INTERNATIONAL JOURNAL OF INTELLIGENT CONTROL AND SYSTEMS
VOL. 14, NO. 1, MARCH 2009, 104-114

Design of an Agent-based Course of Action (COA) Analysis with Radio Effects Toolbox

Robert M. MCGRAW, Gary SHAO, Dean MUMME, Richard A. MACDONALD

Abstract - **RAM Laboratories is developing a distributed agent-based toolbox known as the Course of Action Analysis with Radio Effects Toolbox (CARET). The purpose of this toolbox is to provide agent driven simulation and cognitive modeling capabilities that enable improved analyses and diagnoses of network behavior within a mission context. This paper presents current developments during the second phase CARET prototyping efforts..**

Keywords: **distributed agents, network analysis, mission analysis, simulation, cognitive reasoning**

1. INTRODUCTION

Within the defense community, there is a desire to develop tools and techniques that will enable Commanders to intelligently plan and prepare the battlespace both prior to and during operations. While many techniques, simulations, and tools are available to model, represent, and analyze the operational environment in support of the mission planning and planning support processes, there remains a need to inject high-fidelity real-time data and to utilize high-fidelity models in conjunction with those approaches. In particular, solutions should consider the effects of real-time C4ISR systems and the corresponding RF environment in support of network operations planning and re-planning activities.

Challenges presented by the current network operations planning approach includes the fact that existing planning processes are conservative in nature and don't fully take into account uncertainty presented by forces on the move. To address these needs, future capabilities should rely on an open architecture agent-based system that can provide support for rapid Course-of-Action (COA) Analysis to predict and assess future troop movements, faster-than-real-time network analysis simulation capabilities, calibrated C4I-driven analyses that consider real-time and near-real-time information updates from the operational world, and a variety of cognitive analysis tools to assist the operators by off-loading portions of their cognitive workload in conducting the planning process.

This paper discusses an ongoing effort to provide these capabilities through the development of CARET, a software framework that makes use of distributed agents to gather, process, and inject real-time data into embedded simulations for use in evaluating COAs amidst real-time C4ISR effects. This framework will provide the capabilities needed by Signal Planners to dynamically evaluate current situations and predict future conditions presented by forces on the move through COA simulations, corresponding faster-than-real-time network analysis through agent-driven multi-node simulations, real-time C4I calibrated data updates to consistently update and evaluate the mission and network operations planning processes, and reflective cognitive agents to assist network operations planning in re-planning and re-configuring network topologies in real-time.

2. OPERATIONAL EXAMPLE

An operational example illustrating the potential of CARET for network assessment and re-planning activities is presented as follows. For a given mission plan, specific troops, units, and/or assets are required to address specific mission objectives. Network planners supporting these activities will ensure that these units have the connectivity and Quality of Service (QoS) that they require to meet their mission requirements and ensure mission readiness throughout the operation. For Mobile Ad Hoc Networked (MANET) environments, this may entail ensuring that overlaps in coverage exist between emitters in a manner that allows all participating nodes (units) to be reachable as shown in Figure 1. This figure shows a planned Blue Force movement with network coverage for various transmitters/receivers denoted by the Blue circles (Red Force jamming is depicted by the Red circle).

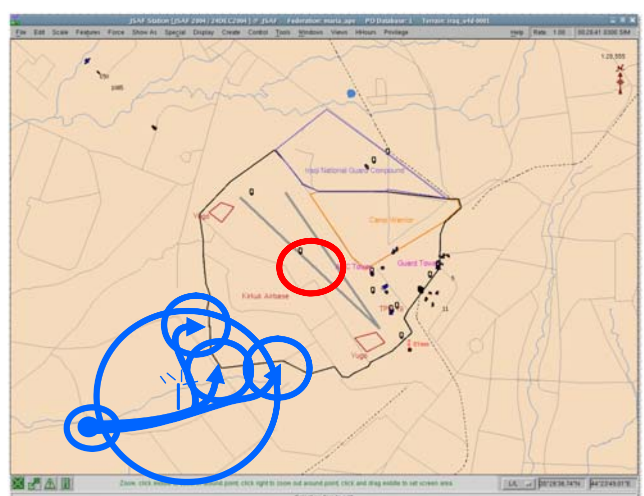


Fig. 1: Mission planning for Blue forces on the move includes ensuring continuing MANET connectivity and QoS performance for participating units.

Manuscript received January 1, 2009. This paper is extended from "A Agent-based Course of Action Analysis with Radio-Effects Toolbox" published at *Spring Simulation Multiconference 2009, Agent-Directed Simulation Symposium*, San Diego, CA, USA, March, 2009.

Robert M. McGraw, Gary Shao, Dean Mumme, Richard A. MacDonald are with RAM Laboratories, Inc. Their email is rmcgraw@ramlabs.com, gshao@ramlabs.com, dmumme@ramlabs.com; rmacd@ramlabs.com, respectively.

The network planning process in support of these activities ensures that the selected network topology operates at the proper frequency or frequencies, achieves the desired level of performance, maintains connectivity as nodes move through varying terrain, and maintains connectivity and desired QoS in the presence of Blue and Red force emitters and electronic countermeasures. Network planning tools and environments, such as Integrated System Control (ISYSCON) and the Coalition Joint Spectrum Management Planning Tool (CJSMPT), utilize emitter databases, network propagation models, and visualization tools in conjunction with output from mission planning tools to plan networks in support of COAs for a given mission.

Existing and emerging network planning tools incorporate information from a wide variety of sources, including data about emitters, topologies, mission types, jammers, and other criteria. Current planning tools stop short of providing real-time network re-planning and reconfiguration of MANET topologies in support of evolving mission operations. The need for such capabilities arises from the fact that mission execution is never ideal. For instance, troops or units may stray from their planned position due to Red force actions or through their own movements to engage emerging targets of opportunity. Figure 2 depicts an altering of Blue force movements from the planned behavior resulting in a loss of network connectivity.

A number of factors can contribute to uncertainty in predicting the outcome of early planning. Radio equipment utilized by units or troops may be subject to interference from terrain obstructions or unpredictable weather conditions. Equipment may be subject to normal wear and tear or suffer damage during battle resulting in degradation of service. These kinds of circumstances highlight the need for technologies that can support the real-time assessment, reconfiguration, and re-planning of networks. The CARET efforts described in this paper addresses some of those needs.

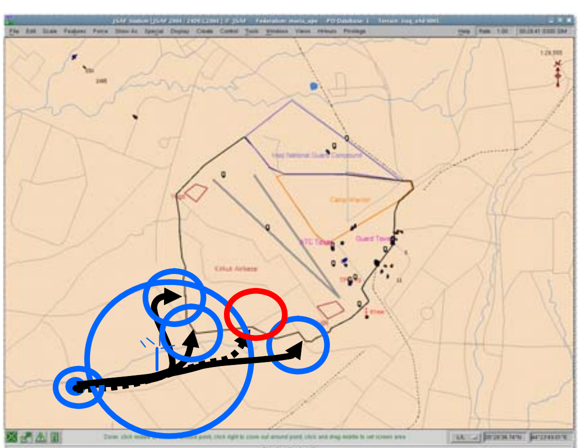


Fig. 2: Effects of Blue force movement and Red force jamming results in momentary loss of network connectivity.

