Künstl Intell (2012) 26:127–140
DOI 10.1007/s13218-012-0179-2

FACHBEITRAG

Sol: An Agent-Based Framework for Cyber Situation Awareness

Jeffrey M. Bradshaw · Marco Carvalho · Larry Bunch · Tom Eskridge · Paul J. Feltovich · Matt Johnson · Dan Kidwell

Received: 12 August 2011 / Accepted: 4 March 2012 / Published online: 8 May 2012

Abstract In this article, we describe how we augment human perception and cognition through Sol, an agent-based framework for distributed sensemaking. We describe how our *visualization approach*, based on IHMC's OZ flight display, has been leveraged and extended in our development of the *Flow Capacitor*, an analyst display for maintaining cyber situation awareness, and in the *Parallel Coordinates 3D Observatory* (PC3O or Observatory), a generalization of the Flow Capacitor that provides capabilities for developing and exploring lines of inquiry. We then introduce the primary implementation frameworks that provide the core capabilities of Sol: the *Luna Software Agent Framework*, the *VIA Cross-Layer Communications Substrate*, and the *KAoS Policy Services Framework*. We show how policy-governed agents can perform much of the tedious high-tempo tasks of analysts and facilitate collaboration. Much of the power of Sol lies in the concept of *coactive emergence*, whereby a comprehension of complex situations is achieved through the collaboration of analysts and agents working together in tandem. Not only can the approach embodied in Sol lead to a qualitative improvement in cyber situation awareness, but its approach is equally relevant to applications of distributed sensemaking for other kinds of complex high-tempo tasks.

J.M. Bradshaw (✉) · M. Carvalho · L. Bunch · T. Eskridge · P.J. Feltovich · M. Johnson
Florida Institute for Human and Machine Cognition (IHMC),
40 South Alcaniz Street, Pensacola, FL 32502, USA
e-mail: jbradshaw@ihmc.us

D. Kidwell
e-mail: DLKidw2@tycho.ncsc.mil

Keywords Cyber security · Teamwork · Software agents · Policy management · Resilience · Coactive design · Emergence · Joint activity · Sensemaking

1 Introduction

Despite the significant attention being given the critical problems of cyber security, the ability to keep up with the increasing volume and sophistication of network attacks is seriously lagging. Throwing more computing horsepower at fundamentally-limited visualization and analytic approaches will not get us anywhere. Instead, we need to seriously rethink the way cyber security tools and approaches have been conceived, developed, and deployed.

We are taking advantage of the combined strengths of humans and software agents to create new capabilities for Network Operations Centers (NOCs). Our goal is to enable continuous situation awareness, rapid detection of threats, and effective protection of critical resources. The new "coactive emergence" approach to embodied in Sol is equally relevant to applications of distributed sensemaking for other kinds of complex high-tempo tasks such as real-time disease control or disaster management.

In Sect. 2, we describe how our *visualization approach*, informed by lessons-learned from IHMC's OZ flight display, has been leveraged and extended in our development of the *Flow Capacitor*, an analyst display for maintaining cyber situation awareness, and in the *Parallel Coordinates 3D Observatory* (PC3O or Observatory), a generalization of the Flow Capacitor that provides capabilities for developing and exploring lines of inquiry. Section 3 describes the challenge of threat understanding in Sol as a process of *coactive emergence*, whereby a comprehension of complex

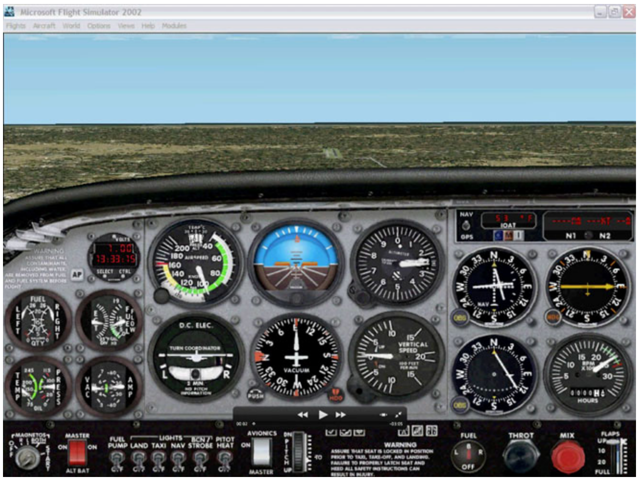

Microsoft Flight Simulator 2002
Flights Aircraft World Options Views Help Modules
D.C. ELEC.
VERTICAL SPEED
FUEL
MASTER
AVIONICS
WARNING
THROT
MIX
FLAPS

Fig. 1 A traditional cockpit display

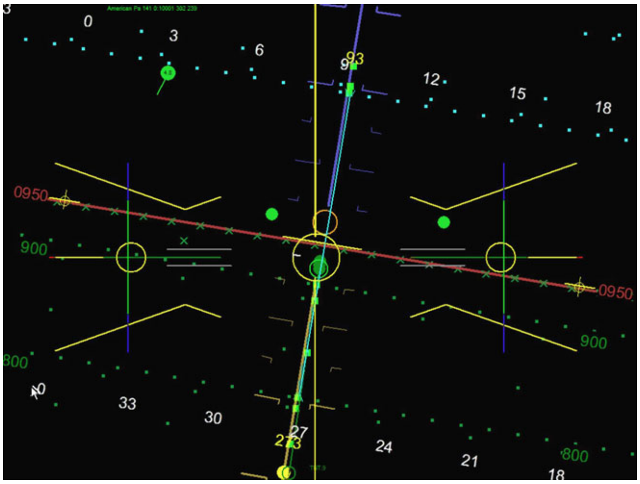


Fig. 2 IHMC's OZ flight display

situations is achieved through rapid convergence to a common model of the situation by analysts and agents working together in tandem. Section 4 describes additional forms of agent assistance that we have explored and implemented within Sol. Section 5 introduces the primary implementation frameworks that provide the core capabilities of Sol: the *Luna Software Agent Framework*, the *VIA Cross-Layer Communications Substrate*, and the *KAoS Policy Services Framework*. In Sect. 6, we conclude that our work on the Sol framework may suggest significant new directions in automated assistance for sensemaking in a variety of application domains.

2 Maintaining Cyber Situation Awareness and Exploring Lines of Inquiry

Our approach to cyber situation awareness displays is informed by lessons learned in the design of IHMC's highly-successful OZ flight display[1] [43, 44]. Although space limitations preclude a complete description of OZ, we will attempt to give enough detail about its major features and benefits to convey its relationship to our current work: the design of displays for cyber situation awareness.

2.1 Lessons learned from the OZ flight display

Instead of relying on a continual visual scan of cockpit instruments, as on the traditional flight display shown in Fig. 1, OZ presents information holistically and in the context of the current state of the world outside (Fig. 2). Presenting flight performance information *in context* allows people to more easily maintain overall situation awareness. Presenting information *holistically* allows dependencies among key flight parameters to be made salient through the direct perception of visual primitives. Modifications made to any part of the model through pilot input or changes in the operating environment immediately affect all related elements.

Though the display's reliance on colored lines and dots on a black background may seem a primitive throwback to first-generation video games, this simplicity is by design, based on a sophisticated understanding of the latest research results in human perception and cognition. Instead of relying on the slow and small human focal vision system, OZ is designed to use the fast and robust *ambient vision system*—the same system that people use to quickly and successfully navigate crowded hallways without conscious thought or to catch a football on the run [31, 32, 45, 46]. As another example, OZ exploits the capabilities of human vision for quickly perceiving changes by using movement to convey difficult, correlated information [33, 38].

A final aspect of OZ to note is the use of an explicit performance model to inform the pilot of normative system information relating to the aerodynamics of flight for the type of aircraft being flown—see, e.g., the four symmetric checkmark-like lines shown in Fig. 2. By displaying the performance model on the screen in terms of lines and limits, OZ gives the operator a reference point against which to compare the evolving situation. Keeping the airplane on course becomes a simple matter of interactive graphical alignment rather than complex mental reasoning [1, 39, 44].

