CIHSPS2004 - IEEE International Conference on
Computational Intelligence for Homeland Security and Personal Safety
Venice, Italy, 21-22 July 2004

Collaboration and Modeling Tools for Counter-Terrorism Analysis

Robert Popp[1], Krishna Pattipati[2], Peter Willett[2], Daniel Serfaty[3], Webb Stacy[3]
Kathleen Carley[4], Jeffrey Allanach[2], Haiying Tu[2], Satnam Singh[2]

[1] DARPA, 3701 North Fairfax Drive, Arlington, VA, 22203, USA
[2]University of Connecticut, Storrs, CT 06269, USA
[3]Aptima Inc., Woburn, MA 01801, USA
[4]Carnegie Mellon University, Pittsburgh, PA 1□213, USA

Abstract *– One of the major challenges in counter-terrorism analysis today involves connecting the relatively few and sparse terrorism-related dots embedded within massive amounts of data flowing into the government's intelligence and counter-terrorism agencies. Information technologies have the potential to empower analysts with a superior ability to process and analyze the data, disseminate and share it, and ultimately put the data into a form that allows senior decision-makers to understand and act on it so that they can anticipate and ultimately preempt terrorist plots or attacks from occurring. Advanced collaboration among multiple analysts or tools is one such crucial technology. In this paper, we introduce NEMESIS, a collaborative environment to integrate and share information among different modeling tools. Two component-modeling tools, ASAM System and ORA, are described in this paper. The functionality of these two tools along with the NEMESIS system is illustrated via a real world example gleaned from open sources.*

Keywords *– Counter-Terrorism, Collaboration, Hidden Markov Models, Bayesian Networks, Social Network Analysis.*

Table I: Acronym Glossary

Acronym	Meaning
NEMESIS	NEtwork Modeling Environment for Structural Intervention Strategies
ASAM	Adaptive Safety Analysis and Monitoring
ORA	Organizational Risk Analysis
ODL	Organizational Descriptive Language
BN	Bayesian Network
HMM	Hidden Markov Model
CAVS	Collaborative Artifact Version Server
IA	Indian Airlines

I. INTRODUCTION

A report filed by Congress assessing the events leading up to the 9/11 attacks suggests that there was sufficient amount of intelligence available for the "dots" to be connected — if it was not for the overwhelming amount of data available at that time, this could have been achieved. In order to prevent this from happening again, researchers within the intelligence community have begun working on a set of advanced information technology tools. These tools can empower intelligence agencies with the ability to find pertinent data faster, conduct more efficient and effective analysis, share information with others, relay concerns to the appropriate decision-makers, and support them with better information to make effective decisions.

From the perspective of an intelligence analyst, the majority of their time is spent on collecting data, when ideally it should be spent on analysis. A few of the key information technologies we consider crucial for counter-terrorism analysis include advanced collaboration, decision support, hypothesis generation, automated data management, and data processing. These technologies represent broad categories and are meant only to provide a relative framework for counter-terrorism analysis and encompass the many new advanced technologies under development. In order to collaborate efficiently and effectively, distributed teams within the intelligence community require a forum in which they can share ideas and solve complex problems, monitor their own effectiveness and dynamics as a team, systematically evaluate differing opinions, and generate alternative scenarios. Considering the massive amount of information and the difficulty in connecting the dots, a set of tools capable of automated evidence collection, evaluating alternative hypotheses, and supporting evidential reasoning would be invaluable [1]. In this paper, we focus on a set of collaborative and modeling tools for identifying, tracking, and mitigating terrorist activities. Three major components of the collaborative environment are described and illustrated using an IA hijacking example: (1) ASAM, which is based on a novel combination of HMMs to detect and provide soft evidence on the states of terrorist activity using partial and imperfect observations, and BN model integrates the soft evidence from multiple HMMs; (2) ORA, which combines ideas from social network analysis, organizational theory, and computational sociology to model the information flow and diffusion within terrorist networks, evolution of terrorist networks, and other related concepts; and (3) NEMESIS, which provides a forum for information exchange between ASAM and ORA, and model-based team collaboration.

