Modeling Complex Socio-technical Systems using Multi-Agent Simulation Methods

Maksim Tsvetovat, Kathleen M. Carley

The study of complex social and technological systems, such as organizations, requires a sophisticated approach that accounts for the underlying psychological and sociological principles, communication patterns and the technologies within these systems. While a number of approaches to studying such systems have been made, non-trivial analytical models have proved to be intractable, and artificial life simulations generally do not operate on empirical data, are constrained to grid based movements, and represent cognition at a high level of abstraction. This results in an oversimplified view of the phenomena. We posit that creation of high-fidelity models of socio-technical systems requires the combination of analytical models with empirically grounded simulation. This results in complex simulation-oriented multi-agent systems that incorporate planning and learning algorithms, while built on an extensive model of social network phenomena and social-psychological findings. We demonstrate the power of this methodology by creating a multi-agent network model of covert networks (such as terrorist organizations). In doing so, we show that AI algorithms and multi-agent systems, combined with an analytic approach, create a generalizable and valuable simulation toolbox for studying complex social systems.

1 Introduction

Multi-agent systems can be effective tools for reasoning about human and group behavior. Their effectiveness is enhanced when the algorithms lead the simulated agents to behave as humans behave, rather than doing what is optimal for the task. Such systems are even more effective when the model's inputs are real data and the generated outputs are comparable to actual data files in the real world. Such systems can be created by combining sophisticated planning and learning algorithms with extensive knowledge of human behavior and underlying networks.

Social networks are defined as the set of relations such as authority and friendship that link people to people. Social networks enable and constrain communication in human systems – as well as agent communication in multi-agent models of social phenomena.

2 Representing Network Knowledge – the Meta-Matrix

Placing sophisticated agents in social networks is insufficient to produce social behavior. Instead, the network conceptualization needs to encompass core entities such as people, resources, knowledge and tasks, and the relations among these entities. This concept is called the *meta-matrix* (table 1). Originally proposed by Krackhardt and Carley [9] as a device for simultaneously reasoning about people, resources and tasks, the meta-matrix approach was extended to include knowledge and groups/organizations. By linking the social network (people to people) to other networks such as task flow (task to task), the agents in the social network are provided with a framework for accomplishing *tasks* - and the user can track the impact of such behavior on the underlying networks. Linking social and knowledge networks allows observation of changes in networks linked to organizational performance. From a simulation perspective, social agents could be used in a multi-agent simulation to act out the impact of changes in these networks on organizational performance.

Meta-matrix data for the entire organization affords a birdseye view of the organization at a point in time, as a static snapshot. Moreover, if collected in the real world, it provides a picture of the entire organization which is quite distinct from what a single person is likely to know. A set of such matrices over time represents the change trajectory for an organization. Boundedly rational individuals make decisions and operate in a climate of uncertainty, with incomplete and inaccurate knowledge of the world. In essence, they make decisions using their perception of the meta-matrix – their personal knowledge and their knowledge of other's relations (*transactive memory* [13]).

This distinction between the actual meta-matrix and the agent's perception is at the heart of the difference between social network analysis, agent modeling, and multi-agent network simulation. In traditional social network analysis, behavioral interpretations are drawn from the actual network, which is viewed as constraining and enabling behavior. In agent modeling systems, agents are designed to act optimally for the task at hand. In multi-agent network simulations, agents act in a boundedly rational fashion on the basis of their personal perception, emulating what people might do. In order to produce a high-fidelity model of a dynamic organization, it is necessary to simulate the world of the agents, imperfections and all. We instantiate this by affording every agent a *belief structure*, which essentially is the agent's private meta-matrix structure populated as the agent interacts with others, learns and performs tasks. In the simplest case, the meta-matrix contains binary values whose meaning is simply existence or lack of connection between two nodes. However, this can be extended with weights assigned to every edge. For example, edge weights in the interpersonal network can be interpreted as trust or frequency of communication, as opposed to the existence of a connection.