Due to all these features, experimentation has repeatedly demonstrated the superiority of OZ over traditional displays in minimizing pilot error, reducing pilot disorientation, and maintaining situation awareness. Because of the OZ display's reliance on the ambient visual system, its advantages are shown even more dramatically in experimental conditions where the pilot is temporarily blinded by a flash of light

[1] OZ relates to the classic film "The Wizard of OZ" and is not an acronym.

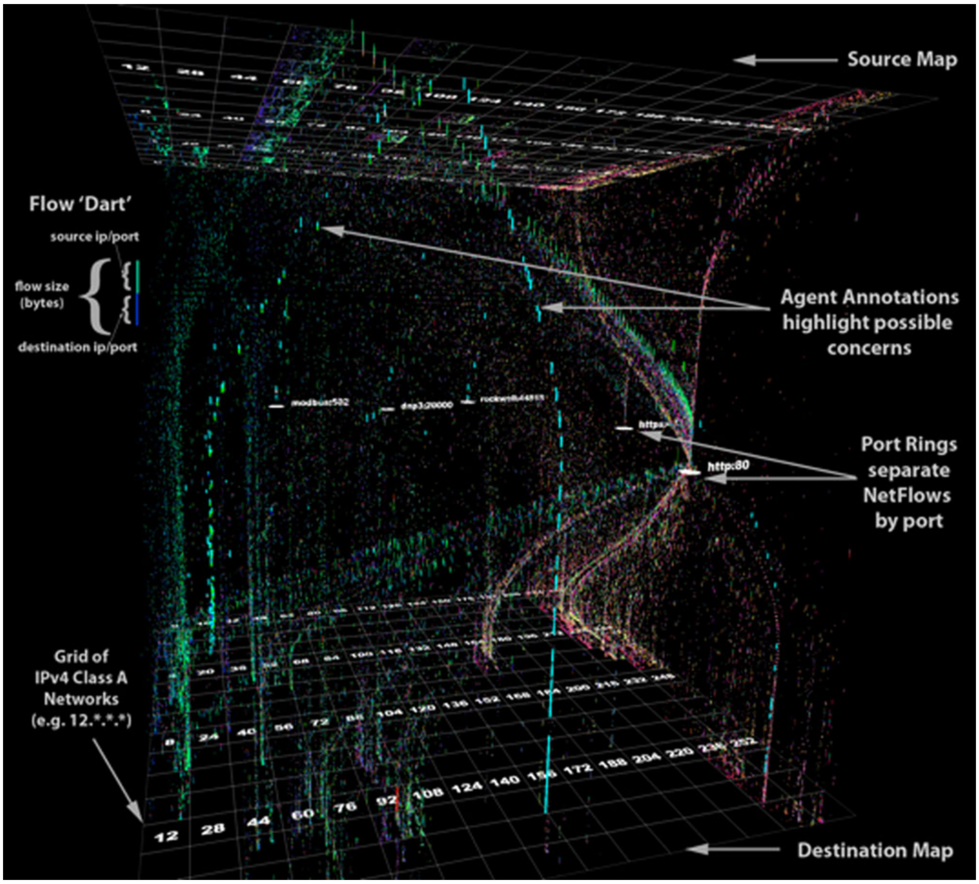


Fig. 3 Annotated Flow Capacitor Example

(as when, e.g., impaired by lack of oxygen) or distracted by performing auxiliary visual tasks that rely on the focal vision system (e.g., reading) [45]. Beyond its role in simplifying flight-related tasks, the integrated performance model has an added training benefit—helicopter pilots trained using the hover functionality of OZ are able to more quickly acquire the depth of understanding necessary to master difficult challenges unique to rotorcraft flight, and fixed-wing pilots learn faster, retain training longer, and have a deeper understanding of the fundamental rules of flight than their conventionally-trained counterparts [40–42].

2.2 Applying OZ Principles in the Flow Capacitor

We have used lessons learned from the OZ flight display in our design of visualizations for cyber situation awareness. Consider, for example, a visualization we call the Flow Capacitor (Fig. 3).

The Flow Capacitor. The Flow Capacitor is a highly-configurable interactive 3D visualization of Internet traffic. The input to this visualization is NetFlow records (Cisco Systems [15]). Each NetFlow record contains information about source and destination addresses of the flow, protocols and ports used, size and rate of the flow, and other information.

The two planes at the top and bottom of the display are mirror images of each other. The top plane shows a "Source IP Map" of the NetFlow records and the bottom plane shows a "Destination IP Map." Each of the two planes shown in Fig. 3 represents the full IPv4 address space where each point on a plane is a unique IP address—defining, in this case, a model of 65,536 pixels cubed. The 256 grid boxes on each plane divide the IPv4 space by the first octet in the address, the class A network. Due to the modularity of the agent architecture, upgrading to IPv6 will be straightforward.

The record of a given flow at a specific moment of time is represented as points on the source and destination planes, creating a result similar to heat maps. The color of the source and destination points encodes the first three octets of the IP address (i.e., the class C network address). Users can drill down at any time to see a more detailed projection of the traffic on a plane, displaying, for example, current flow records from or to all addresses *within* a given Class A network.

As alternatives to the IPv4 maps shown, any number of alternate plane types can be defined. For instance, the framework can geo-locate the IP addresses and project the source and destination locations as latitude and longitude on a map of the world. Conceptually-defined planes, categorizing flows from certain types of groups (e.g., criminals, nation-state attacks) or economic sectors (e.g., financial, energy) can also be defined.

NetFlow "darts." In addition to being shown on the source and destination map planes, each NetFlow is also represented as a short line segment or "dart" that moves in real time from a source in the top plane to a destination in the bottom one. The length of the dart is proportional to the number of bytes that are being transferred between the source and destination by that flow. The appearance of the top half of the dart reflects attributes of the source plane, while the bottom half reflects attributes of the destination plane. For example, the two halves of the dart may be shown in one of 65,536 unique colors corresponding to the source IP and the destination IP. Alternatively, for example, the colors could be defined to correspond to the port number. The properties on which the colors are based and the particular colors chosen for a given property value can be easily redefined to represent other flow attributes such as protocol, duration, and TCP flags.

Port rings. The white rings labeled with protocols and port numbers (e.g., http:80, https:443) "attract" NetFlows that have a matching source or destination port value. This allows them to be visually grouped by the ring as they travel downward. The rings are initially placed in sorted order, but can be manipulated with a pointing device. For example, an analyst can interactively move the ring to a less congested area of the display in order to more easily separate and monitor certain kinds of traffic. Besides ports, other kinds of properties can also be used to define rings.

User controls. Configuration of user controls is performed graphically on-the-fly in auxiliary window panes. A pointing device can be used to rotate, zoom, and pan the view interactively. Modifier keys are used in conjunction with mouse actions (e.g., click, drag) in order to differentiate user intent. A vertical timeline with configurable color-coded key event annotations provides a temporal overview of the unfolding situation, and incorporates a slider control for quick navigation through time (see Fig. 4). The user can pause, rewind, and fast-forward the display for instant replay in slow- or fast-motion—enabling users to engage in different kinds of attentive and preattentive visual processing of the information.

Pausing the display enables the user to mouse-over individual flow darts to display flow metadata. To allow easy selection, darts can be made "bolder" automatically when the display is paused. In addition to specific dart selection, individual flows or groups of flows can be selected for more detailed analysis by software agents or for viewing in other kinds of displays. Selections of interest can also be shared between different people and groups.

The period of time represented between the top and bottom planes can be configured to any length, from weeks or days to milliseconds. Slider controls below the timeline allow the user to specify the time frame of interest and the rate at which time passes. The slider control on the vertical timeline is automatically sized to indicate the proportion of time in the currently-displayed slice relative to the length of the overall timeline. The user also determines whether to render all of the NetFlow records or to filter them based on a combination of protocol, port, IP, and country.

Toward a visual model for network sensemaking. Although nearly all of the principles behind the OZ flight display design have been straightforwardly applied to cyber situation awareness, there are some aspects that have proven more challenging.

One of the most important differences between these two applications is the difficulty in finding the equivalent of the flight performance model for network analysis. Whereas the primary task of the pilot is to fly effectively within the known parameters of a fixed aerodynamic model, the job of the NOC analyst is to understand emerging threats accurately against the moving target of a network that is constantly changing. With this fact in mind, it is easy to see that what the analysts need is not a control device, nor merely an informative picture of the world, but rather a tool for *formulation of hypotheses* about a situation [26, p. 286]. In short, the utility of a given visual model for sensemaking must be, in our view, evaluated pragmatically in terms of its effectiveness in asking and answering a serviceable range of relevant questions.

