15th ICCRTS

"The Evolution of C2"

Mission Profiles and Evidential Reasoning for Estimating Information Relevancy in Multi-Agent Supervisory Control Applications

C2 Architectures and Technologies

Nathan Denny and Brett Walenz

Nathan Denny
21st Century Systems, Inc.
6825 Pine St. Omaha, NE 68106-2857
(402) 502-8439
nathan.denny@21csi.com

Brett Walenz
21st Century Systems, Inc.
6825 Pine St. Omaha, NE 68106-2857
(402) 505-7895
brett.walenz@21csi.com

1 Introduction

With the increasing use of semi-autonomous systems such as UGVs and UAVs, command and control concepts are proliferating further to the edge. Future network-centric environments such as the Global Information Grid (GIG) hold the potential to bring necessary information to those at the edge; however, providing the information is not enough. The flow of information now risks becoming a flood which can drown individuals as they allocate limited attentional and cognitive resources to filtering out extraneous information and assembling situation awareness from the remaining relevant pieces.

One solution to the problem of information overload is to introduce automation that can filter and process information according to the needs of the human user. There has been much interest in this approach in both the civil and military domains. Military approaches with significant resources and a critical need for deterministic computability tend to employ if-then rules constructed by subject matter experts. Civilian information filtering applications, e.g. Internet spam filtering, often use machine learning techniques such as rule-induction or learned Bayesian-networks. Most rule-based systems are based on binary Boolean logic and have difficulties in dealing with trust in the source of the information and the uncertainty inherent in the information. Systems that employ machine-learning techniques do not have explicit goal congruence and may suffer from over-fitting to a particular context as well as a deficiency of trust in the automation on the part of the human operator.

This paper introduces a multi-agent architecture that employs an evidential reasoning mechanism and mission profile structures to perform relevancy estimation for information filtering. The incorporation of the mission profile structure provides a means by which operators can easily develop their own profiles or re-use those that have already been developed. Mission profiles provide a means to explicitly connect perceived objects and inferred relations to arbitrary mission goals and subgoals. In this way, mission profiles are a vehicle by which the software and operator can achieve goal congruence. The resulting system is thus more of a software teammate rather than software tool.

The remainder of the full paper is organized in seven sections. Section 2 presents a review of relevant literature. Section 3 introduces the architecture of the proposed multi-agent system with preparation for more detailed discussion of key components in Sections 4 and 5. Section 4 introduces the concept of the mission profile. The mission profile is a typed semantic network that is assembled in three distinct layers: the goal layer, the conditions layer, and the trust layer. This section will describe each layer and its relation to the other layers and describe a tool by which operators can graphically construct mission profiles using a point and click interface. This section will further describe an extension to this tool which can allow for mission profiles to be shared and collaboratively constructed using wiki as a model of distributed authorship. Section 5 presents the evidential reasoning algorithm used for estimating the relevance of perceived objects and inferred relations. The algorithm is part of a larger belief fusion pipeline that propagates influence and relevancy. The included evidential reasoning mechanism uses subjective logic as a belief algebra and can perform both consensus and discounting of evidence and influence. Consensus allows for multiple pieces of evidence to support the same conclusion and for multiple inferences to "back propagate" their relevancy onto supporting evidence. Discounting provides the capability to describe both trust in production agents and the relevance of subgoals to goals. Section 6 demonstrates the concept by working through a simple supervisory control scenario where the operator must monitor a situation that includes multiple semi-autonomous vehicles. The example focuses on the basic kinematics that are common to many military applications with accompanying figures that illustrate the reasoning process and flow of evidence, influence, and relevancy. Section 7 recapitulates the key points of this paper and presents significant conclusions as well as outlining avenues for further, related research and development.

2 Review

2.1 A Review of Situation Awareness

Situation awareness (SA), or situational awareness as it is also sometimes called, is the internal mental representation and understanding of objects, events, people, system states, interactions, environmental conditions, and other situation-specific factors affecting human performance in complex, dynamic, and often potentially lethal tasks. Intuitively, it is one's ability to give answers to such questions as: What is happening? Why is it happening? What will happen next? What can I do about it?

Endsley's model [E88] is not the only model of SA, however, it is the most widely accepted and generally applicable model. Endsley's model defines three levels of SA. As any entity or event which is processed at higher levels must first be perceived by the operator, Level 1 SA is the perception of elements in the current situation. The Level 2 SA is the establishment of awareness of relationships between the objects and events perceived in Level 1 SA. Level 3 SA is the operator's ability to forecast the situation based upon information gathered and produced in Level 1 SA and Level 2 SA.