While studying real-world networks, and specifically covert networks such as terrorist organizations, data can be collected for each cell of the meta-matrix but it is very difficult to obtain a complete and accurate snapshot of the organization. A key dif-

Auszug aus: Künstliche Intelligenz, Heft 2/2004,
ISSN 0933-1875, arendtap Verlag, Bremen
www.kuenstliche-intelligenz.de
fon +49 421 34889-30 fax: +49 421 34889-31

	People	Knowledge and Skills	Resources	Tasks
People	Structural knowledge: command structures and relationships between agents	Knowledge Network: who has access to what knowledge	Resource Network: who can use what resources	Task Assignment: who does which tasks
Knowledge		Knowledge Precedence: types of skills that go together	Resource Skills: skills needed to use a resource	Skill Requirements: skills needed to accomplish a task
Resource			Resource Precedence: which types of resources go together	Resource Requirements: Which resources are needed to accomplish a task
Task				Task Precedence: the sequencing and precedence of tasks.

Table 1: Meta-Matrix of Organizational Knowledge

ficulty is discerning whether a change in the networks is due to better intelligence or actual change in the networks. Thus, analysis algorithms and simulation systems must be able to operate under uncertainty, provide confidence estimates for results, and approach the domain from a satisficing rather then optimization perspective.

3 Modeling Dynamic Networks

Based on the conceptual framework of multi-agent simulations, we have developed NetWatch, a multi-agent network model for reasoning about the destabilization of covert networks such as organized crime or terrorist organizations under conditions of uncertainty.

NetWatch is built to simulate the communication patterns, information and resource flows in a dynamic organizational network based on cognitive, technological, and task-based principles. In addition, the model is grounded using information about surveillance technologies and intelligence operations (e.g. [1]) and the covert networks (e.g. [2]). The process of gathering intelligence on an organization is simulated, enabling the evaluation of diverse heuristics and technologies for data gathering. Using NetWatch, the user can conduct a vulnerability analysis and examine potential emergent reactions of covert networks in response to attacks, and evaluate diverse destabilization strategies.

NetWatch agents are intelligent, adaptive information processing systems, constrained and enabled by the networks in which they are embedded. These networks evolve as individuals interact, learn and perform tasks. In greater detail, the multi-agent network paradigm is based on the following postulates:

- Agents are independent, autonomous entities endowed with some intelligence, though cognitively limited and boundedly rational.
- Agents and the networks in which they are embedded coevolve.
- Agents do not have accurate information about the world or other agents, limited by their perception.
- Agents can learn the state of the world through interaction.
- Agents can be strategic about their communication.
- Agents do not use predefined geometrical locations or neighborhoods.

Agents obtain information via interaction with other agents. The accuracy of an agent's perception of another decreases with the distance between them in the social network. This corresponds with empirical reality, where people's knowledge of each other decreases exponentially with the social distance between them [8].

3.1 Social Networks in NetWatch

In NetWatch, agents communicate on the basis of network membership. Agents learn of other agents outside their ego network via interaction with agents in their ego network and a process of introduction. Therefore, networks are represented as a directed graph structure representing the probability of communication (social proximity) p_{ij} between two agents a_i and a_j:

$$\begin{aligned} Net &= AG, P \\ AG &= a_i \ : \ Setofagents \qquad (1) \\ P &= p_{ij} \ : \ \forall a_i, a_j \in AG \end{aligned}$$

The directed nature of the graph *Net* allows the user to specify one-way relationships and chain-of-command relationships. While the formal network is generally pre-specified at the start of the simulation, the informal network evolves through interaction.

The agents do not have access to full information about the network, but rather every agent a_k can only access a probability vector $P_k = p_{ki}$ where p_{ki} is a probability of agent a_k communicating with all agents $a_i \in A$. Hence, each agent may only know who it may interact with or is close to, but does not know the complete interaction patterns of other agents. Each agent possesses a belief matrix that it uses to store any information it learns about interrelationships of other agents within the network. However this information is typically incomplete and inaccurate.

3.2 Agents in NetWatch

In keeping with cognitive science research, NetWatch agents representing humans are both cognitively and socially constrained [12]. Thus, their decision-making ability, actions, and performance depend on their knowledge, structural position, procedures and abilities to manage and traverse these networks. Each agent's perception of the meta-matrix consists of the agent's ego network (the set of agents it is directly connected to), its own knowledge, resources and task assignments, and is augmented by the agent's perception of other agents' ego networks, knowledge, resources and task assignments (Fig. 1, 1), or, rather, the agent's beliefs about them. An agent also tries to form beliefs about the networks of other agents (Fig. 1, 2)

The agents implement a layered behavior model, inspired by Rodney Brooks' *subsumption architecture* for robot control [3]. On the lower levels of control lie primitive communication behaviors (Fig. 1, 3), based on cognitive models of human communication.