2.3 Example: Understanding a Distributed Denial-of-Service Attack Using the Flow Capacitor

From the snapshot of the Flow Capacitor in Fig. 4, we can see the sequence of events leading up to a distributed denial-of-service attack portrayed in graphic clarity. Reading from bottom (oldest events) to top (most recent events):

1. Blacklisted scanners [yellow] get control signals from some unknown command-and-control node not yet on our blacklist (yellow flows over Italy)
2. Blacklisted scanners [yellow] hit whitelisted power infrastructure nodes [white] on US west coast (four streaks of yellow)
3. Some power infrastructure nodes respond to the scanners (yellow and white flows cross the tail of the scan attacks with the yellow tags at the opposite end of the darts). There are two sets of four darts moving diagonally from left to right. The set on the left (over the Atlantic) consists of responses from California and Washington to the scanners in Italy. The set on the right consists of the scanners in Italy subsequently passing these responses on to a C2 node in China.
4. Blacklisted bots [blue] receive control signals from their C2 (burst of blue from one to many on the right)

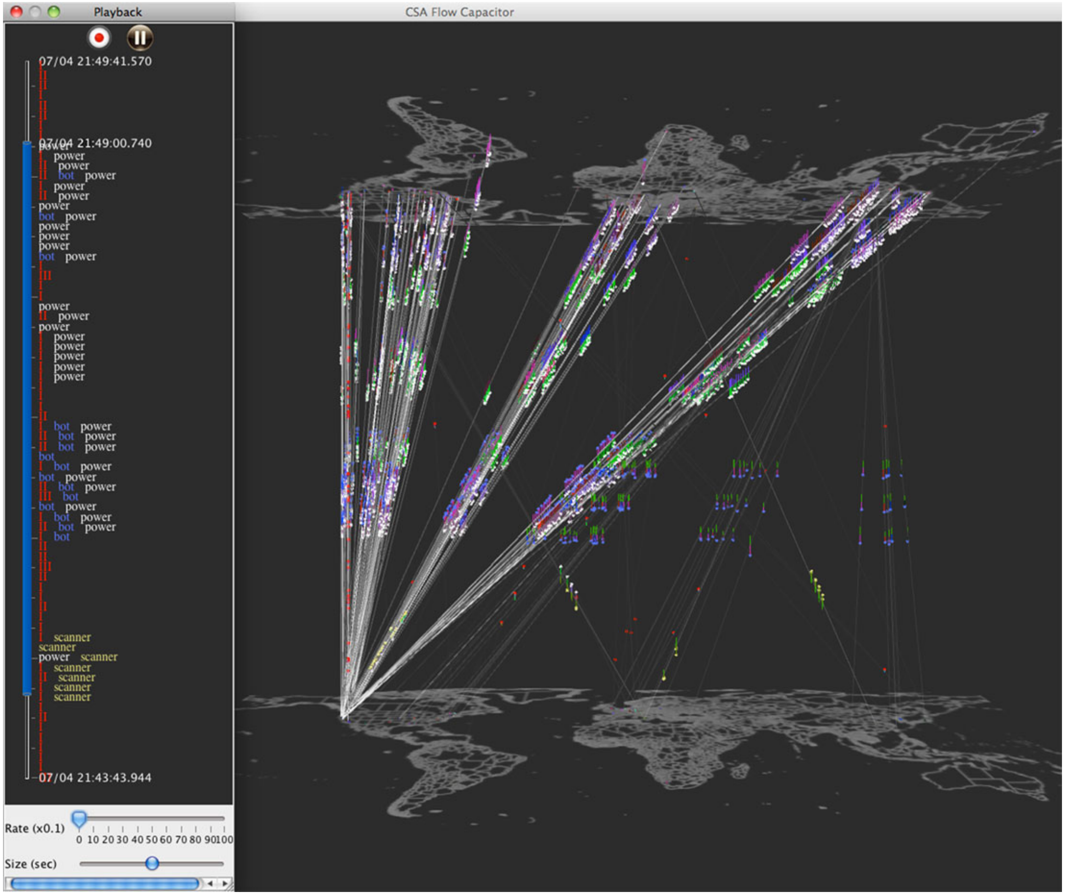


Fig. 4 Distributed Denial-of-Service Attack Example

5. Blacklisted bots attack whitelisted power infrastructure (blue and white "tornados")
6. Unknown nodes, not yet on our blacklist, attack whitelisted power infrastructure nodes (white "tornados").

2.4 Visual Exploration of Lines of Inquiry: The Parallel Coordinates 3D Observatory

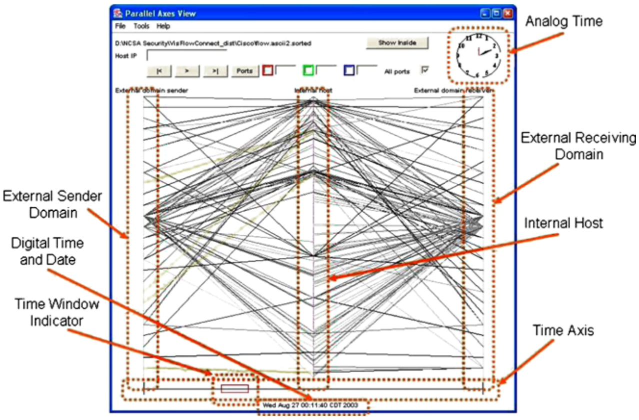
The Flow Capacitor is designed to answer a core set of important questions about the number and nature of flows between sources and destinations in networks of any size. In order to broaden the range of questions that can be asked, we devised a generalization of the concept of a Flow Capacitor called the Parallel Coordinates 3D Observatory—"PC3O" or "Observatory" for short.

Toward a performance model for network analysis. Although nearly all of the principles behind the OZ flight display design have been applied in new ways in our work on cyber situation awareness, there are some aspects that have proven more challenging. These additional challenges help inform the design of the PC3O.

One of the most important differences between these two applications is the difficulty in finding the equivalent of the flight performance model for network analysis. Whereas the primary task of the pilot is to fly effectively within the known parameters of a fixed aerodynamic model, the job of the NOC analyst is to understand emerging threats accurately against the moving target of a network that is constantly changing. With this fact in mind, it is easy to see that what the analysts need is not a control device, nor merely an informative picture of the world, but rather a tool for *formulation of hypotheses* about a situation [26, p. 286]. In short, the utility of a given visual model for sensemaking must be, in our view, evaluated pragmatically in terms of its effectiveness in asking and answering a serviceable range of relevant questions.

Comparison of PC3O to Parallel Coordinate Graph approaches. Parallel coordinate graphs are a common way of visualizing data with a large number of constituent features.[2] These graphs show connections between feature values based on a given set of data, usually with each feature dimension represented by a vertical line, which normalizes that features values in to a continuous range over the length

[2] See [16] for a survey of visualization approaches for network situation awareness.



D:\NCSA Security\VisFlowConnect_dist\Ciscoflow.ascii2.sorted

Host IP

Wed Aug 27 00:11:40 CDT 2003

Fig. 5 VisFlowConnect Parallel Coordinates View

of the line, or in equally spaced points for discrete feature values. For example, Fig. 5 shows a parallel coordinates type display called VisFlowConnect [53]. External senders are shown on the left, internal hosts in the center, and external receivers on the right. This visualization facilitates recognition of intrusions such as port scans or distributed denial-of-service attacks.

While such interfaces are easy to read in low-volume, small network situations, they place a large burden on the operator to notice the patterns indicative of intrusions. Even with large or multiple screens, clutter from overlapping connection lines in larger networks can increase to the point where important information needed by the analyst to recognize the patterns indicative of intrusions may be obscured. Our PC3O approach, coupled with the agent annotations described in the next section, helps address these and other of the drawbacks of conventional parallel coordinate graphs.