3. CONCEPT OF OPERATIONS

The Concept of Operations (CONOPS) for CARET grew out of John R. Surdu's Simulation in Operations research project and system (OpSim) by which he introduces the concept of operationally-focused simulation [1]. Through this concept, he defends his notion that simulation used in real-time operational environments can be effective in supporting decision-makers. By embedding simulation within an operational setting, decision-makers can use simulation to plan operations, monitor current operations, determine deviations from a plan, predict outcomes, and project different outcomes [2]. CARET's CONOPS uses agent-driven operationally based simulation to assess network behavior within the mission and operational context. The CARET CONOPS leverages functionality developed by RAM Laboratories such as its Dynamic Situational Assessment and Prediction (DSAP) Framework and an Adversary Prediction Environment (APE) [2][3][4][5][6][7][8], and greatly enhances delivered flexibility and adaptability with a new agent-based approach to data acquisition and embedded simulations [9].

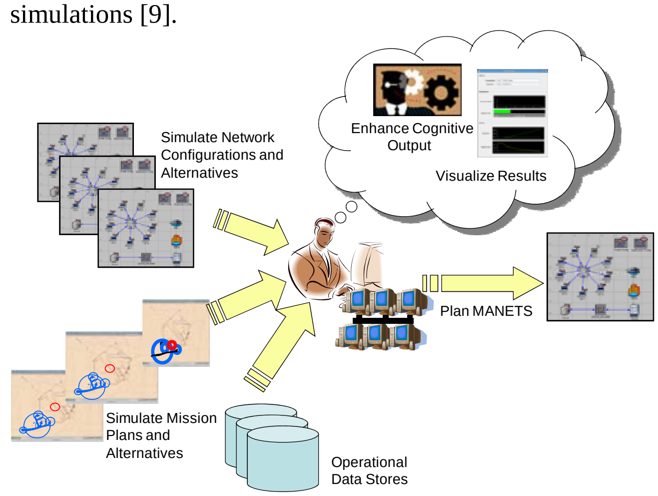


Fig. 3: CARET Operational View depicts convergence of information from multiple data sources, embedded simulation functions, and flexible analyses operations to produce actionable MANET plans.

The Operational View for CARET is shown in Figure 3. Agents are used to provide updates from the actual mission sources by extracting sensor or intelligence data from either databases or near-real-time data sources. Updated data is used to feed mission simulations or trigger agents to drive additional simulations that allow for fully investigating the potential operational space by projecting mission trends or playing out possible mission outcomes. As these simulated missions are playing out, additional agents are capturing the simulated data and relaying that information to agents that drive and analyze the corresponding networks supporting that mission, allowing for the simulation and analysis of a variety of network configurations and investigation of alternative topologies.

Both mission analysis and network analysis results are displayed to signal planners to provide them with a means to evaluate and visualize observed mission performance as missions proceed. Additional agents assist the signal planner in this analysis and diagnosis phase by addressing such issues as equipment degradation, QoS based on Blue force positioning, environmental and atmospheric effects, and issues related to adversary jamming capabilities.

4. CARET DESIGN GOALS

4.1 Design Objectives

In order to address the capabilities called for by the CARET Concept of Operations, the CARET environment will need to provide agents that facilitate a number of these capabilities. In light of this, CARET will provide agents that (1) execute simulations of COAs, (2) calibrate simulations with real-time C4I data, (3) trigger analysis of proposed network architectures in support of COAs via simulation-based network propagation models, (4) query emitter databases and selected C4I databases, and (5) select and filter prospective emitters participating in the environment in question. The CARET prototype will also provide reflective cognitive agents using Bayesian Belief Networks (BBN) and Hidden Markov Models (HMM) that will enable agents to (1) intelligently select or reason through radio parameters and capabilities in cases where there is some uncertainty regarding the available functionality, and (2) intelligently identify or reason through the location and characteristics of potential threats due to jamming and interference that may degrade network connectivity and QoS. The resulting toolbox will better enable the planning/re-planning, analysis, and use of network assets as operations evolve. Moreover, the framework will enable re-planning of network architectures as missions diverge from their original idealized plan.

4.2 Initial Proof-of-Concept

An initial proof-of-concept for CARET was designed and implemented to prove the feasibility of our approach. The proof-of-concept design is depicted in Figure 4.

The proof-of-concept addressed mission analysis and network analysis and reconfiguration capabilities by providing (1) a Simulation Agent that executed simulations of COAs), (2) a Network Analysis Agent that could calibrate network simulations with near-real-time C4I data and network parameters while triggering analysis of proposed network architectures via simulation-based network propagation models, (3) a Real-time Data Agent that could query emitter databases and selected C4I databases, (4) a Simulation Evaluation Agent that could select and filter prospective emitters based on the position of Blue and Red forces participating in the operation/simulation, and (5) various Visualization Agents. A CARET Server agent was also developed to manage the workflow between agents. Placeholders were created for certain agents (RTP Evaluation Agent, the Entity Reasoning Agent, and the Plan GUI) that would not be needed until later phases of development. A high-level overview of the proof-of-concept implementation in support of CARET is shown in Figure 5 and discussed in the following paragraphs.

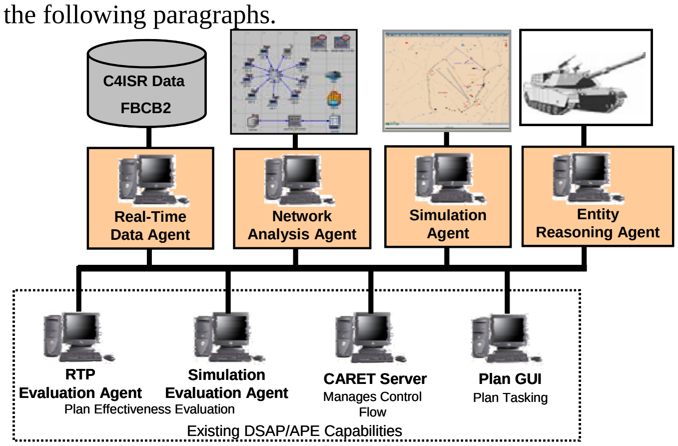


Fig. 4: Agents implemented for CARET proof-of-concept

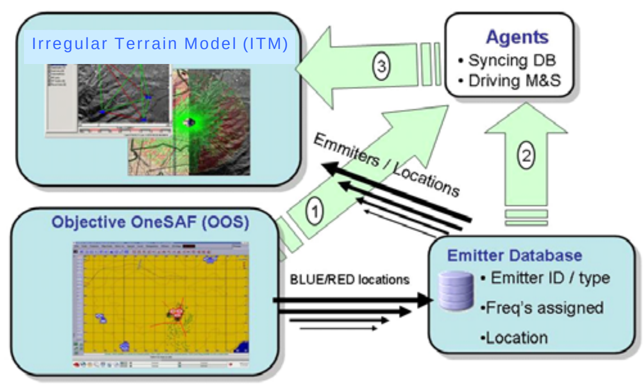


Fig. 5: High level overview of CARET proof-of-concept with participating COTS/GOTS applications

4.2.1 Simulation Agent – OneSAF Objective System

The CARET proof-of-concept implemented a Simulation Agent whose role was to execute a simulation of a given mission or COA. This Simulation Agent acted as a testbed for the CARET proof-of-concept and drove OneSAF entities in execution of the mission plan to represent the “real-world” locations of entities. OneSAF is used to execute a scenario that demonstrates how CARET could be used to evaluate and identify entities and corresponding radio connectivity on the battlefield. This scenario contains scripted behavior that is representative of real world mission execution.