Intelligent task-directed behaviors are facilitated by a hierarchical decomposition planner (Fig. 1, 4), adapted for goal-directed interaction and distributed task execution via delegation of subtasks. The planner is described in section 3.6.

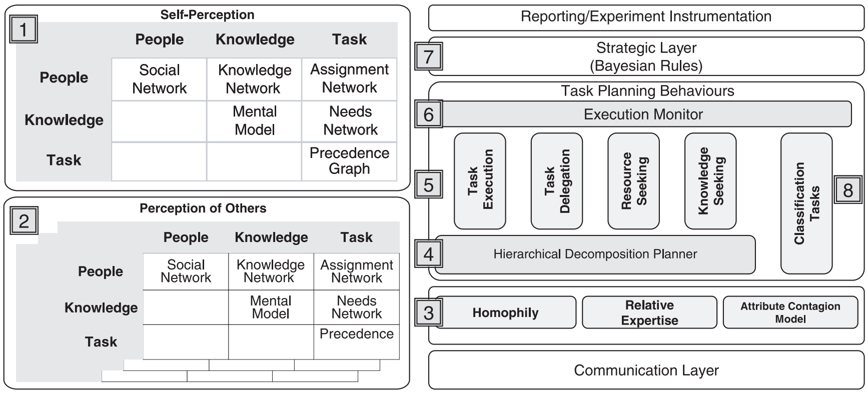

Figure 1: NetWatch Simulation Design

Execution of distributed tasks is monitored by an independent execution monitoring process (Fig. 1, 6), which watches results of delegated tasks, handles exceptions and triggers replanning in case of failure.

Finally, production rules (Fig. 1, 7) governs the agent's strategic reasoning, triggering tasks or high-level behaviors.

Such a layered architecture allows a social modeler to isolate strategic and tactical performance of the agents from their lower-level interaction, and build experiments where any one of the levels is manipulated.

3.3 Processes Governing Communication

The choice of a communication partner at every time period is based on two factors: *social proximity* of the agents and their *motivation to communicate*. Social proximity is defined as closeness of a relationship between two agents, scaled between 0 and 1, where 0 means "no relationship" and 1 is "very close relationship".

Motivation to communicate is computed on the basis of *homophyly* (relative similarity) and *need* (relative expertise). Empirical studies of human communication behavior suggest that, without any external motivation, individuals will spend about 60% of the time interacting on the basis of homophyly and 40% on the basis of need.

We defined homophyly to be based on a measure of relative similarity *RS* between agent *i* and agent *j*: the amount of knowledge that *i* and *j* have in common divided by the amount *i* shares with all other agents (including self), or

$$RS_{i,j} = \frac{\sum_{k=0}^{K}(S_{ik} * S_{jk})}{\sum_{l=0}^{I}\sum_{k=0}^{K}(S_{ik} * S_{lk})} \quad , (2)$$

where S_{ik} is 1 if agent *i* knows fact *k* and 0 otherwise.

In contrast, we defined need from a purely knowledge perspective. Relative expertise RE_{ij} defined as how much agent *i* thinks *j* knows that *i* does not know divided by how much *i* thinks all others know that *i* does not know, or

$$RE_{ij} = \frac{\sum_{k=0}^{K}((1-S_{ik}) * S_{jk})}{\sum_{l=0}^{I}\sum_{k=0}^{K}((1-S_{ik}) * S_{lk})} \quad . (3)$$

Agents operate on their beliefs about what the other agents know. Thus, their calculations can be inaccurate. However, as interaction progresses and agents learn more and more about each other, the accuracy of the agents' perception of the world increases.

3.4 Inter-agent Knowledge Exchange

Tracing back to its roots with the Construct [4] model, the NetWatch model is a cognitive model focusing on knowledge manipulation and learning. Each agent's knowledge is represented by a bit string. A value of 1 in the position *n* means that the agent knows fact *n* and the value of 0 means that it does not.

Both homophyly and need for information, as used in the knowledge exchange protocol are abstract measures and do not weight facts in regard to their importance to a task. More complex behaviors, such as task-directed information seeking, is accomplished using the planner (see section 3.5).

At the start of the simulation, the agents are endowed with some initial knowledge (typically within 2%–10% range), distributed randomly between agents or based on empirical profile of the organization.