Enhanced visual separation of anomalies using custom configurations of multiple planes. The Flow Capacitor can be seen as a base configuration of the Observatory, with two identical planes being shown. PC3O extends this idea by allowing any number of additional planes to be vertically layered so they sort the downward path of the flow darts. Because the data are shown in planar form, combinations of features can be displayed in two dimensions (e.g., packet size *vs.* packets per second). In this way, each plane itself contributes to the understanding of the network situation, as well as contributing as a component of the overall PC3O configuration.

At each vertical layer, all the flows may pass through a single plane that visually highlights their individual features. Alternatively, the flows can be routed by Boolean operators into one of multiple planes (e.g., a plane that captures flows within our network vs. a second plane that captures flows outside our network), allowing analysts to distinguish via visual separation the interesting characteristics of the data versus the mundane. By building visual separation into the graphical model, the analyst gains *comparative information* (e.g., proportion of threats going to hosts in the energy sector vs. the financial sector) and *correlative information*, by seeing untagged flows that are behaving similarly to tagged flows. By allowing analysts to construct a custom environment of heterogeneous planes that separate and characterize the flows, the Observatory allows the incremental formulation of a whole series of hypotheses constituting a line of inquiry, at the price of some added complexity for the novice user. Useful configurations (lines of inquiry) of PC3O planes can be archived for future reuse in analogous situations. One could envision whole libraries of such inquiry tools.

Example: Exploring a line of inquiry. As an example of how the Observatory supports a line of inquiry, consider a network analyst who is investigating a series of attacks on port 20000 to the critical infrastructure of a set of electrical power plants. Wondering whether any attackers were missed in the original report, the analyst widens the search for attackers to include flows using SCADA-related protocols originating from a larger geographical area and using not only port 20000 but also neighboring ports of significance to SCADA systems. The analyst uses the Observatory to define a first plane that plots the use of SCADA protocols on all related ports for the larger geographical region.

Having discovered some previously-unrecognized attackers in this way, the analyst creates a second vertical layer in order to answer the question of whether a particular regional utility company is the sole target of the of the attack, or whether a second utility in the same region is also being threatened. The new layer consists of two planes, one of which captures flows going to portions of the IP space corresponding to one regional utility company and the second of which captures flows going to portions of the IP space used by a second company.

Having found out that attacks are targeting all power utilities in the region, and not just one particular supplier, the analyst now wants to know who needs to be advised of the situation. The analyst constructs a third layer, consisting of two geographical planes that respectively capture the physical locations of the plants under attack. PC3O enables the analyst to discover that, in the case of the first utility, only the supervisor for a small region needs notification, while in the case of the second utility, multiple regional supervisors need to be advised.

3 Threat Understanding as a Process of Coactive Emergence in Human-Agent Teams

Building on a more general theory of joint activity in humans and machines [28, 29], Johnson coined the term "coactive design" as a way of characterizing an approach to human-agent interaction that takes *interdependence* as the central organizing principle among people and agents working together [17, 18, 24]. Sol applies these ideas to sensemaking through in what might be called a process of *coactive emergence*, whereby a comprehension of complex situations is achieved through accelerated convergence on relevant models of the situation by analysts and agents working together in tandem [5]. In this way, Sol offers support for an ongoing scaffolding process among agents and human analysts. For example, analysts use what they know to discover new patterns of attacks, to define agents to detect and monitor them, and to define policy constraints to govern agent behavior. Subsequently, agents hypothesize correlations between sets of flows, and analysts, in turn, use these results to construct new agents, agent policies, and lines of inquiry. Below we explore these ideas in more detail.

Coactive emergence. Emergence describes the phenomenon whereby complex systems may arise from interactions of much simpler primitives. A classic example is the development of ant colonies (see, e.g., [23, pp. 117–118]).

To a greater or lesser degree, emergent systems are bounded by inherent structural and environmental constraints. For example, in their account of human language development, Bratman, *et al.* [6] assert that:

> … specific properties of language might have emerged as an adaptive response to joint pressures from the environment and constraints on an agent's cognitive architecture. The approach suggests that linguistic systems can be described as *boundedly optimal policies in multi-agent dynamic control problems* defined by specific environments, agent computational structures, and task-oriented… rewards.

Human culture also can be described in terms of rule-governed emergence. The normative pressure of culture serves to reduce the number of alternatives available for acceptable behavior in a given setting, thus greatly simplifying the problem of human choice in routine situations [17]. It should be noted, however, that the governing constraints of culture are themselves subject to modification: in other words, while it is true that culture helps determine individual behavior, individuals, in turn, have the power to mold culture. Viewed rightly, therefore, the development of culture, like the joint development of threat understanding by humans and agents in cyber sensemaking, might be described as a process of *coactive emergence*.

Coactive emergence goes beyond relatively static forms of bounded emergence to allow either inherent or policy-based constraints that govern the process of emergence to be themselves subject to change. In coactive emergence, both top-down policy constraints and bottom-up individual behavior are simultaneously shaped in mutual fashion, enabling continuous adaptive refinement. Top-down policy constraints on actors aim to achieve overall system objectives, to rapidly propagate lessons learned about productive and unproductive actions, and to avoid undesirable states and events. On the other hand, bottom-up emergence of novel actor strategies serves to achieve individual tasks. Because both the rules and the individual behavior for each party are subject to change, *coactive emergence* enables a wider range of adaptations.

Coactive emergence is a form of what Langton calls "semantic" or "second-order" emergence whereby "the system is able to detect, amplify, and build upon emergent behavior. The latter can only happen by operating on the behavior programs that causally influence behavior, similar to the way genetic evolution operates on the genes" [30, p. 90].

Ideally, as in cooperative evolution, the process of coactive emergence is symbiotic, leading to progressive convergence on threat hypotheses. Of course, some amount of competitive evolution may also be desirable in sensemaking in order to encourage the exploration of the same space (or a wider space) from different perspectives.

Our emphasis on joint activity of humans and machines proceeds from the premise that the use of mixed human-agent systems can increase the range, richness, and utility of models that could be explored by humans or agents alone. In mixed human-agent teams, people occupy a privileged position because, among other things, they generally know more about the way joint tasks interact with broader ongoing activities and with the situation at large. For these reasons, humans have an important role in keeping agent taskwork aligned with its wider contexts [22]. In their complementary role, agents can help people cope, for example, with the volume, tempo, computational complexity, and highly-distributed nature of joint tasks. In addition to supporting appropriate aspects of taskwork, agents can be used to help support coordination and other aspects of team process, as described below.

Agent support for human-human teamwork. The scaffolding process for human-agent collaboration just described suggests, of course, a similar model for human-human teamwork. One way which Sol agents support this process is through agent-enabled shared windowing and selection in analyst displays. Our advances enable efficient joint control and remote viewing of all or part of a visual perspective while minimizing network loads. Selections of objects within views can also be shared across platforms and exploited across different types of views or in directing agent processing of information. In the future, new kinds of visualizations can straightforwardly reuse these foundational capabilities.

In addressing the problem of situation awareness, we should not only consider the problem of how to maintain continuous awareness of the relevant dimensions of the external environment, but also how to better track team processes—establishing and maintaining the degree of common ground among human participants that is necessary for analysts to coordinate and build upon one another's work. In support of this objective, we implemented an initial prototype of what the team affectionately calls the "Fishtank." The idea of the FishTank is to enable continuous progress appraisal [19] by groups of analysts through a visualization that enables them to easily see what tasks and which human and/or agent team members are significantly ahead or behind schedule, and thus replan their own efforts on interdependent tasks accordingly. The name "FishTank" for the concept comes from the visual idea of tasks needing attention and team members needing help rising gradually upward on the display according to their urgency, like dead fish floating to the top of a fishbowl.

Human-agent teamwork through agent annotations. In monitoring complex, high-tempo events, it is impossible for a human to identify every significant flow anomaly in Internet-scale displays. To help with this problem, we use software agents to automatically enrich the raw NetFlow records with information about attacks and other potentially malicious behavior. As new threats emerge, agents can automatically learn new patterns. Moreover, as new analytic innovations are developed, new kinds of sensing agents also can be straightforwardly added by analysts.