For our proof-of-concept, the scenario used was a slightly modified version of the DIS_TEST demonstration provided with the OneSAF software. The DIS_Test demonstration involves 26 entities and includes a variety of Tanks, Helos, Infantry and Support vehicles. Our proof-of-concept involved finding an entity with a relatively fixed position (in this case a support vehicle) that utilized a transmitter, and a mobile vehicle to position the receiver over varying terrain (in our case a Tank Platoon Commander). The initialized scenario is shown in Figure 6 with the Support vehicle and Tank Platoon Commander highlighted. The subsequent mobile portion of the scenario is shown in Figure 7 (also with the Supply Vehicle and

Tank Platoon commander highlighted. The scenario has the Helos performing an RWA Attack on the adversary target, and then subsequently following up with a Tank attack on the same target. In executing this Tank Platoon attack, the Tank Platoon Commander moves away from the Supply Vehicle throughout the course of the scenario.

The objective of this proof-of-concept is to demonstrate how CARET agents can be used to identify, select, and parameterize a network analysis (propagation) simulation to extract the expected QoS (in this case, transmission loss) between the transmitter/receiver over the given terrain.

The locations of all entities in the OneSAF scenario are passed to the Simulation Evaluation Agent using the Distributed Interactive Simulation (DIS) simulation interoperability standard. This information is collected and analyzed by the Simulation Evaluation Agent for further processing of entity locations and movement.

4.2.2 Simulation Evaluation Agent

The CARET Simulation Evaluation Agent (SEA) performs several functions including collecting, evaluating and analyzing data from the Simulation Agent (OneSAF) in the proof-of-concept. The SEA has several roles in the proof-of-concept that include: (1) collecting entity locations and movement information from OneSAF, (2) determining which entities provided by OneSAF are contained in the area of interest, (3) initiating a query of the Emitter Database for assigned frequencies and allowed emitter locations, (4) submitting the assigned frequencies and entity locations to the Network Routing and Analysis Agent to determine radio waveform propagation and entity connectivity. Each of these roles is explained in further detail in the following paragraphs.

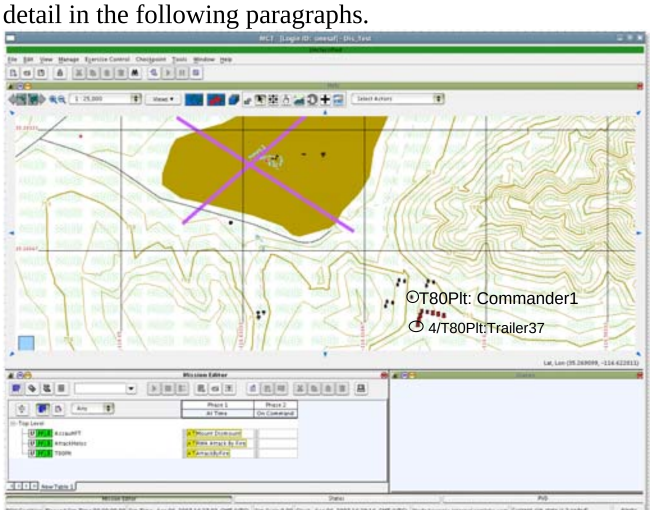


Fig. 4. DIS_Test Scenario with Tank Commander and Support Trailer provides scenario with a single transmitter and multiple mobile receivers.

The SEA collects the entity location and movement information from the OneSAF simulation. Because OneSAF has a Distributed Interactive Simulation (DIS) capability, DIS was chosen as the interface to collect entity locations from OneSAF. This agent utilizes a third party library developed at the Naval Post Graduate School called DisJava to collect and unpack the PDU into a Java object. When a DIS PDU is received, the SEA checks to see if it is the EntityStatePDU. The only PDU that the SEA currently uses is the EntityStatePDU. From the EntityStatePDU the entity’s coordinates, DIS ID, and name are extracted. This information is then passed through the Java Native Interface (JNI) to a C++ high performance computing data structure called the Hierarchical Grid (HiGrid) [10]. This interface is used by the SEA to determine entity locations that are relevant to a desired area.

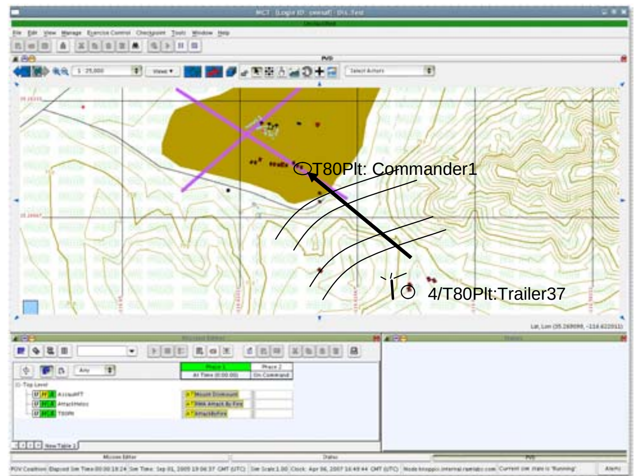


Fig. 5. Mobile receiver on Tank Commander unit exercises proof-of-concept capabilities to track changing mission conditions and perform dynamic network analyses.

The SEA utilizes a HiGrid data structure to efficiently determine if the simulated entities are within a given area of interest. This data structure, developed by RAM Laboratories, is highly optimized for performing proximity calculations and filtering based on a variety of coordinate systems. It stores information on the entity locations and types and compares them to the areas of interest. During the construction of the HiGrid data structure, an XML string identifying the area of interest for the scenario is passed to the HiGrid. The SEA queries the data structure to determine which entities are in the area of interest for the scenario. This information is then used to query the emitter database for the assigned frequencies and allowed emitting locations of entity emitters.

In addition to performing the filtering on data using its HiGrid capability, the SEA is coupled with a visualization agent known as the CARET Entity Locator Agent (ELA). The ELA provides a visualization of the filtered region by allowing the user/operator to “see” only the agents selected by the HiGrid filter. Another feature of the CARET ELA is that it provides corresponding latitude, longitude, and elevation information for each entity selected.