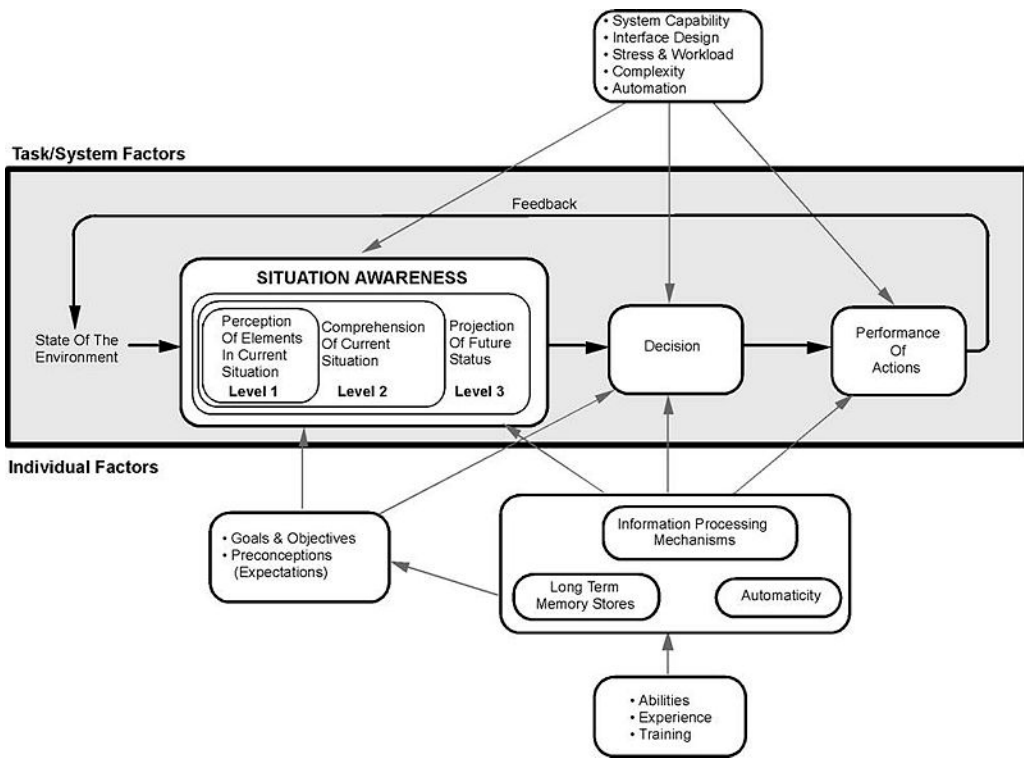

Figure 1. Endsley's SA model [E95]

Endsley's model (depicted in Figure 1) credits SA as the critical input into the operator but SA is decoupled from the actual decision. For our purposes, Endsley’s SA model can be used as the basis of a rough metric for determining information relevancy: informational elements that can support higher levels of SA are deemed more relevant than those that do not. For instance, almost any object can be perceived in Level 1 SA, but those objects that are part of relationships (Level 2 SA) that are impactful with respect to established goals (Level 3 SA) are highly relevant when compared to other objects for which no such impactful relationships exist.

2.2 Multi-Agent Systems

An agent is some entity that can perceive, reason, and act [F98]. These features differentiate an agent from, for instance, a document or database which contains information, but has no understanding of the information nor can it act on this information. As defined, the term agent can be applied to both humans and software entities, however, in the context of this paper, the term agent will be reserved for reference to software entities only. (We will use the term operator or user when referring to human actors.)

Software agents are small, situated, and social software components that can perceive, reason, and act. Agents are most commonly developed to be focused on only one aspect of a much larger problem. Thus, agents are small in that they typically need to contain only a few rules or subject-matter expert (SME) heuristics for reasoning. Software agents are situated and social in that they are intended to participate in larger agent communities, receiving information not only from the problem domain, but also from other cooperating agents. Like human social networks, cooperating agents can achieve synergistic effects.