Agents visually annotate the display in real-time in order to highlight and draw the attention of the analyst to anomalous or otherwise interesting elements, such as possible attacks. For example, in the Flow Capacitor agent-annotated flows of interest are highlighted by attaching "flags" to the end of each dart. The flag colors and what they indicate (e.g., type of attack, presence of flow source in blacklist) can be customized by the analyst. In this way, the Flow Capacitor functions as a *mediating representation*—a highly-communicative visual model of the situation that can be simultaneously used by mixed teams of people and software agents in order to come to a common understanding of a situation [20, 25].

Benefits of organizing agents hierarchically. Agents may be organized hierarchically to facilitate the enrichment of NetFlow records at multiple levels of abstraction. In this way, agent annotations do not merely highlight low-level indicators of intrusion patterns, but can directly identify the type of intrusion itself. For instance, instead of requiring the analyst to notice that a configuration of connecting lines (some of which may be obscured) indicates a distributed port scan, agents working on abstracted data semantics can directly indicate the source of the attack. As another example, if a message is anomalous because it is sending oversized packets to a port associated with an SQL database, higher-level agents can abstract that message and represent it as an instance of an SQL injection attack. This ability to reduce perception and reasoning requirements on the analyst is a major benefit of agent-based analytics.

Having characterized the data in terms of identifiable intrusions enables analysts to carry out standard procedures in response. These procedures could include the automatic configuration of visual displays that allow the analyst to isolate intruder actions, or the spawning of new agents to collect data related to the identity of the network threats. Agents could also perform interdictory actions to prevent the intrusion from propagating further or wasting more network resources.

4 Additional Forms of Agent Assistance

Examples of ways in which software agents could be used within Sol to assist analysts include the following:

- *Freeing up time.* Agents promote continuity in investigation by continuing to work when analysts are unavailable. They can free up analyst time by performing tedious, distracting, complex, and high-tempo chores. For example, agents can not only keep up with real-time tagging of individual flows, but can also work continuously in the background to discover higher-level patterns, such as significant deviations from expected network traffic levels. Agents can help monitor background chat sessions, extract data of interest such as IP addresses that match certain criteria, and automatically enrich information about those addresses by looking up additional metadata. As a help with analyst reporting tasks, agents can also collect specified types of information concerning workflow and investigation results into a tool we call the CogLog. The CogLog is a semantic Wiki-based tool prototype within Sol available to agents and analysts, in which they can keep a log of findings pertinent to a given investigation. Things recorded can range from the mundane (e.g., IP addresses, names, pictures) to more abstracted entities like lines of inquiry or "blind alleys." Stores of CogLogs represent an important kind of a kind of knowledge management for analytic work. Affairs associated with each case are automatically (as well as manually) logged and maintained as an analyst might need to jump around from chore to chore, and from case to case. They can be invaluable as cross-reference in future investigations.
- *Increasing resilience.* Agents can increase system resilience by giving advance warning of network problems or analytic slowdowns through displays such as the "Fishtank" discussed above. Agents also aim to assure graceful, robust, and adaptive performance in the face of stressors and surprise through a combination of the principles of *organic resilience*, coupled with the capability for *semantically-rich policy governance*.

The notion of *organic resilience* [12] was inspired by the concept of "organic computing" proposed in [37]. Organic resilience relies heavily on biologically-inspired analogues and self-organizing strategies for the management and defense of distributed complex systems. Carvalho *et al.* have previously applied the concept for the defense of tactical communication systems [12] and mission-critical cloud applications [9]. Multi-layer defense frameworks following the same principles were later developed for critical infrastructure protection and distributed control systems [7, 13, 14]. These infrastructures included humans as an integral part of the system, working in collaboration with software agents to improve system resilience. This highly-successful and innovative approach is well suited to applications such as the one described in this article.

The use of *semantically-rich policy governance* to help achieve organic resilience builds on our contributions to the DARPA Ultra*Log program. In that effort, IHMC's KAoS Policy Services Framework ([47, 48]; see also Sect. 5) was used in conjunction with software agents [3] to assure the scalability, robustness, and survivability of logistics functionality in the face of information warfare attacks or severely constrained or compromised computing and network resources [34]; see also [35]. We have also drawn on concepts and an initial implementation of the notion of collective obligation policies by van Diggelen *et al.* [49, 50].

Because the latest evolution of our approach to increasing resilience is currently the subject of active research and has not yet been fully implemented, we sketch its major elements only briefly.

As with many biological systems, the goal of an organic resilience approach is to avoid static and centralized single-point-of-failure solutions for organizing work to the greatest degree practical. Thus, although groups of agents within the system are collectively responsible for jointly executing various tasks, the specific responsibilities assigned to agents are not completely sorted out in advance. The goal is to allow the agents to self-organize within the constraints of their individual capabilities and current availability. As described by Carvalho *et al.* [9, 11, 12], our research on organic resilience involves understanding the advantages and disadvantages of particular techniques for self-organization for different problems within a given situation and computing environment.

The use of *collective obligations* is critical for practical applications of organic resilience. Whereas an individual obligation is a policy constraint that describes what must be done by a particular individual, collective obligations are used to explicitly represent a given agent's responsibilities within a group to which they belong, without specifying in advance who must do what. In other words, in a collective obligation, it is the group as a whole that becomes responsible, with individual members of the group sharing the obligation at an abstract level.

The execution and enforcement of collective obligations requires different mechanisms for different contexts. For some applications, a specialized planning system spanning a group of agents may be the best approach. However, in this case our commitment to a biologically-inspired approach requires that the agents themselves, rather than some centralized capability, do this work. In most cases, the agents themselves are in the best position to detect local triggers for collective obligations (e.g., potential threats or opportunities), to determine what support they can offer through their own resources and individual capabilities, and what information should be shared among peers and with agents elsewhere in the system. The self-organizing nature of the approach enables the agents to retune responsibilities and resource allocations themselves on an ongoing basis.

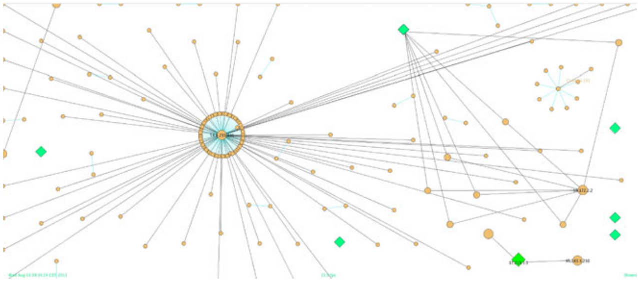


Fig. 6 Agent Learning Results

- *Learning.* Agents can augment human pattern recognition by learning new threat patterns and presenting them to the analyst for validation. For instance, in order to identify additional attacks and targets that analysts may have missed, a group of attacking flows and their targets could be selected, and an agent that uses biologically-inspired learning mechanisms [8, 51] could be launched to find additional, similar flows. Figure 6 shows an example where the learning agent has posted its results to a connectivity graph display. The green node at the upper right-of-center represents one of the analyst's own power plants along with the tan-colored attackers and their presumed command-and-control node. At the lower right is a green node that is a likely next target, due to the fact that it is now experiencing scan attacks from two tan nodes and has the same configuration and vulnerabilities as the first power plant. The large node just to the left of center is another likely target that sits outside our own network. In this way, agent learning has can help the analyst discover additional attacks and potential new targets that otherwise might have been overlooked.
- *Making connections.* Agents can implement capabilities for making connections by continuously doing knowledge discovery: looking for relationships among items of data, people, cases, analysts' activities, and lines of inquiry across individuals and groups of analysts. For example, a KAoS obligation policy (see more on policies below) can be defined to enable the automatic creation and commissioning of a new agent to look for additional data or metadata relevant to a set of flows whenever the analyst

makes a selection using a pointer gesture. As a result, the agent might signal to the analyst that others are also working on related threats when it discovers a given IP address in a live chat interface or within a previous case record in the CogLog.