The paper is organized as follows. Section II describes the NEMESIS collaborative environment for modeling and analysis. Section III summarizes the ASAM system that provides early warning to facilitate preemption and increase

the range of options for counter-terrorism agencies. Section IV provides a brief description of the ORA tool that detects risks and vulnerabilities in the terrorist organizations. In section V, the utility of the ASAM and ORA tools is illustrated by way of application to the IA hijacking example. Section VI concludes the paper with a summary and current research direction.

II. COLLABORATION VIA NEMESIS

NEMESIS is an IT-based collaborative environment for counter-terrorism analysis, which provides access to different modeling and analysis tools. With a set of collaborative analysis tools, intelligence and policy analysts can improve their capabilities to identify, understand, and mitigate terrorist activities from an organizational perspective using a well-designed XML-Schema-based language termed ODL.

Collaboration artifacts such as E-mail, chats, or forums, are rarely or never need to be revised. However, modern collaborations produce artifacts that are subject to revisions. Examples include collaboratively authored documents, scenarios and models of terrorist organizations, structured argumentation, strategies, and plans. The revisions come about because participants in the collaboration agree that new data, new analysis, or new discussion should be reflected in the official "best guess." Multiple hypotheses about the best representation of the actual state of the world, exploratory investigation, and changes to subordinate collaboration artifacts on which a super-ordinate artifact depends should be accommodated. It is important to keep the revisions of the collaboration artifacts synchronized, so that multiple analysts can work on an intelligence problem independently and concurrently. The collaboration artifacts thus need to be controlled to prevent work from being lost or delayed.

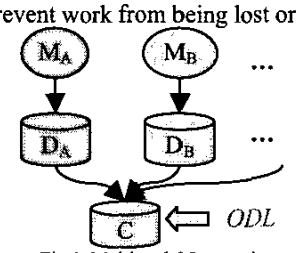

Fig.1 Multimodel Integration

One of the key purposes of NEMESIS is to integrate different modeling methods. Fig.1 illustrates that different modeling methods 'M_A, M_B, ...', may have their own data sets 'D_A , D_B , ...', and that each data set has a reference to an organizational description 'C' expressed in the ODL. ODL represents commonalities among modeling methods with a core set of constructs, and accommodates unique requirements of specific methods with ODL extensions.

As shown in Fig.2, NEMESIS has a service-oriented architecture, and ODL provides a focal point for integration. Multiple applications extract or produce ODL descriptions via adaptors, and models and associated data are stored via a CAVS. Collaboration tools include Groove Workspace™ from Groove Networks and clients for CAVS such as RapidSvn [6]. Network Visualization is a tool that graphically visualizes the organizational network described in ODL. The NEMESIS repository is hosted on the server side and saves the ODL files in an appropriate mainline or branch.

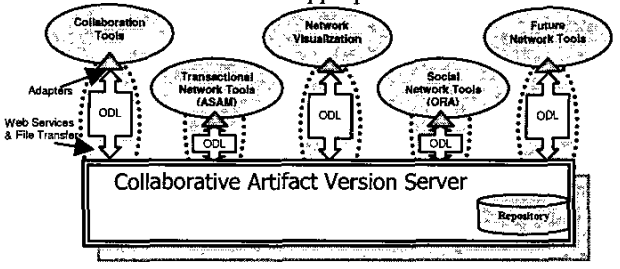

Fig.2 NEMESIS Architecture

ODL provides a platform to experiment with ways to represent organizations in NEMESIS. It builds on the foundations of DyNetML [9] and is designed for modularity following the general approach of XML. ODL consists of groups of node types: agents, knowledge, tasks, events, resources, locations, communications, and organizations. Each node type has attributes such as name, ID, delta, and binding. A delta element describes the difference between the same nodes in different time slices, and a binding expresses a placeholder to represent uncertain or incomplete facts. The major objective of ODL is to integrate organizational analysis tools by expressing core organizational facts. However, these organizational analysis tools are quite diverse and require significant amount of specialized information in addition to the core organizational facts. In the following two sections, we introduce two tools, viz., ASAM and ORA. Models in ASAM are represented in an extension to ODL named AsamML. HMMs in AsamML are described as organizational elements that are snapshots of an actual organization. That is, one HMM corresponds to one organization. BNs are captured in an ODL extension similar to a well-known representational format named XMLBIF. Models in ORA are represented in DynetML, which has a rich representation of organizations. NEMESIS provides XSLT-based translations between ODL and DynetML.