A multi-agent system (MAS) is a system that assembles agents into a cooperating collection and derives its computing power and utility primarily through the interaction of the agents in the collection [F98]. Generally, MAS systems are highly flexible. Although there is a necessary minimal ontological commitment and shared communication protocol, an MAS agent is otherwise not committed to any particular computing language, framework, or reasoning method. MASs are also highly distributable. All agents can run on a single multi-tasking machine, or the agents can be distributed across several machines for improved performance by the utilization of additional hardware processing power.

2.3 Subjective Logic

Subjective Logic [J97; J09] is a type of probabilistic logic that is often used in evidential reasoning (e.g. [LP07; LZ07]) where belief, disbelief, and uncertainty must be explicitly and simultaneously accounted.

In contrast to systems described by Boolean Logic, for those systems described by Subjective Logic the basic object is an opinion rather than a fact. An *opinion* $\omega^A(x)$ about some proposition "x" held by source "A" is a 4-tuple of the belief (b_x^A), disbelief (d_x^A), uncertainty (u_x^A), and relative atomicity (a_x^A). (Atomicity is the base-rate of the proposition.) Note that $b_x + d_x + u_x = 1$, so while it is not necessary to specify all three of the values, it is convenient when performing certain calculations.

$$K = u_x^A + u_x^B - u_x^A u_x^B$$

$$b_x^{A,B} = \frac{b_x^A u_x^B + b_x^B u_x^A}{K}$$

$$d_x^{A,B} = \frac{d_x^A u_x^B + d_x^B u_x^A}{K}$$

$$u_x^{A,B} = \frac{u_x^A u_x^B}{K}$$

$$a_x^{A,B} = \frac{a_x^A u_x^B + a_x^B u_x^A - (a_x^A + a_x^B) u_x^A u_x^B}{K - u_x^A u_x^B}$$

Figure 2. Subjective Logic Consensus Operation

The Subjective Logic algebra provides an array of operations that manipulate opinions. These operators have many applications in evidential reasoning and data fusion. For the present purpose, only the consensus and discount operators are of interest.

$$b_x^{A,B} = b_B^A b_x^B$$

$$d_x^{A,B} = b_B^A d_x^B$$

$$u_x^{A,B} = d_B^A + u_B^A + b_B^A u_x^B$$

$$a_x^{A,B} = a_x^B$$

Figure 3. Subjective Logic Discount Operation

The consensus operator (written as $\oplus$) is used for belief fusion, providing the capability to fuse possibly conflicting opinions while still forming coherent, summary judgments. The underlying calculations on the belief *tuple* elements are given in Figure 2.

Subjective logic also provides a well developed "discount" operation (written as $\otimes$) that can be used for modifying the contribution of evidence based upon a subjective measure of confidence in the source of the evidence. The discount operator thus provides a rather general means of describing degrees of influence and can be used to represent semantic similarity, relevance, trust, etc. The calculations for implementing a discount operator over belief *tuples* is shown in Figure 3.

3 Architecture

The general architecture of our proposed system is shown in Figure 4.

3.1 Information Ports

Information flows into the system through a generalized input port (101). In our internal research and development, our reference implementation assumes a JSON encoding of JC3IEDM statements that are provided via simulated GIG (102) services; however, the architecture itself is neutral in this respect and different implementations can vary accordingly.

The product of the system is a stream of statements that are emitted through a generalized output port (103). Again, the encoding or particular language of these statements is not specified by the architecture. The only requirement for implementation is that the emitted statements be able to describe the filtered situation and corresponding beliefs about the situation. The output port can then drive a variety of applications which work directly with a human operator or act as an input source for additional automation.

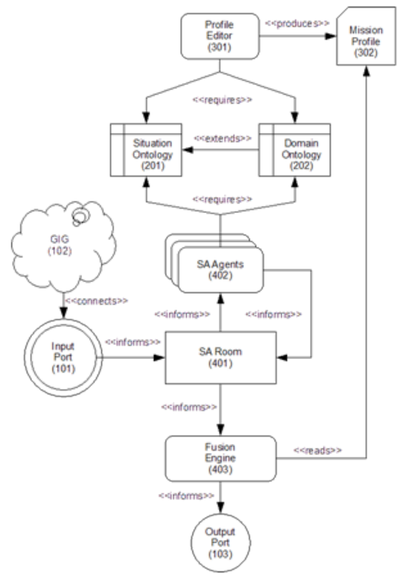

Figure 4. Architecture

3.2 Ontology

The internal mechanisms of the system and the emitted output (102) assume the existence of a well-defined ontology. In the architecture as shown, the system ontology is explicitly decomposed into an upper ontology (201) and a lower ontology (202).