To learn new facts, the agents execute the **Construct Knowledge Exchange Protocol**. For ease of description, we shall refer to the parties in knowledge exchange as Alice (agent $a_a \in AG$) and Bob (agent $a_b \in AG$).

1. **Determine who to communicate with:** Alice does this by evaluating *relative similarity* (Eqn. 2) or relative expertise (Eqn. 3) of every agent accessible through its social network (i.e. $p_{a,i} > 0 \forall\ a_i \in AG$ (Eqn. 1)). Then, a_a throws a dice that reflects the computed probability vector and picks an agent to communicate with (e.g. a_b).
2. **Determine what to communicate:** This is done by weighting information seeking vs. similarity-driven communication. If a_a is in information seeking mode, it chooses at random a part of the knowledge string that is not known and queries the agent chosen in step 1. In similarity-based communication, a_a chooses a part of the known knowledge string and sends it to that agent.
3. **Determine proper response:** On receipt of a query, a_b determines if it should answer by checking whether the sender of the query is part of its network. If yes, a_b sends a reply; otherwise he discards the message. If a_b does not know the facts requested, he may respond to a_a with a name of another agent („Clare") that may be better suited to answer the question, known as referential data. On receipt of knowledge, a_b determines if the knowledge is useful and whether it came from one of the agents in its network (and thus can be trusted), chooses some knowledge from its knowledge base and sends it in return.
4. **Update internal knowledge base:** On receipt of the reply, a_a determines the usefulness of the answer and uses that to update its internal knowledge of a_b ("Bob knows fact *n*") as well as its knowledge base ("I now also know fact *n*"). If a_a re-

ceives referential data, it uses that to update its knowledge of a_b ("Bob does not know *n*" and "Bob knows Clare") and a_c ("Clare may know *n*"). This may be followed by a query to Clare (a_c), which may or may not be answered, depending on the strategic position of a_c.

Clare may not have been originally a part of Alice's network, but now through Bob, Alice has learned about her existence. Thus, agents within the organization use referential data about each other to form an informal network.

3.5 Planning and Execution of Complex Tasks

The agents use a hierarchical decomposition planner to execute complex tasks that require coordination of knowledge and resources, as well as delegation of subtasks to other agents. In the meta-matrix representation, the agents possess a definition of their hierarchical task structures, an acyclic directed graph specifying precedence of tasks. The skill requirements and the resource requirements are also specified in the meta-matrix.

The planner starts with a top-level task, and finds its subtasks, resource and knowledge requirements. Then, the agent plans for each of the subtasks. If one of the subtasks requires knowledge or resources that the agent doesn't possess, the agent sends out request messages. The requests are handled by the Construct Knowledge Exchange protocol (see section 3.4).

Similarly, if the agent does not have the ability to execute a task, a task delegation message would be sent to an agent determined likely to execute the task. The delegated knowledge or tasks are then passed to the execution monitor process. The execution monitor keeps track of the currently delegated tasks, and handles failures and other exceptions.

Performance in planning tasks is measured as (a) time that it takes, (b) amount of re-work or replanning needed to complete the task, and (c) percentage of tasks that have failed.

3.6 Classification Tasks

One simple measure of organizational performance in NetWatch is based on the binary classification task. The task is represented by a vector of binary values. An agent can only access bits in the task vector that correspond to non-zero values in it's knowledge vector. The task is then decided by a "majority rule". An agent's decision accuracy is computed by taking a series of classification tasks and comparing the agent's decisions to "true answers" – computed by applying a majority rule to each task given complete information and access. Task performance is measured as a percentage of correctly decided tasks.

While appearing simplistic, performance in classification tasks have been shown [11] to correspond to organizational performance in real cases, thus making classification tasks a suitable substitute for more complex tasks for purposes of simulation modelling.

4 Modeling Terrorist Networks

For reasons of national security it is important to understand the properties of terrorist organizations that make them efficient and flexible. Based on this understanding, intelligence operators and decision makers can devise successful strategies to destabilize such organizations or curtail their efficiency, adaptability, and ability to move knowledge and resources. From the perspective of this paper, reasoning about terrorist organizations is a perfect example for demonstrating the power of tools that combine AI algorithms, multi-agent systems and social simulation.