III. THE ASAM SYSTEM

The ASAM system, developed by the University of Connecticut, is an information analysis tool designed to support strategic decision-making, provide early warning to facilitate preemption, increase the range of options and probability of success, integrate information in a scalable way, and provide efficient and effective methods for analysis. The basic premise of the ASAM system is that terrorist networks can be evaluated using transaction-based models. This type of model does not rely solely on the content of the information gathered, but more on the significant links between data (people, places, things) that appear to be suspicious. For example, an unknown person withdraws money from his/her bank account, uses that money to

purchase chemicals that could be used to make a biological weapon, and then buys a plane ticket destined for the United States. This sequence of events suggests a reason to be concerned; it may or may not arise from terrorist activity, but ought to be flagged for more careful scrutiny. The ASAM system interprets the information by comparing a repository of *a priori* story schemas to actual observations of temporal data stored in an intelligence database. Based on the relationship between the observed temporal data and the given story schemas (templates), the likelihood of observed data given the templates is assessed. The following section summarizes the methods by which the ASAM system models and detects terrorist activities.

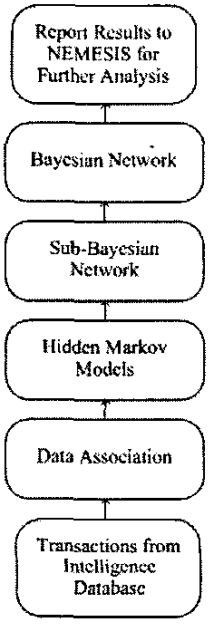

Fig. 3 ASAM Hierarchy

As shown in Fig. 3, the ASAM process is a hierarchal combination of HMMs and BNs. HMMs are used for modeling embedded stochastic processes and are therefore an ideal way to make inferences about the evolution of terrorist networks [3]. HMMs function at the lowest level of the ASAM process by taking observations of temporal data and comparing them to *a priori* story schemas via data association methods. As the HMMs track the evolving terrorist activities, they *continuously evaluate the likelihood* of the observed events and report these to a BN (or sub-BN). BNs combine the likelihood from many different HMMs (story schemas) to evaluate the overall probability of terrorist activity. In other words, the BN represents the overarching terrorist plot and the HMMs – which *are related to each BN node* – represent more detailed terrorist subplots. It is important to note that the HMMs function in a faster time-scale than BNs because the HMMs model the evolution of the transaction space, i.e., they process new information every time a transaction occurs. Each HMM can be viewed as a detailed stochastic time-evolution of a particular node represented in the BN. Data sent from the HMMs to the BNs, also known as soft *evidence, is the probability of observing the incoming data* given an *a priori* model of a terrorist network. Fig. 4 shows an example of a BN, where the gray nodes represent HMMs and the shaded nodes represent BN nodes with specified prior probabilities. A more detailed example highlighting the interrelationships between the HMM and BN models is provided in section V.

The input to the ASAM system is a series of transactions, taken from an intelligence database, that represent any kind of travel, task, trust, or communication between any person, place, or item of suspicious origin. As more transactions are detected, more links representing the transactions are made in the transaction space. The idea behind using HMMs is that we can represent its underlying states as snapshots of the growing transaction space, as shown in Fig. 4. Note that within each of the states of the HMM is a graphical representation of the terrorist network. One of the benefits of using HMMs is that there exist ways to detect the presence of an HMM among noisy benign data – this is analogous to finding a needle in a haystack and is one of the major problems associated with counter-terrorism analyses. By using advanced detection methods and standard HMM algorithms such as the forward, backward, Viterbi, and Baum-Welch [7], we can detect suspicious activities and develop models for counter-terrorism that are more accurate and effective than is possible with manual methods being practiced today.