The upper ontology, which we refer to as the Situation Ontology (SO), specifies the core ontological components that are needed to describe entities, relations, statements, agents, situations, beliefs, evidence, entailments, and so on. The lower ontology is the Domain-specific Ontology (DO) and extends the SO with specialized elements for describing the domain of interest.

General ontologies, such as SUMO [NP01; PNL02] and DOLCE [G02], might serve as starting points for developing an SO. More recently, there has been some interest in developing standardized situation ontologies. Of particular interest are SAWA [B02; MKB03; M05], which is lightweight and relatively easy to implement, and the Situation Ontology described in [KMB09] which is more formal and naturally expressed in predicate logic. In developing the reference implementation, we developed our own SO and DO. Our SO, which we call COSAUS (Composite Ontology for Situation Awareness in Unmanned Systems) is loosely derived from SAWA. The knowledge representation elements of COSAUS (COSAUS-KR) will be used later in this text and is shown in Figure 5 for reference.

Where the SO is essentially universal and re-usable across almost all applications of the proposed system, the DO must be tailored to the particular use or class of use for the system. Our reference implementation is employed as a situation awareness aid for supervisory control of unmanned systems in a tactical context. Hence our DO is able to describe situations in terms of kinematics, assets, etc. To highlight the specific nature of the DO, our kinematic DO can be reused in similar tactical applications, but would, for example, be a poor fit for logistics applications.

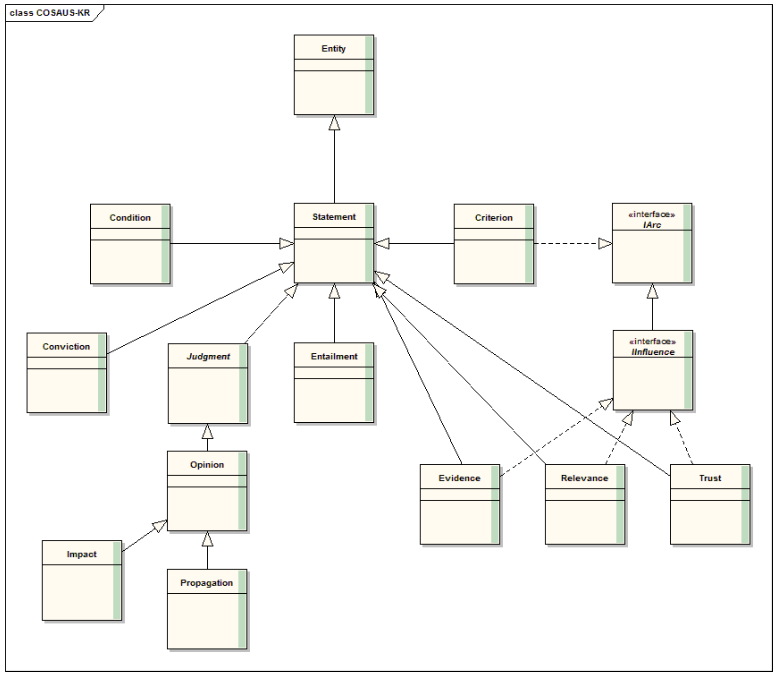

Figure 2. COSAUS knowledge representation elements

3.3 Mission Profiles

The Mission Profile Editor (301) is a lightweight graphical-user interface tool that is used offline to construct a Mission Profile (302). The mission profile is used to describe high-level mission goals and relevant situation elements that have the potential to impact the mission goals. Relevant situation elements are described in terms of the domain-specific ontology. Mission profiles and the editing tool are discussed in more detail in the next section.

3.4 Reasoning Components

The proposed system is a kind of blackboard architecture [C88], where agents communicate by posting to a centrally visible space (the blackboard). The event of a new post to the blackboard can trigger none, one, or more agents to opportunistically reason upon the state of the blackboard with the possibility of posting the consequences of the reasoning back to the blackboard for further work by other agents.

In our reference implementation, communication is via Extensible Messaging and Presence Protocol (XMPP) and the visible spaces are implemented as rooms hosted on the XMPP server. Hence, in the reference implementation we refer to the blackboard as the Situation Awareness (SA) Room (401). The use of XMPP is merely convenient for our prototyping and is not necessarily recommended for other implementations.