- *Intelligent reporting.* Agents can provide active, actionable information by generating advisories, indications, and warnings in the form of intelligent, dynamic, multimedia components that can be shared remotely. For instance, in order to notify the power plants who are likely next targets of attack as discussed in the learning example above, the analyst can graphically select the nodes in question and send what we call a "live advisory" in order to notify, and even provide active assistance to, remote colleagues. A "live advisory" is an agent that contains not "just the facts" of a situation, but also contain active analytic tools, views, and capabilities useful in ongoing monitoring and response to a threat. In addition, both analyst expertise and automated learning of new patterns are embodied in the live agents sent to colleagues, rather than buried in "dead" notes and reports. Once remote colleagues receive a Live Advisory, they can open it up (in compliance with the constraints of security policy) to view the rationale, replay the relevant data—and, potentially, launch protective actions.
- *Dynamic scalability.* Agents enhance system scalability by adapting to highly-distributed, rapidly-reconfigurable service-oriented computing architectures. Extensibility to new kinds of threats is as easy as plugging in a new agent—or adding new behaviors to existing ones. Moreover, because of the inherent capabilities of an agent platform coupled with the overall Sol architecture, virtually every aspect of system performance can be multiplied in proportion to the amount of distributed computing resources available. Dynamic reconfiguration of processing among different servers or between clients and servers is made possible by Luna's capabilities for state migration (see more below). The highly-efficient parallel processing enabled by new graphics-chip-level computational approaches (e.g., OpenCL—see [27]) is fully exploited.

5 Sol Implementation Frameworks: Luna Agents, VIA, and KAoS Policy Services

Much of the flexibility and power of Sol comes from its incorporation of three IHMC implementation frameworks. KAoS Policy Services and the VIA Cross-Layer Substrate are mature frameworks that have been used in a variety of previous applications. Luna is a new agent framework that we have optimized for the demanding requirements of applications such as cyber security. Efficiency, flexibility, and security are its hallmarks.

The Luna Agent Framework. IHMC has been a pioneer in the field of software agents, and we have applied our long years of experience to the design of the Luna framework (e.g., [4, 14]).

As summarized in the examples described earlier, Luna agents function both as interactive assistants to analysts and as continuously-running background aids to data processing and knowledge discovery. Though implemented using standard programming languages, Luna agents are more powerful than conventional software because of built-in capabilities that allow them to be proactive, collaborative, observable, and directable [2, 3, 28, 29].

In order to support dynamic scalability and other features of the Sol framework, the Luna platform supports the option of allowing agents to migrate between operating environments and hosts. Unlike the more common platforms that support only a weak form of mobility, Luna supports *weak mobility*, where agents can move while preserving essential aspects of their execution state, and both *voluntary* and *forced mobility* where, completely transparent to them, agents may be moved from one system to another by an external asynchronous request. Since only agent *execution state* is moved, not the agent software itself, the Luna platform is protected from the security vulnerabilities of typical code migration approaches to agent mobility.

The KAoS Policy Services Framework and the VIA Communications Substrate. Because agents are powerful, we use powerful policy management and enforcement frameworks to govern their actions. The KAoS Policy Services framework [47, 48] was the first to offer an ontology-based approach (OWL 2) to policy representation and reasoning. It is currently the most successful and mature of all such efforts. In a policy language overview presented to the US Government Digital Policy Management Standards Subgroup, KAoS was highlighted as the "recommended policy ontology starting point" [52], and IHMC is collaborating with the DPM effort to refine a common core policy ontology based on the KAoS implementation.

KAoS ensures that the Luna agents respect all security and privacy policies, that they respond immediately to human redirection, and that they have the teamwork knowledge they need to work with analysts and other agents collaboratively. KAoS policies also ensure that the entire system adapts automatically to changes in context, environment, task reprioritization, or resources. New or modified policies can be made effective immediately.

VIA [10, 11] is a next generation cross-layer communications substrate for tactical networks and information systems. Operating below the network layer, VIA enables applications to adapt and leverage the characteristics of the dynamic communication environment and enables the underlying communications infrastructure to better support application QoS requirements and constraints.

KAoS Policy Services and the VIA substrate allow desired agent and system behavior to be enforced from top to bottom. Indeed, Luna was built from the ground-up in order to be able to take advantage of the security and responsiveness provided by such comprehensive policy-based control. KAoS allows high-level policies at the level of analyst intent to be dynamically mapped to specific methods at the detailed task level for the realization of that intent. VIA allows fine-grained enforcement of policy down to system-level operations such as the opening of a socket and the monitoring and filtering of specific elements of agent messaging.

6 Concluding Reflections on Tools for Sensemaking

Current research on human sensemaking (e.g., [36]) typically focuses on the ways to shape the sensemaker's investigative démarche in order to help them counteract lines of reasoning that can lead to misconceptions. What such work has failed to adequately consider until now is the impact of new forms of visualization and automation on the sensemaking process, and how such tools ought to be designed in light of what we already know about such things. The work on Sol breaks new ground in theory and implementation by putting questions about the role and benefits of computer assistance to people in center stage. We intend to address some of these questions in future research and performance assessment studies.

In light of the current emphasis on multi-method approaches within the sensemaking literature, the question for the system designer becomes not only "How can we help analysts know whether their hypotheses are correct?" but also "How can we, to the greatest possible degree, use visualization, automation, and collaboration tools to help them expose their hypotheses to the light of experience in order to evaluate and refine them as thoroughly as possible?" In the complex and high-tempo world in which we live, we cannot afford anything less than full engagement of the perceptual strengths, experience, and know-how manifested in both humans and automation as we grapple with the high-consequence problems of the future.

The innovations in human-agent collaboration embodied in Sol suggest significant new directions in automated assistance for sensemaking. Not only can the approach embodied in Sol lead to a qualitative improvement in cyber security effectiveness, but its approach is equally relevant to other applications of distributed sensemaking for other kinds of complex high-tempo tasks.

References

1. Bergen JR (1991) Theories of visual texture perception. In: Regan D (ed) Spatial vision: vision and visual dysfunction, vol 10. CRC Press, Boca Raton, pp 71–92
2. Bradshaw JM, Feltovich P, Johnson M (2011) Human-agent interaction. In: Boy G (ed) Handbook of human-machine interaction. Ashgate, London, pp 283–302
3. Bradshaw JM (1997) An introduction to software agents. In: Bradshaw JM (ed) Software agents. AAAI Press/MIT Press, Cambridge, pp 3–46
4. Bradshaw JM (ed) (1997) Software agents. AAAI Press/MIT Press, Cambridge
5. Bradshaw JM, Carvalho M (2011) Policy services in the cloud: Leveraging dynamically-bounded emergence. In: Workshop on safe in the clouds: biologically-inspired approaches to system resilience and security, Ocala, FL, March 2011
6. Bratman J, Shvartsman M, Lewis RL, Singh S (2010) A new approach to exploring language emergence as boundedly optimal control in the face of environmental and cognitive constraints. In: Proceedings of the 10th international conference on cognitive modeling (ICCM)
7. Byrski A, Carvalho M (2008) Agent-based immunological intrusion detection system for mobile ad-hoc networks. In: Proceedings of the 8th international conference on computational science, Part III (ICCS '08). Springer, Berlin, pp 584–593
8. Carvalho M (2009) A distributed reinforcement learning approach to mission survivability in tactical MANETs. In: Proceedings of the 5th annual workshop on cyber security and information intelligence research (CSIIRW '09), New York, NY, USA, pp 1–4
9. Carvalho M, Dasgupta D, Grimaila M, Perez C (2011) Mission resilience in cloud computing: a biologically inspired approach. In: Proceedings of the sixth international conference on information warfare and security
10. Carvalho M, Granados A, Perez C, Arguedas M, Winkler R, Kovach J, Choy S (2009) A cross-layer communications susbtrate for tactical environments. In: P McDermott, L Allender (eds), Chap 5: Collaborative technologies alliance. Advanced decisions architecture
11. Carvalho M, Granados A, Usbeck K, Loyall J, Gillen M, Sinclair A, Hanna JP (2011) Integrated information and network management for end-to-end quality of service. In: Proceedings of MILCOM
12. Carvalho M, Lamkin T, Perez C (2010) Organic resilience for tactical environments. In: 5th international ICST conference on bio-inspired models of network, information, and computing systems (Bionetics), Boston, MA, December
13. Carvalho M, Perez C (2011) An evolutionary multi-agent approach to anomaly an evolutionary multi-agent approach to anomaly detection and cyber defense. In: Proceedings of the 7th annual workshop on cyber security and information intelligence research (CSIIRW '11), New York, NY, USA, September. ACM, New York
14. Carvalho M, Rebeschini M, Horsley J, Suri N, Cowin T, Breedy M (2005) MAST: intelligent roaming guards for network and host security. Scientia 16(2):125–138
15. Systems Cisco (2007) Netflow services solution guide. http://www.cisco.com/en/US/docs/ios/solutions_docs/netflow/nfwhite.pdf
16. Eskridge TC, Lecoutre D, Johnson M, Bradshaw JM (2009) Network situation awareness: a representative study. In: Proceedings of the fourth workshop on human-computer interaction and visualization (HCIV 2009), Kaiserslautern, Germany, 2 March 2009
17. Feltovich P, Bradshaw JM, Jeffers R, Suri N, Uszok A (2004) Social order and adaptability in animal and human cultures as an analogue for agent communities: toward a policy-based approach. In: Engineering societies in the agents world IV. Lecture notes in artificial intelligence, vol 3071. Springer, Berlin, pp 21–48
18. Feltovich PJ, Bradshaw JM, Clancey WJ, Johnson M (2006) We regulate to coordinate: limits to human and machine joint activity. In: Proceedings of ESAW 2006, Dublin, Ireland, 6–8 September 2006