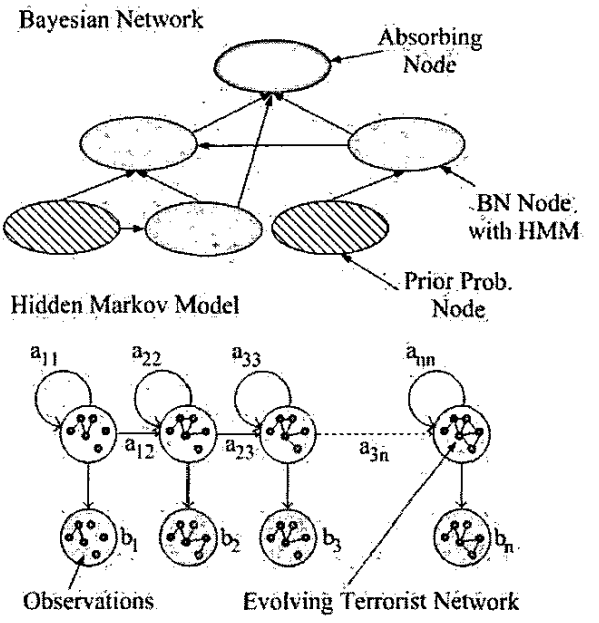

Fig. 4 The ASAM System Uses BNs and HMMs to Model Terrorist Activity

The architecture within the ASAM system consists of four modules: HMM, BN, web-based visualization, and knowledge repository. All of these modules have the ability to communicate via a local host or network so that different agencies can work together on developing models, sharing information, and exchanging opinions. The web-based visualization provides users with the ability to obtain real time information about the state of the terrorist threat and the ability to test new hypotheses (so called what-if scenarios). In order to facilitate collaboration within the intelligence community, the ASAM system has been designed to communicate with a central repository in NEMESIS via ODL. Whenever ASAM detects new activity or receives a new transaction, it updates the corresponding model and inference results for analysts to view from the NEMESIS repository. Once received, the NEMESIS system reports any

new transaction data to the analyst, and provides the capability to analyze the new data using other counter-terrorism analysis tools, such as the ORA.

IV. THE ORA TOOL

Carnegie Mellon University has developed a series of integrated tools for dynamically extracting terrorist network data, visualizing terrorist networks, identifying the "network elite" and points of vulnerability, and then evaluating the potential impact of various types of attacks on those networks. These tools include network-vis, a network visualization tool; ORA [8], a statistical toolkit for analyzing dynamic networks composed of multiple organizations; DyNetML [9], an xml based interchange language for relational data to handle roles, personality factors and action attributes; AutoMap [10], an automated text analysis tool that can extract relational data including social and role network data from text; and DyNet [11], a tool for simulating the evolution of these networks in general and after they have been attacked. Fig. □ shows the interoperability of these tools.

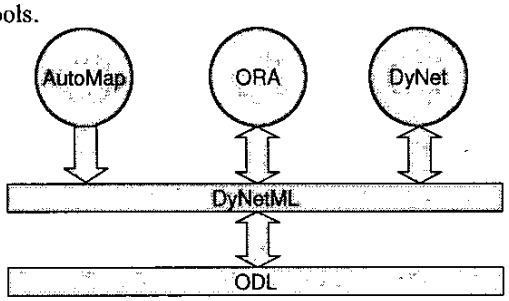

Fig.□ Suite of CMU Tools

As part of NEMESIS, we have focused mostly on ORA, with network-vis as the network visualization tool and DyNetML as a way to exchange information with the ODL in the NEMESIS environment.

ORA is a network tool that detects risks and vulnerabilities in an organization's design structure. The design structure of an organization is the relationship among its personnel, knowledge, resources, tasks, and entities. These entities and their relationships are represented by a collection of networks called the Meta-Matrix. Table II lists the available Meta-Matrix in ORA.

The main input to ORA is an organization. ORA can analyze an organization for weaknesses and vulnerabilities, either at an individual or organizational level. Such risks include, but are not limited to, tendency to groupthink, overlook information, communication barriers, and critical actors. ORA analyzes the Meta-Matrix using measures. An

Table II Meta-Matrix Showing Networks of Relations

Entities	Actor	Knowledge/ Resources	Events/Tasks	Organizations
Actor	**Social Network** *Who talks to, works with, and reports to whom*	**Knowledge/ Resource Network** *Who has what expertise, or has access to which resource*	**Attendance Network** *Who is assigned to which task, who does what*	**Membership Network** *Who belongs to which organization*
Knowledge/ Resources	—	**Information Network** *Connections among types of knowledge, resources, substitutions*	**Needs Network** *What type of knowledge□ resource is needed for that event□task*	**Core Capabilities** *Which organization has what kind of knowledge□ resources*
Events/ Tasks	—	—	**Precedence/ Dependencies** *Which events□ tasks are related to which*	**Sponsorship Network** *Which organization is sponsoring which task*
Organizations	—	—	—	**Inter-organizational Network** *Alliance formation*