The last role of the CARET SEA is to submit the assigned frequencies, entity locations, and emitter types to a Network Analysis Agent (NAA). The NAA uses this information to determine radio waveform propagation and entity connectivity based on the entity locations, emitter types, frequencies, and terrain.

4.2.3 Real-time Data Agent - Emitter Database

The CARET Real-time Data Agent (RTDA) receives queries from the SEA and returns the characteristics for the emitters requested by that agent. The design calls for connecting to an Emitter Database that contains information on emitter IDs/type, assigned frequencies, and allowed emitting locations. This database is traditionally a hand typed database that requires a human in the loop to assign emitter frequencies. Initial proof-of-concept implementations have emulated this Emitter Database via a flat file as shown in Figure 8. This file contains fields pertaining to the platform name (type), country code, assigned frequency and antenna height.

Tank platoon support trailer units in the proof-of-concept scenario used 500 MHz transmitters with 2.5m high antenna (as specified by the last entry in the table). Subsequent implementations of the RTDA that will be implemented later during the Phase II effort will expand on this design to utilize an SQL database. Agents implemented in support of the SQL model of a Spectrum Knowledge Repository (SKR) database will also perform necessary transformations to bridge gaps between SKR entries and the information required by the network propagation models.

4.2.4 Network Analysis Agent

A CARET NAA performs the high fidelity network analysis of transmitters and receivers for a selected area. The NAA receives transmitter and receiver characteristics, parameters, location and movement information for the radios under analysis. A goal of this effort is to provide this capability as an input to CJSMPT and models used in support of that tool, such as TIREM. TIREM is a set of Fortran simulations that determine radio frequency waveform propagation based on entity location using ray tracing algorithms to determine how waveforms propagate through varying terrain.

```
"AssaultFT:A", 225, 500.0, 2.5
"AssaultFT:F", 225, 500.0, 2.5
"AssaultFT:G", 225, 500.0, 2.5
"AssaultFT:R", 225, 500.0, 2.5
"AttackHelos", 225, 500.0, 100.0
"OOSUH60", 225, 500.0, 200.0
"T80Plt:Comm", 225, 500.0, 2.5
"1/T80Plt:T8", 222, 250.0, 3.0
"2/T80Plt:T8", 222, 250.0, 3.0
"3/T80Plt:T8", 222, 250.0, 3.0
"4/T80Plt:DE", 222, 250.0, 3.0
"4/T80Plt:FI", 222, 250.0, 3.0
"4/T80Plt:SU", 222, 250.0, 3.0
"4/T80Plt:Tr", 222, 250.0, 2.5
"4/T80Plt:Tr", 225, 500.0, 2.5
```

Fig. 8: Example of text file information used to emulate an SKR database. The fields represent: <platform name>,<radio type>,<frequency band>,<antenna height>

As an alternative to TIREM, the proof-of-concept used the Irregular Terrain Model (ITM) as the propagation model in the implementation of the NAA. The ITM model takes in the parameters listed in Table 1 and outputs the transmission loss (using its point-to-point mode) for a given transmitter/receiver pair. Our proof-of-concept converted a Windows C++ version of the Longley-Rice ITM model for use in Linux environments, and used that model for analyzing network propagation. The agent driving the model provided each of the required input values (a number of environmental parameters were fixed) including receiver/transmitter height, location, and characteristics extracted from the Emitter Database emulation. The NAA was configured to query a Globe terrain database for topological information.

The NAA constantly performs a network analysis of the QoS between points for each PDU update in position. This NAA in our proof-of-concept demonstration performed a point-to-point transmission loss calculation each update cycle.

4.2.5 Visualization Agents

The NAA in the proof-of-concept is coupled with a CARET Visualization Agent – Radio (VAR). The VAR provides a visualization of results from calculations of transmission loss between emitters and receivers over time. The VAR also depicts the distance between the emitter and receiver over time to enable the user/operator to visualize the increasing/decreasing loss in signal strength as the mission proceeds. Example results from VAR operations are shown in Figure 9.

Table 1: Parameter Fields and Values for ITM Model

Category	Fields
Surface Refractivity	Equatorial (360), Continental Subtropical (320), Maritime Subtropical (370), Desert (280), Continental Temperate (301), Maritime Temperate – Land (320), Maritime Temperate – Sea (350)
Dielectric Constant of Ground	Average Ground (15), Poor Ground (4), Good Ground (25), Freshwater (81), Seawater (81)
Conductivity of Ground	Average Ground (0.005), Poor Ground (0.001), Good Ground (0.020), Freshwater (0.01), Seawater (5)
Polarization	Horizontal, Vertical
Reliability	1 to 100
Confidence	1 to 100
Frequency	In MHz
Antenna Height (transmit and receive)	In meters
Path Ground (transmitter and receiver)	In LatLon
Terrain	WOTL, Globe

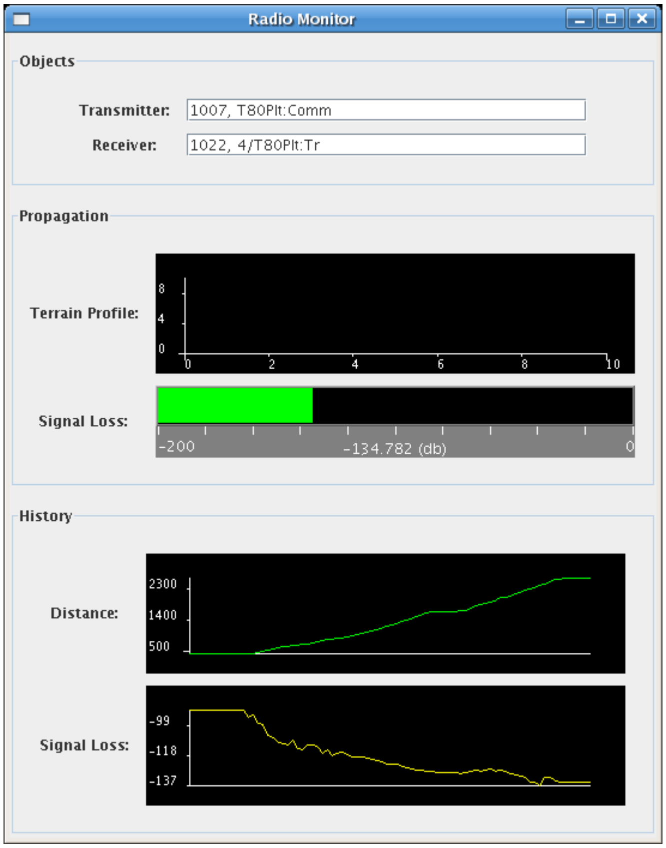


Fig. 9: CARET Visualization Agent – Radio display illustrates that transmission loss increases as distance from Trailer increases for proof-of-concept scenario

4.3 Proof-of-Concept Results

The proof-of-concept was demonstrated using DIS_Test scenario to evaluate transmission loss between the Tank Platoon Commander and the Trailer as depicted in Figure 7. Figures 9 shows the transmission loss experienced as the mission evolved. This report showed that as the receiver (the Tank Platoon Commander) moved away from the transmitter (note the increasing distance), the network propagation results shifted from a line-of-sight to a double-horizon, diffraction dominant result. Additionally, the transmission loss between the transmitter and receiver increased from a loss of 93.56 dB to around 135 dB.