19. Feltovich PJ, Bradshaw JM, Clancey WJ, Johnson M, Bunch L (2008) Progress appraisal as a challenging element of coordination in human and machine joint activity. In: Artikis A, O'Hare GMP, Stathis K, Vouros G (eds) Engineering societies in the agents world VIII. Lecture notes in computer science. Springer, Heidelberg, pp 124–141
20. Ford KM, Bradshaw JM, Adams-Webber JR, Agnew NM (1993) Knowledge acquisition as a constructive modeling activity. In: Ford KM, Bradshaw JM (eds) Knowledge acquisition as modeling. Wiley, New York, pp 9–32
21. Cabri G, Leonardi L, Zambonelli F (2000) Weak and strong mobility in mobile agents applications. In: 2nd international conference and exhibition on the practical application of Java, April 2000
22. Hoffman R, Feltovich P, Ford KM, Woods DD, Klein G, Feltovich A (2002) A rose by any other name… would probably be given an acronym. IEEE Intelligent Systems, July–August 2002, pp 72–80
23. Holland JH (1998) Emergence: from chaos to order. Addison-Wesley, Reading
24. Johnson M, Bradshaw JM, Feltovich P, Jonker C, van Riemsdijk B (2011, in press) The fundamental principle of coactive design: interdependence must shape autonomy. In: Proceedings of COIN. Springer, Berlin
25. Johnson NE (1989) Mediating representations in knowledge elicitation. In: Diaper D (ed) Knowledge elicitation: principles, techniques and applications. Wiley, New York
26. Kaplan A (1963) The conduct of inquiry. Harper & Row, New York
27. Khronos Group (2011) http://www.khronos.org/opencl/
28. Klein G, Feltovich PJ, Bradshaw JM, Woods DD (2004) Common ground and coordination in joint activity. In: Rouse WB, Boff KR (eds) Organizational simulation. Wiley, New York, pp 139–184
29. Klein G, Woods DD, Bradshaw JM, Hoffman R, Feltovich P (2004) Ten challenges for making automation a team player in joint human-agent activity. IEEE Intell Syst 19(6):91–95 November–December
30. Langton CG (ed) (1989) Artificial life. Santa Fe institute studies in the sciences of complexity, vol 6. Addison-Wesley, Reading
31. Leibowitz H, Shupert CL (1984) Low luminance and spatial orientation. In: Proceedings of the tri-service aeromedical research panel fall technical meeting. NAMRL monograph, vol 33. Naval Aerospace Medical Research Laboratory, Pensacola, pp 97–104
32. Leibowitz H, Shupert CL, Post (1984) The two modes of visual processing: implications for spatial orientation. In Peripheral vision horizon display (PVHD), NASA conference publication 2306 (pp 41–44). Dryden Flight Research Facility, NASA Ames Research Center, Edwards Air Force Base, CA
33. Lind M (1996) Perceiving motion and rigid structure from optic flow: a combined weak-perspective and polar-perspective approach. Percept Psychophys 58:1085–1102
34. Lott J, Bradshaw JM, Uszok A, Jeffers R (2004) Using KAoS policy and domain services within Cougaar. Presented at the Proceedings of the open Cougaar conference, New York City, NY, 20 July 2004, pp 89–95
35. Loyall J, Gillen M, Paulos A, Bunch L, Carvalho M, Edmondson J, Schmidt D, Martignoni A III, Sinclair A (2011) Dynamic policy-driven quality of service in service-oriented information management systems. Softw Pract Exp 41(12):1459–1489
36. Moore DT (2011) Sensemaking: a structure for an intelligence revolution. Clift series on the intelligence profession. National Defense Intelligence College, Washington
37. Müller-Schloer C (2004) Organic computing on the feasibility of controlled emergence. In: Proceedings of the international conference on hardware/software codesign and system synthesis, CODES + ISSS '04. IEEE Comput Soc, Washington, pp 2–5
38. Pollick FE (1997) The perception of motion and structure in structure-from-motion: comparisons of affine and Euclidean formulations. Vis Res 37:447–466
39. Siddiqi K, Tresness KJ, Kimia BB (1996) Parts of visual form: psychophysical aspects. Perception 25:399–424
40. Smith CF (2008) The effect of functional display information on the acquisition and transfer of novice piloting knowledge. PhD Dissertation in psychology. George Mason University, Fairfax,
41. Smith CF, Boehm-Davis DA (2005) Improving novice flight performance using a functional flight display. In: Proceedings of the international symposium on aviation psychology 13th annual meeting, Oklahoma City, OK
42. Smith CF, Fadden S et al (2005) Use of a functional avionics display under varying conditions of workload. In: Proceedings of the human factors and ergonomics society 49th annual meeting, Orlando, FL
43. Still DL, Temme LA (2003) OZ: A human-centered computing cockpit display. In: Interservice/industry training, simulation & education conference (I/ITSEC), Orlando, FL
44. Still DL, Eskridge TC, Temme LA (2004) Interface for non-pilot UAV control. In: Cooke NJ (ed) Human factors of UAVs workshop, Mesa, AZ
45. Temme LA, Still DL, Acromite M (2003) OZ: a human-centered computing cockpit display. In: 45th annual conference of the international military testing association, Pensacola, FL, pp 70–90
46. Thibos LN, Still DL, Bradley A (1996) Characterization of spatial aliasing and contrast sensitivity in peripheral vision. Vis Res 36:249–258
47. Uszok A, Bradshaw JM, Breedy MR, Bunch L, Feltovich P, Johnson M, Jung H (2008) New developments in ontology-based policy management: increasing the practicality and comprehensiveness of KAoS. In: Proceedings of the 2008 IEEE conference on policy, Palisades, NY, 2008
48. Uszok A, Bradshaw JM, Lott J, Johnson M, Breedy M, Vignati M, Whittaker K, Jakubowski K, Bowcock J (2011) Toward a flexible ontology-based approach for network operations using the KAoS framework. In: Proceedings of MILCOM 2011, pp. 1108–1114
49. van Diggelen J, Bradshaw JM, Johnson M, Uszok A, Feltovich P (2009) Implementing collective obligations in human-agent teams using KAoS policies. In: Proceedings of workshop on coordination, organization, institutions and norms (COIN), IEEE/ACM conference on autonomous agents and multi-agent systems, Budapest, Hungary, 12 May
50. van Diggelen J, Johnson M, Bradshaw JM, Neerincx M, Grant T (2009) Policy-based design of human-machine collaboration in manned space missions. In: Proceedings of the third IEEE international conference on space mission challenges for information technology (SMC-IT), Pasadena, CA, 19–23 July
51. VanderHorn N, Haan B, Carvalho M, Perez C (2010) Distributed policy learning for the cognitive network management system. In: The 2010 military communications conference—unclassified program—cyber security and network management (MILCOM 2010-CSNM), San Jose, California, USA, November
52. Westerinen A (2011) Digital policy management: policy language overview. Presentation at the DPM meeting, 19 January 2011, updated 27 March, 2011
53. Yin X, Yurcik W et al (2004) VisFlowConnect: netflow visualizations of link relationships for security situational awareness. In: Proceedings of the 2004 ACM workshop on visualization and data mining for computer security, Washington DC, USA. ACM, New York

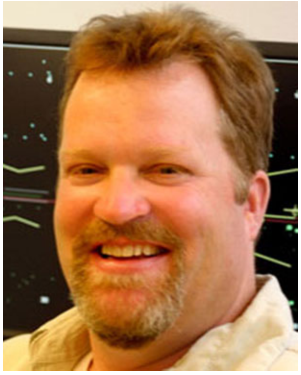
Jeffrey M. Bradshaw (Ph.D., Cognitive Science, University of Washington) is a Senior Research Scientist at the Florida Institute for Human and Machine Cognition (IHMC) where he leads the research group developing the KAoS policy and domain services framework. With Marco Carvalho, he co-leads the group developing IHMC's Sol Cyber Framework. Formerly, Jeff led research groups at The Boeing Company and the Fred Hutchinson Cancer Research Center. He helped pioneer the research area of multi-agent systems, and his first book on the topic, *Software Agents*, became a classic in the field and a best-seller for The MIT Press.