ORA measure is a function that takes a Meta-Matrix and examines a particular aspect of its mathematical structure. ORA contains over □0 measures, and provides three classifications of them based on risk and vulnerability, input requirements, and type of output produced. For example, *Critical Actor Risk* is the risk based on the actors having exclusive knowledge, resources, or task assignments. ORA reads and writes network data in multiple formats to make it interoperable with existing network analysis software, such as DyNet [8]. In NEMESIS, the Meta-Matrix data is formatted as DyNetML. DyNetML supports multiple Meta-Metrics to be written in the same file, and each Meta-Matrix can have different Agent, Knowledge, Resource, Task, and Organization node sets. ORA generates the risk and vulnerability report from the measure analysis, both for a single organization and for comparing two Meta-Matrix organizations.

ORA advances the state of the art in network analysis tools by being organized around the unifying concept of the Meta-Matrix. Measures are organized to facilitate their coherent use. In particular, they are categorized by how they measure the risk and vulnerability of an organization's design structure. ORA reads and writes in multiple data formats and is interoperable with existing network analysis software. Entire Meta-Matrices can be visualized using different layout algorithms. The integrated Optimizer adapts an organization's design structure according to user specified criteria, and the resulting organization can be visualized and analyzed with ORA. ORA is being actively developed and tested in a wide range of contexts.

V. EXAMPLE: INDIAN AIRLINES (IA) HIJACKING

The IA hijacking example is extracted from open source information from the Embassy of India [4] and the Frontline magazine [□]. The example contains patterns of actions and responses that are present in the actual hijacking of IA's IC-814 flight, which occurred on December 24th, 1999 in

Kathmandu and ended on December 31st when the government of India released three high profile terrorists. The following sections describe the analyses of IA hijacking example via the ASAM and ORA modeling tools in NEMESIS.

A. The Organizational Model of the IA Hijacking Example

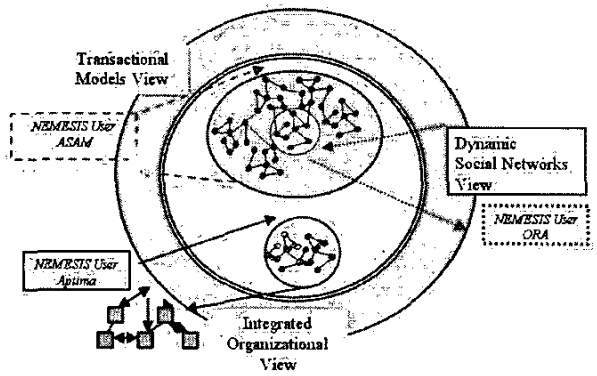

Fig.6 NEMESIS Collaborative Environment

The organizational model of IA hijacking is represented in such a way that the ASAM and ORA tools can work with the ODL data. Fig. 6 shows that multiple organizations analyze the same problem via diverse tools under the NEMESIS environment. The key feature of NEMESIS for collaboration is its capability to store, and manage different versions and configurations of models. ODL is structured to allow for modular extensions to accommodate current and future specific tool requirements.

The ODL model of IA hijacking consists of agents (people, avatars etc.), knowledge items, tasks (organized or planned activities), events, resources, locations, communications and organizations (ties between nodes and/or singletons). 'Agent' nodes consist of specific entities in the transactions such as fundamentalists, planners, hijackers, a weapons team, and local smugglers. The properties of agents are expressed as attributes: ID, name, etc. 'Resource' nodes are weapons, miscellaneous tools, forged documents, money, etc. "Location" nodes refer to target country, potential target, hijack flight, target airport, etc. The 'Organization' nodes are snapshots of an actual organization. For example, the HMMs in ASAM can be described as an "Organization" node.