Much of the reasoning of the proposed system is distributed to a pool of agents (402). These agents reason about the present situation and produce entailments that fuse lower level data and information into higher level (Level 2 SA and Level 3 SA) relationships. In conjunction with the produced entailments, each agent should also supply its conviction in the entailment. The conviction, together with the entailment product, forms an opinion as defined by the calculus of subjective logic. Products of the agent must conform to the chosen SO and DO ontologies.

The agents act independently and are the “subconscious” reasoning of the proposed system. The products of the agents and the raw input need to be refined into a unified structure which can be emitted as the output of the whole system. This unification is performed by the fusion engine (403) which computes and assigns the overall influence of situation elements, both raw and those inferred as products of the agents.

4 Goal Congruence and Mission Profiles

Operator intent and mission goals are captured in a structure called a mission profile. Providing the information filtering system a model of the operator's goals allows for the system to assume those same goals. This is an early form of explicit goal congruence between humans and software agents.

A mission profile is a digraph with three levels. Level 1 is the mission model, which contains the goals, subgoals, and degrees of influence between subgoals and goals. Level 2 builds on top of Level 1 by associating degrees of influence between situation elements and subgoals. Level 3 builds on top of Level 2 by associating levels of trust to the ARID agents that make judgments on situation elements.

4.1 Profile Levels

4.1.1 Level 1 – Mission Goals

The most abstract level of the mission profile is the goal level. In this level the operator can recursively and arbitrarily decompose goals. Starting from the overarching mission goal, the operator declares subgoals that support the overarching goal, then further decomposes that goal to subordinate goals, increasing the level of detail. Each subgoal is related to its super goal by an arc which can carry a relevance score defined over the range (0.0, 1.0]. Higher weighted arcs indicate that the subgoal is highly relevant to the satisfaction of the super goal. When creating goals, more detail (deeper recursion) is generally preferred as the deeper decomposition will provide finer granularity and more opportunity for associating agents at intuitive points in the goal network.

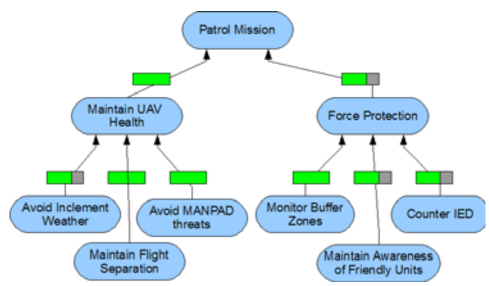

Figure 6. Level 1 - Mission goals

Figure 6 illustrates an example of a simplified Level 1 mission profile that has been constructed for a UAV patrol mission. Goal and subgoal nodes are blue ovals and contain arbitrary, descriptive text. Arcs between the nodes point from subgoals to supergoals and are labeled using the green relevancy boxes. In the simplified example shown in Figure 6, we see that the operator has created the abstract concept of a “Patrol Mission” and decomposed the “Patrol Mission” into two high-level goals: “Maintain UAV Health” and “Force Protection.” We also see that “Maintain UAV Health” is further decomposed into “Avoid Inclement Weather”, “Avoid MANPAD Threats”, and “Maintain Flight Separation.” Both “Avoid MANPAD Threat” and “Maintain Flight Separation” are considered to be maximally relevant to “Maintain UAV Health” while “Avoid Inclement Weather” is only partially relevant.

4.1.2 Level 2 – Conditions, Relations, and Relevance

The next profile level assigns situation elements to the goals described in Level 1. Such assignments are depicted graphically as an arc from an angled box containing an ontological element to one of the oval goal nodes, as seen in Figure 7. These associations communicate the impact of situation element on one or more aspects of the operator's overall mission goals. That is, if the associated element is inferred by one or more agents, then the relationship and any evidence supporting its existence should be considered for fusion.

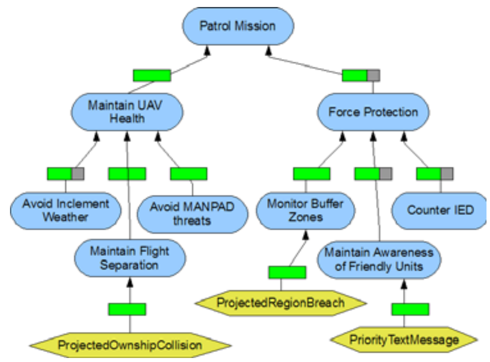

Figure 7. Level 2 - Conditions

Similar to the arcs between subgoals and supergoals, the impact of a relation on a goal node can be discounted to represent degrees of impact or relevancy. In Figure 7, we see that the operator has declared the existence of a ProjectedOwnshipCollision (an element from our reference domain ontology) to be maximally relevant to the mission's subgoal, "Maintain Flight Separation." Similarly, we see that the operator has declared that a ProjectedRegionBreach (another element from our reference domain ontology) is maximally impacting on the goal of "Monitor Buffer Zones."