The results of our proof-of-concept proved that CARET could provide a distributed agent-based framework that could be used to augment COA analysis with corresponding network planning activities, provide agents to drive high-fidelity simulations, provide agents to interoperate with multiple simulation and data sources to identify radios characteristics of interest, and provide basic prediction approaches for estimating where radios may be placed at some point in the future. This proof-of-concept provided a demonstration of framework concepts that would be the basis for future full-fledged prototype development.

5. Prototype Design

The second phase of development for CARET involves the development of a full-featured prototype. The design for this prototype builds on the lessons learned during the proof-of-concept and involves the complete definition, design, and implementation of functional agents in CARET.

5.1 Goals for a CARET Agent Framework

The CARET prototype effort encompasses creation of a CARET Agent Framework whose design addresses a number of important goals: (1) applications can be constructed from sets of composable cooperating components, (2) data sources can be accessed through abstract data interfaces supporting the use of metadata and dynamic data introspection, (3) framework provides support for implementing observation-interpretation-action workflow cycles, (4) framework provides support for readily applying modular and flexible analysis functionality, such as cognitive inferencing, (5) framework provides support for actively managing embedded interactive simulations, and (6) framework provides support for interoperability with external applications and services.

5.1.1 Composable Components

Applications developed under the CARET Agent Framework can acquire functionality by composing services offered through components found within each application's environment. New application capabilities and behavior can be achieved with minimal effort by simply substituting compatible alternative service providers into an application. These substitutions may even be performed dynamically in a running system in response to planned or unplanned system events.

5.1.2 Abstract Data Interfaces

To minimize unnecessary dependencies between applications and specific data source interfaces, the CARET Agent Framework employs a general abstract data interface for all data exchanges between application components. The abstract interface supports the attachment and modification of metadata associated with each data object. New application data elements may implement the abstract interface natively, while legacy data sources may operate through compatible proxy servers.

5.1.3 Observation-Interpretation-Action Workflows

The CARET Agent Framework provides direct support for marshalling service components into workflow patterns. Support for workflows following the Observation-Interpretation-Action cycle is especially important for applications providing commanders with views to improve situation awareness of battlefield conditions. Workflow support should be flexible and dynamic, allowing for rapid configuration and reconfiguration of utilized services.

5.1.4 Flexible Functional Analyses

Components executing analysis-related functions within the CARET Agent Framework implement simple functional input and output service interfaces, enabling different analysis elements to be readily plugged in as needed. Analysis services allow chaining different simple services together to form a more complicated analysis services, as well as providing support for decomposing complicated analysis functions into several more manageable pieces.

5.1.5 Embedded Interactive Simulations

The CARET Agent Framework specifically targets applications that rely on running one or more simulation instances to complete designated tasks. Simulation instances developed within the framework present interfaces that allow interactive monitoring of internal simulation states and interactive control over simulation operations.

5.1.6 Interoperable External Interfaces

Applications developed within the CARET Agent Framework are able to interoperate with external applications and services that are compliant with common industry-standard protocols. It is quite possible that application components may support more than one protocol, but at a minimum CARET applications should support W3C-compliant Web Services as mechanisms for interaction with external applications. In many cases the target application in the CARET Agent Framework is not a stand-alone application, but is itself a meta-component presenting a service interface to the outside world.

5.2 CARET Agent Framework Prototype Design

Working from the core principles described in Section 5.1, RAM Laboratories has developed a design for the CARET Agent Framework. The key features of the framework design serve three main purposes: (1) enable the development of highly capable applications that are able to meet the demands of current and future requirements, (2) provide a development platform that is highly flexible and easily extensible, and (3) ensure that applications developed from the framework are highly interoperable with other applications, systems, and platforms, particularly legacy software and systems.

5.2.1 Basic Components

The first principle discussed in Section 5.1 is the need to build applications from composable components. The best choice for a basic software component is one that fits in naturally with today's predominant net-centric leaning towards greater distribution of computing and information resources across different networks. To fit this highly decentralized computing environment, applications should be composable from distributed functional entities that need to be only lightly coupled at their time of use. Software agents appear to a good technological fit to our requirements, and prior experience with agent technology in programs such as CARET gives us confidence that software agents are a good choice as a basic building block for the kinds of applications we are interested in.

To implement our selection of software agents as a base component type, we have selected the Java Agent Development Environment (JADE) [11] as our initial agent development environment. JADE is a software framework to develop agent-based applications in compliance with the Foundation for Intelligent Physical Agents (FIPA) specifications for interoperable intelligent multi-agent systems [12].

5.2.2 Handling Data

The second core principle from Section 5.1 was the use of abstract data interfaces wherever possible. When the numbers and different types of data sources that are available for use by CARET are examined, the rationale for simplifying the burden on application developers to support many different kinds of data interfaces becomes clear. Fortunately, there is a well-suited data interfacing technology already specified that appears to meet our requirements. The Service Data Object (SDO) specification [13] prescribes a way for all compliant data exchanges to take place using a common representation for all data types. Figure 10 shows a UML diagram of the basic SDO representation for data.

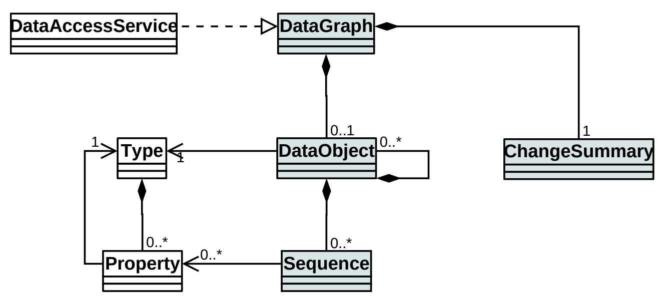


Fig. 10: Service Data Object (SDO) representation provides data abstraction and support for metadata attributes and change logging

With SDO, individual units of data are contained in Data Objects. When data needs to be exchanged between application components, sets of Data Objects are assembled into a Data Graph and sent from a DataAccessService to its final destination. Data Graphs can also keep track of all changes or modifications made to the graph over time in a Change Summary.

Another benefit of adopting SDO as the standard protocol for data exchanges is its explicit support for the use of metadata attributes associated with data and the ability to have clients dynamically apply introspection during data exchanges. This allows clients to discover data-specific properties and potentially adapt data handling activities to specific circumstances as they become apparent.

The choice of SDO for data interfaces may also reduce problems caused by the need for applications to frequently support the quirky ways that legacy data sources may have to be accessed. For legacy data sources that cannot be practically changed to support more modern data standards, SDO translation agents can be written just once for each particular legacy source, thereafter enabling any SDO-compliant component to access the same legacy source.