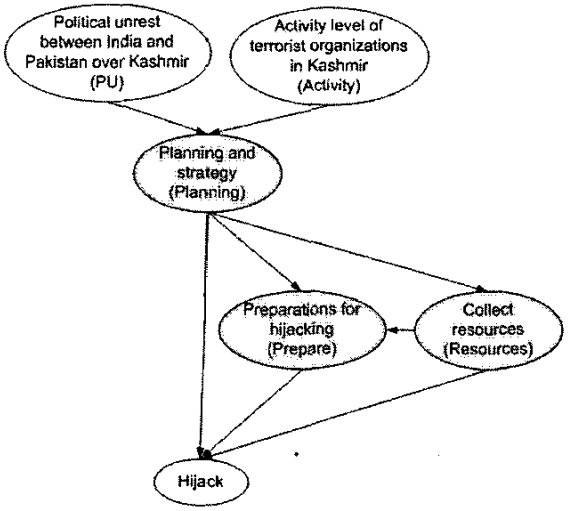

Fig.7 The BN Model for IA Hijacking

B. Analysis of the IA Hijacking Example via the ASAM System

The analysis of IA hijacking model is done by importing the model from the NEMESIS repository to a local repository in the ASAM system. The top level of the ASAM process is a BN, which represents the causal relationships among the events. Fig. 7 illustrates the BN model of the IA hijacking example. In the following simulation, the prior probabilities associated with the BN nodes are held constant, while the statistical inferences calculated by the underlying HMMs ('Planning and Strategy', 'Collect Resources' and 'Preparations for Hijacking') update the global probabilities of the BN. The global effect of these numerous terrorist activities causes the belief of the BN node, 'Hijack', to change. The state of 'Hijack' BN node is a probability mass function, which shows the posterior probability of hijacking. All the BN nodes are assumed to have discrete states.

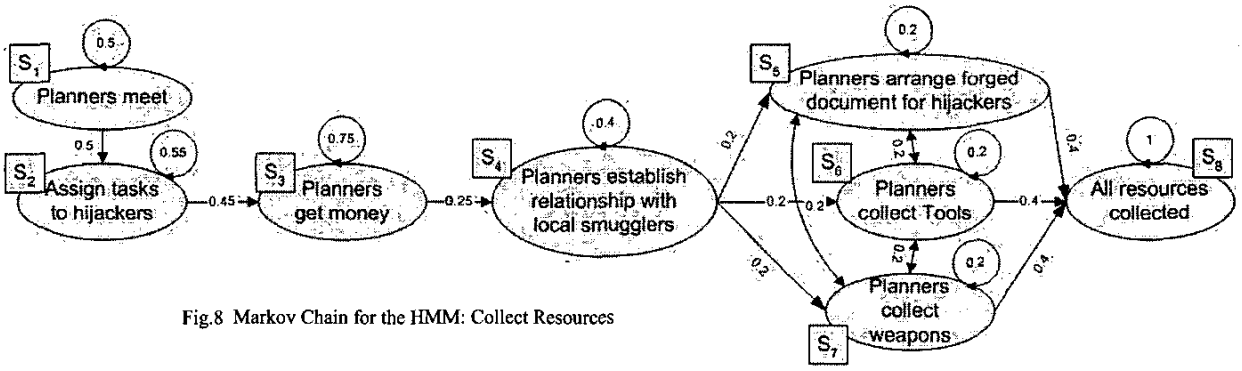

Fig.8 Markov Chain for the HMM: Collect Resources

In this model, three HMMs symbolize the planning and strategy, resource collection and the preparations for hijacking. Due to space limitations, only the Markov chain of 'Collect Resources' is shown here. Details of other HMMs are discussed in [2].

Fig. 8 shows the HMM corresponding to 'Collect resources', while it includes the transactions that are involved in collecting the necessary resources to carry out a hijacking. The 'Collect Resources' HMM has eight states which are indicated by S_1, S_2, ...S_9, and the transition probabilities are indicated next to transitions. Planners hold meetings with hijackers, and assign individual roles and identities for the hijacking. Planners obtain money through the high command of the terrorist organization, and they utilize the money to purchase forged passports, fake driving licenses, and satellite phones. Planners also acquire and transport the arms, ammunition through connections with local organized crime cells.

The BN merges all the available information from diverse sources and generates a global alarm, which is shown in Fig. 9. For simulation purposes, we speeded up the flow of the new transactions to every few seconds, with the actual dates associated with the IA hijacking events labeled in the figure. BN updates the belief only when the HMMs detect significant new evidence.