4.1.3 Level 3 – Trust and Relevance in Agents

We see a complete profile in Figure 8, at right. Here, we have added three agents into the profile: Air Collision Monitor, Blue Force Monitor, and the RTC Monitor. Each of these agents produces one or more judgments on the existence and relevance of situation elements, as depicted by an arc from an agent to a conditional element. As in Levels 1 and 2, arcs carry a discounting weight. In this case, the discount weight is a measure of trust or confidence in the ability of the agent to make good judgments about the situation element.

Figure 8. Level 3 - Agent trust

The origin of trust between operator and agents is much the same as it is between members of any team; trust is learned through practice. Initially, all agents will likely be maximally trusted. Through practice in simulation, an operator can become more comfortable with the abilities of each agent as well as get a feel for which agents are best at making which judgments. As each agent embodies heuristics that mimic a small part of human reasoning, we do not expect *a priori* trust in an agent to be optimal. Rather, we expect a learning process that will modify trust over time. In human-human team relations, many of the factors that lead to trust are highly specific to the individuals in the team. While this is true also to some extent with artificial agents, their limited scope and singular purpose lead us to believe that human-machine trust will be very portable. That is, if one operator establishes trust on an agent after practice with the agent, other operators will likely be able to reuse the assigned trust metric without the extensive practice that was used to achieve the initial trust estimate.

In the example, we see that the operator has high confidence in the judgment of the Air Collision Monitor for inferring or producing ProjectedOwnshipCollision (another domain ontology element) relations.

The operator also has high confidence in the RTC Monitor and its ability to make effective judgments about PriorityTextMessage high level alerts. Note that it is possible that two or more agents can make judgments about the same kind of relations and events. The mission profile can accept multiple links onto a conditional node, allowing for multi-agent assemblies that can produce a wide diversity of opinions.

4.2 Propagation Through the Mission Profile

The proposed system estimates the salience of entities and inferred relations by computing belief propagation (from a matched condition node up to the root node of the mission profile). Figure 9 shows another mission profile that has been scaled down for better understanding. The yellow angled block represents a condition node that matches against instances of ProjectedBreach. Given a matching frame, Breach, of type ProjectedBreach, the associated belief for an instance of Breach would propagate up from the condition node along two paths. The left path traverses "Monitor Hostile Intent," "Force Protection," and terminates at the root node "Patrol Mission" after being slightly discounted in the last link. The right path traverses "Monitor Buffers" and is slightly discounted in propagating to "Force Protection." As with the left path, when the right path traverses the "Force Protection" to "Patrol Mission" link, it is discounted before terminating at the "Patrol Mission" root node. Note, in the example, that a single frame propagated its influence along two different paths to the root node. There is no limitation in the number of paths along which influence can propagate. In the later process of fusion, multiple propagation paths will strengthen the accumulated impact of a matched situation element.

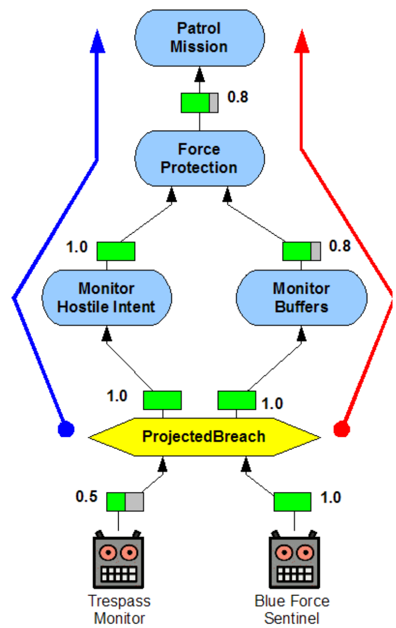

Figure 9. Multi-path profile

Internally, the mission profile is held as a network structure in a memex (memory context - a typed semantic network). The example profile in Figure 9 can be seen in an "exploded view" as a network of SO and DO instance frames in Figure 10. When computing a belief propagation, our reference implementation uses the mission profile network structure as the core of an instance of our Evidential Reasoning Network (ERN®). We then use subjective logic operations to propagate the belief, producing associated Propagation (an element of our COSAUS-KR SO) frames for each path through the profile.