5.2.3 Interoperability with Everyone Else

There is a clear trend today for software to emphasize compatibility with net-centric environments and service-oriented architectures. Among the many service-oriented software standards available today, Web Services are a popular choice. In order to promote high levels of interoperability between applications developed under the CARET Agent Framework and other external applications and systems, the CARET Agent Framework includes options to incorporate support for Web Service technologies such as SOAP [14] and UDDI for interoperability with external components.

The choice of JADE as the underlying agent development framework benefits us here because the JADE developers have created a Web Services Integration Gateway (WSIG) component that automatically performs two-way translations between web service requests/responses and JADE agent requests/responses. This means that JADE agent entities can invoke web service functionality hosted outside the JADE runtime environment using normal JADE agent protocols, and that external entities can invoke JADE agent functionality from outside the JADE environment using normal web service protocols.

5.2.4 Managing Workflows

The third core principle from Section 5.1 was support for workflows, particularly those following the Observation-Interpretation-Action cycle of programmed system responses to dynamic events. The choice of JADE as the agent development framework provides us with a good starting point to build up from with a complementary product to JADE called the Workflows and Agents Development Environment (WADE). WADE is built on top of JADE, as shown in Figure 11.

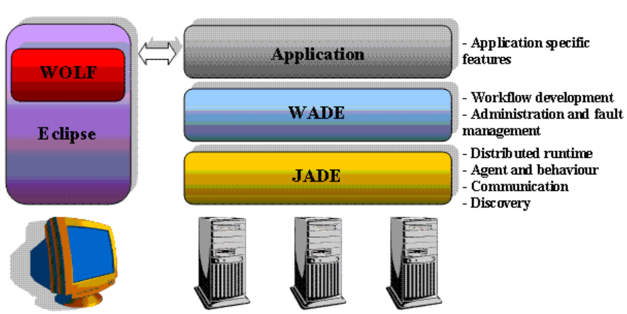


Fig.11: WADE extensions to JADE enable more sophisticated workflow behavior patterns for agent-based applications.

Through the WOLF graphical manager for WADE, which runs within Eclipse, it may be possible to provide graphical tools for application developers to configure workflows for applications developed under the CARET Agent Framework.

5.3 CARET Agent Design and Hierarchy

Once the design principles for the CARET Agent Framework had been laid out, the design of the agents populating CARET began. This design required slight adjustments in the agent definition and agent hierarchy from the initial proof-of-concept development effort. Agents implemented in the CARET prototype are depicted in Figure 12. A definition of each type of each agent class and each underlying agent type are provided below. The basic agent types are Operational Data Agents (ODAs), Analysis Agents (AAs), and Process Agents (PAs). Each of these is derived from a general agent class.

5.3.1 Operational Data Agents

ODAs are responsible for capturing or extracting data from a variety of sources in the CARET environment. Such sources may be SQL databases, real-time data sources, documents such as simple text files or structured documents employing XML, or live simulation environments.

For our prototype, we are implementing two types of ODAs: an ODA DOC to provide capabilities for extracting data from XML documents, and an ODA Sim to provide capabilities to extract data from running simulations. Our implementation has two varieties of ODA Sim agents: ODA Sim agents that extract data from OneSAF Objective System simulations and ODA Sim agents that extract data from network simulations such as the ITM that are used to calculate transmission loss (in the proof-of-concept) or commercial network simulation environments such as Qualnet.

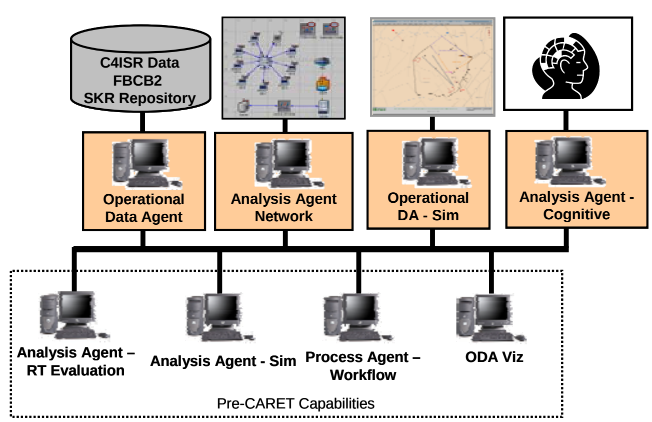


Fig. 12: Agents implemented for the CARET prototype

5.3.2 Process Agents

The second type of agent designed for use in CARET is the PA. PAs are used to implement control of executing processes over time, such as triggering or calibrating multiple simulation runs or managing the flow of data between agents. The CARET prototype uses two types of PAs: PA Workflow and PA Sim Control. The PA Workflow agent manages the flow of data between agents in a business process-oriented fashion. The PA Sim Control Agent manages the execution and calibration of running simulations across available processing resources on a distributed computing grid. For our implementation of the CARET prototype, there are two instances of PA Sim Control agents: ones that manage network simulations (using tools such as Qualnet or more special-purpose RF models), and ones that manage mission simulations (using mission-level simulation environments such as OneSAF).

5.3.3 Analysis Agents

AAs are responsible for performing multiple types of analysis on data in the CARET environment, including performing analysis of live simulation results, analysis of real-time or near-real-time intelligence, surveillance, and reconnaissance sensor data from the operational world, predictive analysis of trends with regard to missions or network configurations supporting active missions, and analyses of particular alternative or “what-if” scenarios in addressing particular mission objectives. The prototype has several AA types including AA Network Analysis and AA Cognitive Reasoning. The AA Network Analysis agent is responsible for evaluating the performance of networks using outputs from live network simulation models or environments. The AA Cognitive Reasoning agent is responsible for applying a variety of cognitive reasoning approaches to analyze or diagnose performance issues identified from outputs of multiple AA Network Analysis agent evaluations.

6. Cognitive Agents

Two key capabilities that CARET will provide signal planners are a capability to analyze and diagnose issues related to network performance as a mission proceeds, and a capability to analyze and diagnose issues related to network performance as a mission is projected forward from a given point in time (using simulation) in order to identify potential outcomes and assess the impact of various mission alternatives.

To provide such a capability to signal planners, CARET utilizes a specific type of cognitive reasoning agent known as an AA-CR. The AA-CR agent works by comparing operational or simulated network behavior against expected patterns of behavior and observed QoS performance for operating networks. The AA-CR agent compares simulated or actual network performance against “ideal” results and then proceeds to diagnose possible problems with the network by following sets of rules which may be contained in a fuzzy rule base, statistical analysis, or through the use of domain-specific inferencing techniques.

Multiple instances of the AA-CR agent are being designed and implemented to address three key areas of reasoning that will enable the off-load of some cognitive functions from network operations planners. These instances are (1) an agent that assists the network operations planner in understanding radio selection based on “known” data and metrics extracted by the network, (2) an agent that assists the network operations planning in identifying and locating jamming capabilities based on “known” data and metrics extracted by the network, and (3) a cognitive agent that assists network operations planners in addressing the re-planning process. Goals of these agent implementations will be to provide cognition that takes into account observable phenomena while addressing uncertainties due to the “fog of war”, forces on the move, and degrading or uncertain equipment choices.