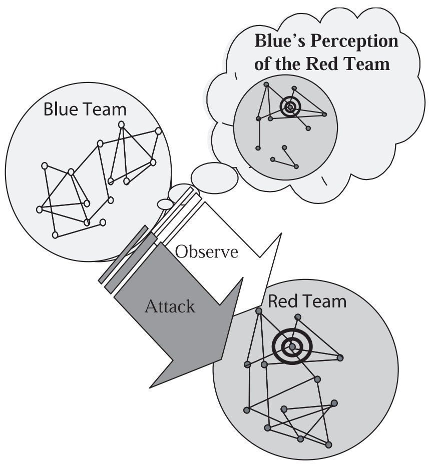

Figure 2: NetWatch Simulation Design

Terrorist groups are characterized as cellular organizations composed of quasi-independent cells and distributed command. This is a non-traditional organizational structure to which the lessons learned from the work on distributed and decentralized organizations apply. In particular, empirical studies suggest that due to pressures to operate efficiently while remaining covert, a network will emerge that combines massive redundancy with secrecy. Many of these networks are separated into cells that are sparsely interconnected with each other through the leaders. At the operational level, no clear hierarchy emerges from observation of these networks, other then the definite role of a cell leader, who is often the only contact between the cell and the outside world.

We use NetWatch to build a series of experiments studying evolution of terrorist networks, examining strategies for mapping and destabilizing terrorist networks.

The simulation (see figure 2) consists of two networks of agents: the **Red Team**, representing the covert network of a terrorist organization and the **Blue Team** representing the anti-terrorist forces.

4.1 Red Team

Based on publicly available data, the following profile of the structure of covert networks has been derived [5]:

- The network consists of small cells (mean cell size of 6 members) with very low interconnections between cells.
- Internally, the cells exhibit all-to-all communication patterns.
- There is a very low probability of two individuals communicating by chance (0.007).
- The probability of triad closure (link from x to y being more likely if both x and y are linked to third party z) is 0.181.
- Senior members of each of the cells are often also parts of other cells and interact with other senior members on the network.
- Cell leaders are more knowledgeable than other members.
- Cell members share an ideological doctrine but also specialized knowledge (i.e. bombmakers, drivers, operatives).
- Cells use information technologies and electronic communication.

Above parameters form a statistical profile from which we can generate simulated organizational networks. The plot on figure 3 shows a covert network generated using parameters specified above, based on the structure of terrorist networks as shown by Valdis Krebs [10]. The agents in the Red Team network

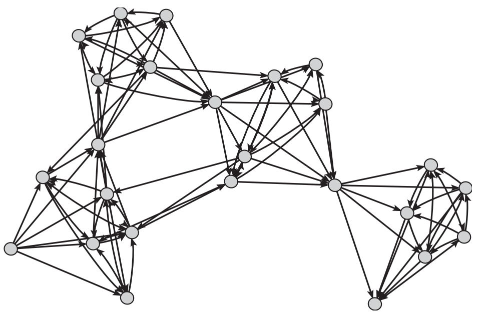

Figure 3: Red Team: A Cellular Covert Network

execute a set of tasks modelled on tactical descriptions (i.e. sequences of tasks, and their skill and resource requirements) of terrorist bombing of the U.S. embassy in Tanzania.

4.2 Blue Team

The Blue Team is an Anti-Terrorism organization consisting of a small number of fully interconnected law enforcement agents.

The goals of the Blue Team are:

- Learn the structure, task assignments, and knowledge distribution of the Red Team.
- Remove or isolate Red Team members, aiming to maximally impair Red Team's performance.

The Blue Team has no access to the actual information about the Red Team. Its only source of information is a set of wiretaps on the communication network of the Red Team. Based on the information it is able to collect, the Blue Team takes action to destabilize the Red Team by isolating agents of the Red Team. Further, like the Red Team, the initial meta-matrix for the Blue Team can be read in from actual data or set up experimentally.

4.2.1 Wiretaps

A wiretap in NetWatch is an agent that selectively intercepts messages from the message stream and routes them to one or more Blue Team members. Blue Team agent receives and records the origin and destination of the wiretapped message.