Jeff has been a Fulbright Senior Scholar at the EURISCO in Toulouse, France; an Honorary Visiting Researcher at the University of Edinburgh, Scotland; a visiting professor at the Institut Cognitique at the University of Bordeaux; is former chair of ACM SIGART; and former chair of the RIACS Science Council for NASA Ames Research Center. He served as a member of the National Research Council Committee on Emergent Cognitive Neuroscience and as a scientific advisor to the HCI and Visualization program at the German National AI Research Center (DFKI).

Jeff currently serves as a member of the Board on Global Science and Technology for the National Research Council and as an external advisory board member of the Cognitive Science and Technology Program at Sandia National Laboratories. He is a member of the Technical Committee for IEEE Systems, Man and Cybernetics. Jeff served for over a decade on the Board of Directors of the *International Foundation for Autonomous Agents and Multiagent Systems*, and is a co-organizer of the Human-Agent-Robot Teamwork (HART) workshop series. With Robert Hoffman and Ken Ford, he serves as co-editor of the Human-Centered Computing Department for *IEEE Intelligent Systems*, and co-edited *Essays on Human-Centered Computing* (with Robert Hoffman, IEEE Press, 2012).

Marco Carvalho is an Associate Professor at the Department of Computer Sciences at the Florida Institute of Technology. He also holds a Research Scientist position at the Florida Institute for Human and Machine Cognition (IHMC). Dr. Carvalho received his Ph.D. from Tulane University in New Orleans, following a M.Sc. in Computer Science from the University of West Florida, a M.Sc. in Mechanical Engineering from the Federal University of Brasilia (UnB), and a B.Sc. in Mechanical Engineering, also from UnB. Dr. Carvalho has participated and led several research projects sponsored by the U.S. Army Research Laboratory, the U.S. Air Force Research Laboratory, the National Science Foundation, and Industry. His research interests are primarily in the areas of biologically inspired security and computer network defense. Other research activities also include projects in the areas of tactical communication systems, wireless sensor network security, and information management systems. Dr. Carvalho is a permanent member of the editorial board of the *Scientia* Magazine (ISBN 0104-1770), and has also acted as a Program Committee Member for several conferences and journals. He is an Associate Editor for the IEEE Transactions on Systems, Man and Cybernetics: Part B, and the organizer of several conferences and workshops in the areas of distributed systems resilience, smart grids, and biologically inspired resilience.

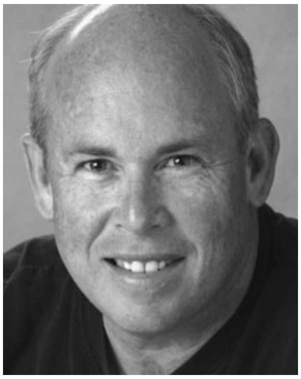
Larry Bunch received his B.S. in computer science from the University of West Florida in 1992. He joined IHMC in 2002 where his research interests include human-agent teamwork, knowledge representation and reasoning, and cyber defense. Larry has published many papers on these topics including several book chapters and journal articles concerning his extensive work with AFRL, ARL, and NASA.



Tom Eskridge is a Research Associate at the Florida Institute for Human and Machine Cognition. At IHMC, Tom has been working on the OZ human centered cockpit display, the SOL cyber security architecture, the CmapTools knowledge modeling system, and the COE Ontology Editor extensions to CmapTools. Previous to IHMC, he founded a company that for 11 years investigated applying artificial intelligence techniques to production quality problems, and produced a line of in-line production visual inspection machines. Mr. Eskridge holds a patent for the VLSI design of neural elements, two for user interface designs, and 5 other patents relating to machine vision and classification. He has over 40 refereed publications. His research interests and experience include models of human analogical reasoning, machine learning, classification, knowledge discovery and data mining, situated cognition, knowledge representation, software and real world agents and foundational issues in artificial intelligence, cognitive science and philosophy. He is completing a Ph.D. in Philosophy with an emphasis on artificial intelligence and cognitive science from Binghamton University and holds an M.S. in Computer Science from Southern Illinois University.

Paul J. Feltovich is a Research Scientist at the Florida Institute for Human and Machine Cognition, Pensacola, FL. Dr. Feltovich received a B.S. in mathematics from Allegheny College (PA) and a Ph.D. in educational psychology from the University of Minnesota in 1981. He was a post-doctoral fellow in cognitive psychology at the Learning, Research, and Development Center, University of Pittsburgh, from 1978 to 1982. He was a professor in Medical Education and Psychiatry (honorary) at Southern Illinois University School of Medicine, 1982–2001. He has conducted research on topics such as expert-novice differences in complex cognitive skills, conceptual understanding and misunderstanding for complex knowledge, and novel means of instruction in complex and illstructured knowledge domains (Cognitive Flexibility Theory and Reductive Bias Theory—with Rand Spiro and Richard Coulson). Since joining IHMC in 2001, he has been investigating (with Jeff Bradshaw and the KAoS team) coordination, regulation, and teamwork in mixed groups of humans and intelligent software agents. The work also addresses Human-Agent coordination in mixed teams and factors that contribute to making software agents acceptable to humans as partners in complex and consequential work. He has authored more than one hundred and twenty professional articles and three books. In particular, he is co-author (with Micki Chi and Robert Glaser) of a designated *Science Citation Classic* article on problem solving in physics which has contributed to the development of human expertise as a field of study in cognitive science. Feltovich was chosen to write one of the articles on expertise for the *Third International Encyclopedia for the Social and Behavioral Sciences*, and has co-edited (with Anders Ericsson, Neil Charness, & Robert Hoffman) the first ever *Cambridge Handbook on Expertise and Expert Performance.* He is also co-editor (with Ken Ford and Robert Hoffman) of *Expertise in Context: Human and Machine (*AAAI/MIT) and (with Ken Forbus) *Smart Machines in Education* (AAAI/MIT).

Matt Johnson has worked at the Institute for Human and Machine Cognition in Pensacola Florida since 2002. He received his B.S. in Aerospace Engineering from the University of Notre Dame in 1992, a M.S. in Computer Science from Texas A&M—Corpus Christi in 2001, and is working on his Ph.D. in Computer Science with TU Delft. Prior to working for IHMC, he spent ten years in the Navy flying both fixed and rotary wing aircraft. He has worked on numerous projects including the Oz flight display for reducing the cognitive workload in the cockpit, Augmented Cognition for improving human performance, and several human-robot coordination projects for both NASA and the Department of Defense. He has worked on advanced robotic control projects such as the DARPA Little Dog project developing walking algorithms for a quadruped robot on rough terrain and the IHMC lower body humanoid developing low-gravity walking gaits for NASA. He has developed advanced information sharing systems designed to support the forward deployed soldier. Most recently, he has been working on developing interfaces to enable micro-air vehicle control in complex urban environments. Matthew's research interests focus on improving performance in human-machine systems and include the areas of teamwork, coordination and human-machine interaction.



Dan Kidwell is a computer system researcher with the DoD, has over 35 years of corporate, consulting and academic experiences with human-centered and information systems design. His expertise & stories have guided him in working at the intersections of industrial design, cognitive systems engineering, computer science, and human-computer interaction design. His current thrust is developing collaborative decision-support tools using software agent technology with security data visualizations in network operation centers.