Initial designs for these agents are focusing on the use of fuzzy rule sets and statistical approaches as a way of identifying potential problems for the network in comparison to the expected performance. The statistical metrics are being generated through the use of multiple simulations of the support network with variances added to various parameters to gain an understanding of the state space facing a variety of problems including radio equipment degradation, platform positioning, and dynamic environmental effects.

6.1 Inferencing Techniques

Once sets of rules and statistics have been gathered in support of the cognitive analyses, the design for each PA CR agent will incorporate more advanced techniques such as inferencing to gain further knowledge and guide the signal operator about the cause of certain phenomena. We have researched a variety of models for implementing reflective cognitive models required by this effort [15][16][17][18][19] that considered the use of neural networks, genetic algorithms, simulated annealing approaches, Bayesian networks, and Markov models. Based on the initial research, the initial AA-CR agent design is focusing on utilizing Bayesian Belief Networks and Hidden Markov Models for the prototype implementation. These approaches are being used to implement the reflective cognitive agents in support of radio frequency selection, emitter identification, and diagnosis activities required by the CARET prototype. An example of how such models can be used to provide basic inferencing capabilities is provided below.

6.2 Inferencing Example

Bayesian Belief Networks (BBNs) can be used to provide the one form of cognitive analysis capabilities for AA-CR agents. BBNs are directed acyclic graphs that provide a representation of the joint probability distribution for a group of variables [15]. These models were chosen as a prospective cognitive modeling environment because of their ability to (1) predict the outcome or expected output (in terms of a probability distribution) of expected behavior based on known or estimated probabilities and conditional probabilities, and

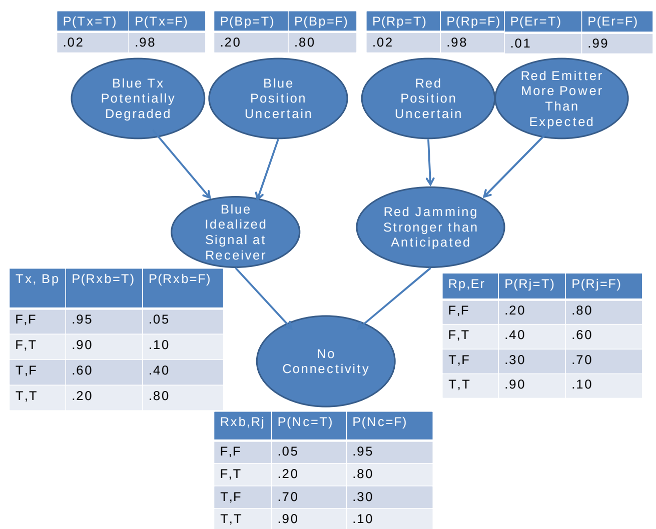
(2) provide a means to "reflect" back on observed behavior (known output values – i.e. network metrics and extracted data regarding transmission loss, attenuation, latency, jitter) and known probabilities and conditional probabilities to identify the probability distribution of "unknown" or "unobservable" nodes through the use of experimentation.

An application of case (1) in this regard would be determining the expected or predicted behavior of a network when considering known environmental parameters, known transmitter and receiver characteristics, the probability distribution of known jamming capabilities, and probability distributions associated with the position of mobile forces as they execute their mission. The BBN implementation of this case would receive information regarding the potential probability distribution for a unit's location during the execution of a plan (such as via OneSAF COA simulations) and potential probability distributions for locations of adversary or mobile jamming capabilities, and then evaluate the observed behavior against expected behavior determined by the AA Network Analysis agent and the ITM or TIREM models. An example of such a BBN is shown in Figure 13, considering a case where a Blue transmitter is fixed but its transmission capability is not 100% certain, the Blue entity holding the receiver is potentially out of position, and the adversary jamming capabilities may be affected by more powerful jammers or a change in position.

For this case, we would set the initial probabilities for the Blue transmission capabilities (Tx), Blue position uncertainty (Bp), Red position uncertainty (Rp), Red emitter power (Re) and expected conditional probabilities for the idealized Blue signal capability (Rxb) and Red jamming strength at the receiver (Rj). Subsequent simulation runs for the COA and Network Analysis simulations can be run to refine the conditional probabilities for several of these nodes. Additional work will be performed to evolve these BBN representations into HMM that evolve as the operation evolves while capturing the history of past COA behaviors, along with an evolving picture of the expected and observed network operational behavior.

In addition to these capabilities, the use of these cognitive models and embedded COA simulation capabilities can be used to provide future support for real-time network re-planning and reconfiguration of mobile networks. COA simulations can be calibrated with real-time data detailing Blue force and adversary position information (as well as updates in radio equipment). As operations evolve and forces potentially move out-of-position, the CARET environment can enable the AA-CR agents and their cognitive models to initiate network re-planning and re-configuration processes that may result in adding additional links (i.e. UAV based links) to ensure continued connectivity during challenging and uncertain conditions.

An application of case (2) with regard to the use of BBN models would consider the identification of the probability distribution or conditional probability distribution for unknown or unobserved nodes in the presence of known output behavior (network metrics and data extracted from a network management system or self-diagnostics). In this particular case, experiments can be performed using the AA - Network Analysis agents in conjunction with the AA-CR supporting this type of cognitive reasoning by using network analysis simulations (via ITM, TIREM, or other RF propagation models) to evaluate potential values/distributions for unknown link values concerning an adversary jamming capability. In this regard, the agent-driven simulation is provided known values of Blue force locations on the move, fixed values or known probability distributions representing environment, terrain, and atmospheric parameters and radio characteristics along with known network link data and behavior. The AA-CR can drive the process for conducting analysis experiments to determine the potential characteristics, location, and parameters of an adversary jamming threat. In this regard, the use of three or more Blue entity radios can serve to help triangulate adversary jammer location and characteristics.



P(Tx=T)	P(Tx=F)
.02	.98

P(Bp=T)	P(Bp=F)
.20	.80

P(Rp=T)	P(Rp=F)
.02	.98

P(Er=T)	P(Er=F)
.01	.99

Tx, Bp	P(Rxb=T)	P(Rxb=F)
F,F	.95	.05
F,T	.90	.10
T,F	.60	.40
T,T	.20	.80

Rp,Er	P(Rj=T)	P(Rj=F)
F,F	.20	.80
F,T	.40	.60
T,F	.30	.70
T,T	.90	.10

Rxb,Rj	P(Nc=T)	P(Nc=F)
F,F	.05	.95
F,T	.20	.80
T,F	.70	.30
T,T	.90	.10

Fig. 13: BBN example shows probabilities of causative events may be used to help diagnose observed network problems

In addition to BBN-base approaches, the CARET prototype will evolve representations into Hidden Markov Models (HMM) to address each of the reasoning types that will be deployed by CARET entity reasoning Agents. The HMM approach utilizes one discrete hidden node and one discrete or continuous observed node for each slice of time [15]. The use of HMM will be especially useful in developing a more advanced reasoning approach because they can not only consider the current probability densities and values for a given scenario, but can also rely on past snapshots with regard to the executing COA simulation (with regard to Blue and Red force position) and past network connectivity, reliability and QoS parameters. The resultant analysis will allow us to reason through observed deltas and their correlations with observed behaviors. The reasoning agents will use inference techniques to infer the results, distributions and expected values for hidden nodes based on this observed behavior over time.