If a wiretap could capture all communication, the Blue Team would be able to create a full and accurate picture of the Red Team network. However, a wiretap is not able to capture all relevant messages. Based on empirical information on the use of wiretaps, the wiretap efficiency is set between 5% and 25%. In this paper, we study three different wiretap strategies:

- **Random:** any message in the message stream has an equal probability of being intercepted. Signal-to-noise ratio is an independent variable, and ranges from 5% useful signal to 25% useful signal - thus allowing us to study the effect of signal-to-noise ratio on the quality of collected data.
- **Snowball:** Captures traffic originating from one agent, and sequentially targets every agent it communicated with. This essentially is a breadth-first search of the network.
- **Socially Intelligent Traffic Analysis:** As the Blue Team agents receive messages from the wiretap agents, they use their address information to build a representation of the network of the Red Team, or the *Learned Network*. Then, the agents analyze the perceived network and move the wiretap to an agent that is the highest in one or more network measures, such as *degree centrality* [6] and *cognitive load*.

  Cognitive load is a notion similar to the task load measure developed at NASA [7]. It measures the extent to which the person has to engage in mental activity to do the assigned tasks, defined as:

  1. number of people person *i* interacts with / total number of people in the group;
  2. number of tasks person *i* is assigned to / total number of tasks;
  3. sum of number of people who do the same tasks person *i* does / (total number of tasks * total number of people)

  The agents periodically reevaluate the target of wiretapping and switch if a better target is found or if the amount of new information about the current target becomes too low.

4.2.2 Network Destabilization Tactics

In some of the experiments, the Blue Team collects information about the Red Team and attempts to influence Red Team performance by attacking its vulnerability points (e.g., by isolating or terminating key agents). We test a number of strategies for finding key agents, including:

- **Random:** a base-line strategy; isolate one random individual from the network.
- **Highest degree centrality:** isolate one agent from the covert network that, in the data collected by the Blue Team, has the highest degree centrality.
- **Highest cognitive load:** isolate one agent from the covert network that, in the data of the Blue Team, has the highest cognitive load.

4.3 Results

Following are some results from NetWatch. We include these results to illustrate the strength of this approach for addressing real-world problems. Note that these results are preliminary and will become more complete as additional aspects of the situation are included such as technological capabilities and alternative task structures.

For these results, we simulated 150 Red Team networks ranging in size from 100 to 400 agents. In each case we simulated the system for 100 time periods, which, in simulation time, is equivalent to approximately 25 days. The Blue Team began with no knowledge of the Red Team. The Blue Team is further acting like a single homogeneous unit where all actors or subagencies completely cooperate.

Figure 4 presents results indicating the average accuracy of the Blue Team's perception of the Red Team as it changes over time using different wiretapping strategies. The performance of a strategy is measured as the probability of correctly identifying the top actors within the group based on various network measures averaged across all networks. We see that a socially intelligent wiretap strategy enables the collection of a more accurate picture of the opponent. However, optimality of one socially intelligent strategy over another changes over time.

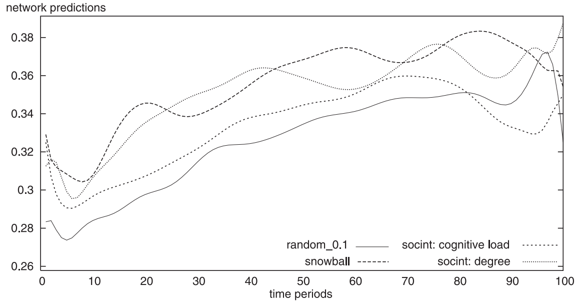

Figure 4: Average Performance of Wiretapping Strategies

		wiretapping strategies			
		Random	Snowball	Degree	Cog. Load
attacks	Random	-5.4%	-13.4%	-3.9%	-18.3%
	Degree	-21.2%	-24.0%	-21.5%	-21.1%
	Cog. Load	-5.7%	-11.0%	-13.5%	-3.0%

Table 2: Reduction in Organizational Performance of the Red Team due to Anti-Terrorist Activity.

In further experiments we simulated a set of 150 networks allowing members of the Blue Team to isolate one of the agents within the Red Team network. Each network was run 4 times using different destabilization strategies. The four strategies examined are:

- no attacks on the Red Team.
- isolate a member of the Red Team at random.
- isolate the Red Team agent with the highest degree centrality.
- isolate the Red Team agent with the highest cognitive load.

Table 2 presents results indicating the change in performance for each of the analyzed strategies. The performance of the Red Team is measured as a ratio of number of successfully completed tasks to the number of assigned tasks. Each cell in this table shows the percentage difference in performance from the 50 time periods prior to when the first agent is isolated and 50 time periods after the isolation. This shows the immediate impact of the various destabilization strategies. Note, in general, any of these strategies does lead to a performance reduction thus indicating that there has been some destabilization. Second, there is an interaction between the type of wiretapping strategy and the type of destabilization strategy.

