific agent models that attend more closely to behavioral interactions between agents and their environments focus more on aggregate agent populations than on their relational links and network topologies [23]. Other dynamic agent networks model large-scale energy markets [24] and outbreaks of pandemic disease [25], although this latter system uses stochastic methods rather than intentional agents.

The ForeTell platform underlying DynNetSim was previously applied to decision support for counter-terrorism preparedness [15]. This application models terrorist groups, potential attacks on anticipated targets, and prospective strategies to interdict would-be attackers or mitigate the effects of attacks. Strategies can be assessed and compared across multiple scenarios with respect to cost, risk reduction, and effectiveness metrics. ForeTell integrates a Geographic Information System (GIS) for visualizing evolving intelligence, attacks, and responses. (The next version of DynNetSim will provide an option to project Nodes and Links as another map overlay.) ForeTell is also being applied to assess and mitigate maritime security risks, using a portfolio management approach to identify robust counter-terrorism investment strategies [26]. Finally, ForeTell was used to

develop a pilot decision support system for pandemic preparedness (IDODSS). We intend to replace that system's initial "top-down" epidemiological model with a "bottom-up" agent-based epidemiological model using DynNetSim.

V. CONCLUSION

Static network analysis facilitates intelligence analysis by promoting data fusion, visualization, and situational awareness. DynNetSim's dynamic network framework enables modeling and analysis of environmental influences on social networks, and adaptive, goal driven behaviors of network members. DynNetSim provides a virtual environment for safely exploring the likely consequences of current terrorist activities, as well as practicing and learning about prospective counter-measures. As such, DynNetSim can reduce risk and improve confidence and consistency in counter-terrorism analysis, decision-making, and preparedness.

REFERENCES

- [1] A.-L. Barabasi, *Linked: The New Science of Networks*. Cambridge, MA: Perseus Publishing, 2002.
- [2] R. Cross and A. Parker, *The Hidden Power of Social Networks*, Cambridge, MA, Harvard Business School Press, 2004.
- [3] R. Goolsby, "Developing Social Science Based Applications Lessons From ONR," *Navy Enterprise Conference*, August, 2004. [Online]. Available: http://www.au.af.mil/au/awc/awcgate/navy/onr_soc_sci_04aug.pdf.
- [4] V. Hougham, "Sociological skills used in the capture of Saddam Hussein," *Footnotes (Newsletter of the American Sociological Association)*, Vol. 33, No. 6, July/August 2005. [Online] Available: <http://www.asanet.org/footnotes/julyaugust05/index.html>
- [5] Office of Homeland Security, *The National Strategy for Homeland Security*, July, 2002, pp. 42ff. [Online]. Available: http://www.dhs.gov/xlibrary/assets/nat_strat_hls.pdf.
- [6] A.K Cebrowski and J.J. Garska, "Network-Centric Warfare: Its Origins and Future," *United States Naval Institute Proceedings*, January 1998, [Online] Available: <http://www.usni.org/Proceedings/Articles98/PROcebrovski.htm>
- [7] P.K. Davis, *Effects-Based Operations: A Grand Challenge for the Analytical Community*. RAND Corporation. MR-1477-USJFCOM/AF. 2001. [Online]. Available www.rand.org/publications/MR/MR1477/.
- [8] J. Sterman, *Business Dynamics: Systems Thinking and Modeling for a Complex World*. Cambridge, MA: Irwin McGraw-Hill, 2000.
- [9] Defense Advanced Research Projects Agency (DARPA) *Integrated Battlefield Command (IBC) Program Solicitation Briefing*, 2005. [Online] Available: <http://www.darpa.mil/sto/solicitations/IBC/>.
- [10] R. Albert and A.-L. Barabasi, "Statistical Mechanics of complex networks," *Reviews of Modern Physics*, vol. 74, Jan 2002, pp. 47-97.
- [11] G.S. Fishman, *Monte Carlo: Concepts, Algorithms, and Applications*, New York, Springer, 1996.
- [12] J. H. Holland, *Hidden Order: How Adaptation Builds Complexity*, Reading, MA, Addison-Wesley, 1995.
- [13] C. Langton, *Artificial Life: An Overview*, MIT Press, Cambridge, MA, 1995.
- [14] E. Bonabeau, M. Dorigo, and G. Teraulaz. *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press. NY. 1999.
- [15] R. M. Adler, "ForeTell: A Simulation-Based Modeling and Analysis Platform for Homeland Security Decision Support". *Proceedings Second IEEE Conference on Technologies for Homeland Security*. Cambridge, MA. May, 2003.
- [16] International Network for Social Network Analysis, entry bibliographies and tool lists. [Online] Available: <http://www.insna.org>.
- [17] P. Fellman and R. Wright, "Modeling Terrorist Networks – Complex Systems at the Mid-Range." [Online] Available: www.psych.lse.ac.uk/complexity/Conference/FellmanWright.pdf.
- [18] D.B. Skillcorn, *Social Network Analysis via Matrix Decompositions*, in *Emergent Information Technologies and Enabling Policies for Counter-Terrorism*, R.L Popp and J. Yen (Eds.), New Jersey: IEEE Press, pp. 367-391.
- [19] R. Brieger, K. Carley, and P. Pattison, Eds., *Dynamic Social Network Modeling and Analysis: Workshop Summary and Papers*. 2003. [Online]. Available: <http://newton.nap.edu/books/0309089522/html/346.html>.
- [20] R. Popp, S. Kaisler, D. Allen, C Cioffi-Revilla, K. Carley, M. Azam, A. Russell, N Choucri, and J. Kugler, *Assessing Nation-State Instability and Failure*, IEEE Aerospace Conference Big Sky, MT, March, 2006.
- [21] Tsvetovat, Max & Carley, Kathleen. (2002). *Knowing the Enemy: A Simulation of Terrorist Organizations and Counter-Terrorism Strategies. CASOS Conference 2002, Electronic Publication, Pittsburgh, PA.*
- [22] Tsvetovat, Max & Carley, Kathleen. (2003). *Bouncing Back: Recovery mechanisms of covert networks. NAACOS Conference 2003, Electronic Publication, Pittsburgh, PA.*
- [23] J.M. Epstein and R. Axtell. *Growing Artificial Societies: Social Science from the Bottom Up*. Brookings Institution Press, Washington DC, 1996.
- [24] C.M. Macal and M.J. North, "Simulating Energy markets and Infrastructure Interdependencies with Agent-Based Models" May 2006. [Online] Available: <http://www.dis.anl.gov/CEEESA/EMCAS.html>
- [25] L. A .Meyers, M.E.J. Newman, and B Pourbohloul, "Predicting epidemics on directed contact networks," Working Paper 04-12-037, Santa Fe Institute, Santa Fe, NM. [Online] Available: <http://www.santafe.edu/research/publications/workingpapers/04-12-037.pdf>
- [26] R.M. Adler and Jeff Fuller, "An Integrated Framework for Assessing and Mitigating Risks to Maritime Critical Infrastructure," forthcoming in *IEEE Conference on Technologies for Homeland Security*, Woburn, MA, May, 2007.