7. SUMMARY AND FUTURE WORK

This paper has chronicled our work in progress for the design, development and implementation of the CARET Agent-based Framework. The resulting capability will be used to provide an aide to Signal Operations Planners in analyzing and diagnosing issues with MANETs as operations are underway. The capability provides a variety of agents that can interact with multiple data sources including databases and real-time sensors, drive and calibrate multiple simulation environments, and apply various analysis techniques to guide signal planners in planning and reconfiguring networks on-the-fly.

The paper has provided an overview of a proof-of-concept that demonstrated how data sources, simulations, and evaluation tools could be brought together to provide a capability that could enhance signal planner cognitive output. The paper also presented descriptions of specific agents and their hierarchical structure used to form the basis for CARET prototyping efforts.

Future work involves developing additional series of cognitive models that can be hosted on AA-CR agents. These cognitive models will be used to provide inferencing techniques that can be combined and orchestrated to address problems such as the diagnosis of unknown jamming threats and the effects of equipment degradation over time. Additional work will incorporate re-planning capabilities with the diagnosis models to assist the signal planner in reconfiguring networks on the fly and address alternative "what-if" scenarios.

REFERENCES

1. Surdu, J.R. "Connecting Simulation to the Mission Operational Environment". Ph.D. Thesis. Texas A&M University. 2000.
2. Sisti, A. "Dynamic Situation Assessment and Prediction (DSAP)" Proceedings of SPIE, Enabling Technologies for Simulation Science VII Vol.5091. 2003.
3. McGraw, R. et al. "Real-time Updates to the DSAP Framework to Improve Calibration and Dynamic Situational Awareness." SPIE, Defense and Homeland Security Symposium, Modeling and Simulation for Military Operations II," April 2007.
4. Valinski, M, Mia, N, and McGraw R. "An Adversary Prediction Environment Based on DSAP Capabilities," SPIE, Defense and Homeland Security Symposium, Modeling and Simulation for Military Operations II, April 2007.
5. Reaper, J., Trevisani, D., and Sisti, A. "Real-Time Decision Support System (RTDSS)" Proceedings of the Western MultiConference. January, 2003.
6. McGraw, R., et al. 2004. "Software Framework in Support of Dynamic Situation Assessment and Predictive Capabilities for JSB-RD". In proceedings of the SPIE - Enabling Technologies for Simulation Science VIII Conference.
7. McGraw, R., et al. 2004. "Dynamic Situation Assessment and Predictive Capabilities in Support of Operations". In proceedings of the Fall Simulation Interoperability Workshop, Orlando, FL. 2004.
8. McGraw, R., et al. 2005. "A DSAP Infrastructure for the Global Information Grid's Modeling and Simulation Community of Interest". In proceedings of the Spring Simulation Interoperability Workshop, San Diego, CA. 2005.
9. Shao, G., et al. 2009. "Service Oriented Simulations for Enhancing Situation Awareness". In proceedings of 42nd Annual Simulation Symposium, San Diego, CA. 2009.
10. RAM Laboratories, Incorporated. "Management of Distributed Real-time Databases Final Technical Report". February 2007 in support of contract W9113M-04-C-0075.
11. Java Agent Development Environment (JADE), http://jade.tilad.com/
12. Mayk, I. and Regli, W.. Agent Systems Reference Model. Version 1.0a http://www.fipa.org/docs/ACIN-reference_model-v1a.pdf
13. SDO for Java Specification V2.1, http://www.osoa.org/download/attachments/36/Java-SDO-Spec-v2.1.0-FINAL.pdf?version=1
14. SOAP Version 1.2 Part 1: Messaging Framework (Second Edition), http://www.w3.org/TR/soap12-part1/
15. Murphy, K.. "A Brief Introduction to Graphical Models and Bayesian Networks". http://www.cs.ubc.ca/~murphyk?Bayes/bayes.html
16. Rabiner, L.R. "A Tutorial in Hidden Markov Models and Selected Applications in Speech Recognition", Proceedings of the IEEE, vol 77. pp. 257-286.
17. Mead, R; Paxton, J. And R. Sojda, "Applications of Bayesian Networks in Ecological Modeling". Montana State University.
18. Lund, T.; Faultner, E.; Robinson, M.; „Condition Monitoring Using Bayesian Networks". INFORMS Annual Meeting 2006.
19. Cassimatis, N. "Integrating Cognitive Models Based on Different Computational Methods".

BIOGRAPHY OF AUTHORS

Dr. Robert McGraw is a co-founder of RAM Laboratories where he is Vice-President and Chief Technology Office. Dr. McGraw's interests lie in the areas of multi-resolution modeling and distributed information management. He is currently overseeing RAM Laboratories efforts in developing a Dynamic Situation Assessment and Predictive (DSAP) infrastructure for the Air Force, and developing a CARET framework for the Army. Dr. McGraw received his Ph.D. in Electrical Engineering from the University of Virginia.

Dr. Gary Shao is Chief Scientist at RAM Laboratories where his interests lie in the areas of parallel and distributed computing, distributed information management, mobile ad hoc networks, and service-oriented software architectures. He is currently the chief software architect for various RAM Labs projects, including the COA Analysis with Radio Effects Toolbox (CARET) program. Dr. Shao received his Ph.D. in Computer Engineering from the University of California, San Diego.

Dr. Dean Mumme is a Senior Staff Scientist at RAM Laboratories. Dr. Mumme's interests lie in the areas of artificial intelligence, soft-computing, information assurance, and anti-tamper technologies. Dr. Mumme has been the Principal Investigator on RAM Laboratories' efforts to develop agent-based anti-tamper hardware and software and has also developed numerous soft-computing technologies for anomaly detection of sensor and network environments. Dr. Mumme received his Ph.D. in Computer Science from the University of Illinois.

Dr. Richard MacDonald is a co-founder of RAM Laboratories and currently serves as the company's President and CEO. Prior to starting RAM Labs, Dr. MacDonald worked at several companies developing simulation technologies for the semiconductors, avionics, and wireless industries and is active in Armed Forces Communications and Electronics Association (AFCEA) and the National Defense Industry Association (NDIA). In addition to his role as CEO, Dr. MacDonald also serves on the Board of Directors for the Society of Modeling and Simulation International. Dr. MacDonald earned both a Ph.D. and Masters of Science (M.S.) in Electrical Engineering from the University of Virginia, and a Bachelor of Science (B.S.) degree in Computer Engineering from the University of Michigan.