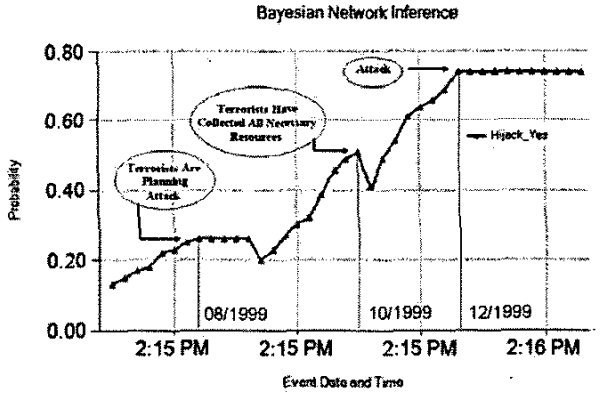

Fig. 9 The Posterior Probability of Hijacking

C. ORA Analyses of the IA Hijacking Example

Based on the ODL file for the IA Hijacking model, a customized extraction engine for ORA pulls out the relevant networks and puts the data into DyNetML. This data is then read into ORA and analyzed and visualized with network-vis. In this example, there are 20 actors, 13 resources and 13 tasks. There is a one-to-one mapping of resources and knowledge. Further, all the actors are associated with a single group.

Currently, ORA simply reports the results, as shown in Table III, as part of the Intel report. We are currently expanding this report to include confidence in the results, comparative evaluation with other networks, and key possible actions indicated by that data. For example, a typical network might have a density closer to .28, and an average degree centrality of .28 and betweenness of .06. Thus, the individuals who stand out here are much less connected than we would see in typical western networks. This may suggest a different mode of conducting operations; however, a more likely explanation is that there is substantial missing information – possibly as high as 87%.

Table III. Intel Report from ORA for Indian Hijacker Data

Measure	Value	Definition	Meaning
Cognitive Demand	.069 Akhtar	Actor with highest Cognitive Demand	Individual most likely to be an emergent leader, isolation of this person will be moderately crippling for a medium time
Total Degree Centrality	.079 Abdul Latif	Actor with highest Total Degree Centrality	Individual most likely to diffuse new information, isolation of this person will be slightly crippling for a short time
Betweenness Centrality	.012 Harkat-al-Ansar	Actor with highest Betweenness Centrality	Individual most likely to be able to find out information, also good at spreading information, isolation of this person will be slightly crippling for a short time
Task Exclusivity	2 Akhtar	Actor with highest task exclusivity	Critical individual, if the tasks are mission critical, isolation of this person is likely to be crippling
Overall Complexity	.03□	Density of the entire Meta-Matrix	Very low density. Probably major amounts of missing data, possibly cells are self directed.
Component Count	12	Number of undirected components in the entire Meta-Matrix	Possible indication that cells are self directed, possibly competitive factions exist.

The full meta-matrix is shown in Fig. 10. One of the keys in facilitating analysis will be to extend ORA to display visually changes in these networks over time and to highlight critical actors.

VI. SUMMARY

Information technologies are essential for the global war on terrorism [1]. This paper proposed a collaborative analysis environment, termed NEMESIS, that utilizes various information technologies to collaborate, evaluate, share, and act on the information faster to detect and prevent terrorist attacks. We described the versioning of collaboration artifacts in NEMESIS when multiple tools or analysts are concurrently working on the same problem. The two analysis