5 Conclusions

These, admittedly preliminary results, show the potential power of multi-agent network simulations for addressing real-world issues. Agent-based simulation frameworks enable a more realistic and extendable architecture for addressing policy issues in a manner comparable to human behavior. Such work moves us from the realm of building agents to act more or less independently on behalf of people to the realm of using collections of agents to reason about how people as groups behave.

The advantage of an agent approach is that it enables the simulated actors to behave like humans - in that they are cognitively and socially bounded with knowledge of themselves and others dependent on their personal history. When such agents are embedded in dynamically evolving networks, the entire simulated system takes on the social and technological constraints consistent with empirical findings. The advantage of using AI planning, reasoning and decision making techniques is that complex intelligent agents are extensible to multiple tasks and scenarios.

Such models enable the researcher to examine the nature, not just of cognition but of social cognition, and to explore policy and managerial issues. In doing so, the goal is not to "predict" specific events, but to decrease uncertainty in detecting trends. As such, these tools are valuable assets to decision makers.

References

[1] D. Alberts, J. Garstka, and F. Stein. *Network Centric Warfare: Developing and Leveraging Information Superiority*. CCRP Publication Series, 1999.

[2] N. Berry. The international islamic terrorist network. *CDI Terrorism Project*, September 2001.

[3] Rodney Brooks. A layered intelligent control system for a mobile robot. In *Proceedings of the Third International Symposium of Robotics Research*, page 8. MIT Press, october 1985.

[4] Kathleen M. Carley. On the evolution of social and organizational networks. *Research in the Sociology of Organizations*, 16(special issue on Networks In and Around Organizations):3–30, 1999.

[5] Kathleen M. Carley, Ju-Sung Lee, and David Krackhardt. Destabilizing networks. *Connections*, 24(3):31–34, 2001.

[6] L. C. Freeman. Centrality in social networks: Conceptual clarication. *Social Networks*, 1:215–239, 1979.

[7] S.G. Hart and L. Staveland. Development of nasa-tlx (task load index): Results of empirical and theoretical research. In *Human mental workload*, pages 139–183. P.A. Hancock and N. Meshkati (Eds.), Amsterdam: Elsevier, 1988.

[8] David Krackhardt. Assessing the political landscape: Structure, cognition, and power in organizations. *Administrative Science Quarterly*, (35):342–369, 1990.

[9] David Krackhardt and Kathleen M. Carley. A pcans model of structure in organizations. *Proceedings of the 1998 International Symposium on Command and Control Research and Technology*, pages 113–119, june 1998.

[10] Valdis E. Krebs. Mapping networks of terrorist cells. *Connections*, 24(3):43–52, 2001.

[11] Zhiang Lin and Kathleen M. Carley. *Designing Stress Resistant Organizations: Computational Theorizing and Crisis Applications*. Kluwer, Boston, MA, 2003.

[12] H. Simon. A behavioral model of rational choice. *Quarterly Journal of Economics*, 69:99–118, 1955.

[13] Daniel M. Wegner. Transactive memory: A contemporary analysis of the group mind. *Theories of group behavior, edited by B. Mullen and G. R. Goethals*, pages 185–208, 1987.

Contact
Maksim Tsvetovat, Kathleen M. Carley
Institute for Software Research, International
Carnegie Mellon University
5000 Forbes Ave
Pittsburgh, Pennsylvania 15213, USA
maksim@cs.cmu.edu, kcarley@ece.cmu.edu

Kathleen Carley is a professor at the Institute for Software Research International in the School of Computer Science at Carnegie Mellon University and director of the center for Computational Analysis of Social and Organizational Systems (CASOS). She carries out research that combines cognitive science, dynamic social networks, text processing, organizations, social and computer science in a variety of theoretical and applied venues. Vita, papers, and abstracts can be found at www.casos.ece.cmu.edu/bios/carley/bio_carley.html

Maksim Tsvetovat is a Ph.D. student in the Computation, Organizations and Society programme at the Institute for Software Research International in the School of Computer Science at Carnegie Mellon University. His research is centered on building high-fidelity simulations of social and organizational systems using concepts from distributed artificial intelligence and multi-agent systems. Maksim's vita and publications can be found on www.cs.cmu.edu/~maksim/research