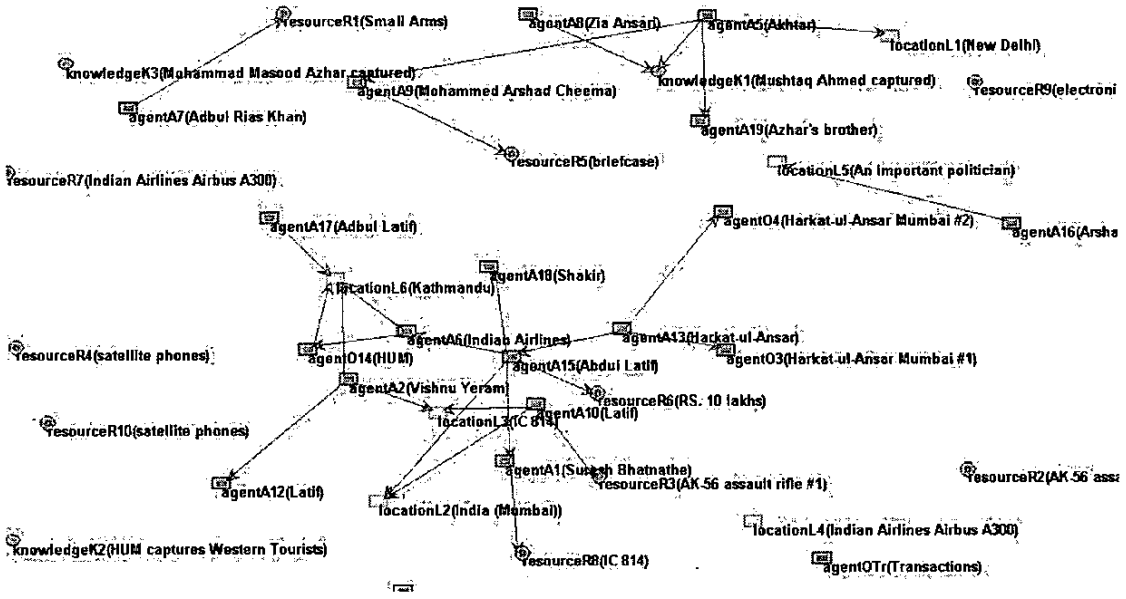

Fig. 10 The IA Hijacking Meta-Matrix

tools integrated within the NEMESIS environment, the ASAM system and the ORA tool, were then introduced. The ASAM system combines the HMM and BN methods to detect terrorist activities and generate global threats. The ORA tool, based on social network analysis, models the information flow within terrorist networks and the evolution of the terrorist networks over time. The feasibility and functionality of the NEMESIS collaboration was demonstrated using a real world example, the 1999 Indian Airlines hijacking problem, extracted from open sources.

The current implementation of NEMESIS provides collaboration among different tools by sharing the same data. In the future, ODL will be extended to describe transactions so that the adaptors associated with analysis tools are able to accept ODL formatted transactions as inputs. The NEMESIS environment has the potential to be integrated with additional organizational modeling tools. Meaningful collaboration and tool effectiveness measurement will also be developed.

There are some major extensions being pursued for both the ASAM system and ORA tool. For example, the ASAM system will incorporate feature-aided (attribute-aided) threat tracking to include people, places and infrastructure targets into the HMM models, and the influence of counter-terrorism actions to preempt terrorist attacks. Future work on ORA will address all aspects of its core functionality, including: extending the meta-matrix manager to allow multiple matrices of a single type; allowing the user to specify the input for measures; displaying matrix data in an editable spreadsheet window; generating reports with multiple types; and improving the user interface.

REFERENCES

[1] R. Popp, T. Armour, T. Senator and K. Numrych, "Countering terrorism through information technology" *Communications of the ACM*, March 2004.

[2] H.Tu, J. Allanach, S. Singh, K. Pattipati and P. Willet, "The adaptive safety analysis and monitoring system", *SPIE Defense and Security Symposium*, April 2004.

[3] J. Allanach, H. Tu, S. Singh, K. Pattipati and P. Willet, " Detecting, tracking and counteracting terrorist networks via hidden Markov models," *IEEE Aerospace Conference*, March 2004.

[4] Indian Embassy, "Information of Indian hijacked flight IC-814," http://www.indianembassy.org/archive/IC_814.html.

[5] Frontline magazine, "Kashmir after Kandahar," http://www.flonnet.com/fl1702/17020040.htm.

[6] RapidSvn, http://rapidsvn.tigris.org.

[7] L. Baum, T. Petric, G. Soules, and N. Weis, "A maximization technique occurring in the statistical analysis of probabilistic function of Markov chains," *Annals of Mathematical Statistics,* 41(1), pp.164 - 171, 1970.

[8] ORA, K. Carley and J. Reminga, http://www.casos.cs.cmu.edu/projects/ora .

[9] DyNetML, J. Reminga and K. Carley, http://www.casos.cs.cmu.edu/projects/dynetml/ Tsvetovat.

[10] AutoMap, J. Diesner and K. Carley, http://www.casos.cs.cmu.edu/projects/automap.

[11] DyNet, http://www.casos.cs.cmu.edu/projects/DyNet/dynet_info.html.