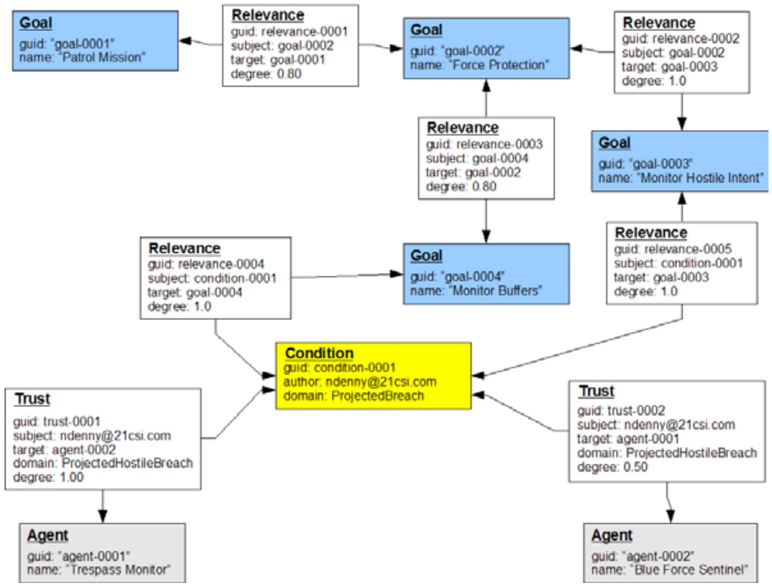

Figure 10. Mission profile "exploded view" in memex

For example, given the profile in Figure 9 and the simplified situation shown in Figure11, the ProjectedBreach frame with the guid "situation-0101" would match against the condition node with the guid "condition-0001." Using subjective logic operations to compute the propagation of the belief of the "situation-0101" (as held in a memex), the result would be similar to that shown in Figure 12.

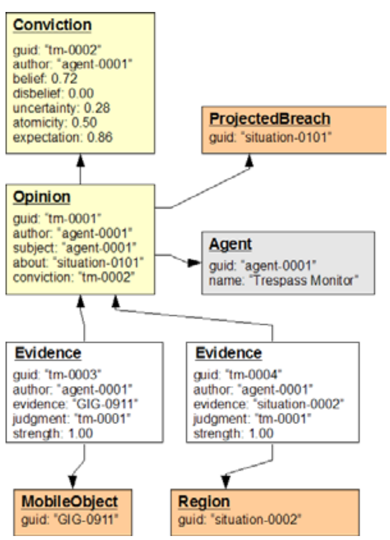

Figure 11. Simplified situation

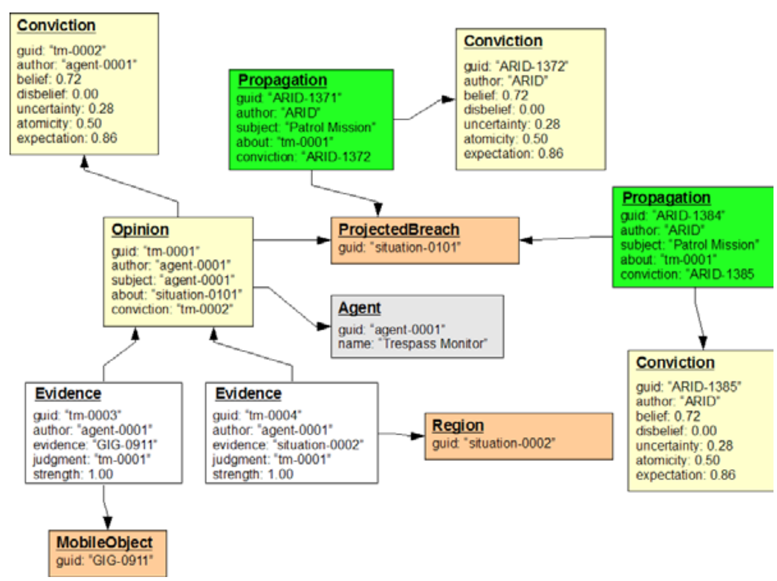

Figure 12. Memex after executing propagation

4.3 Mission Profile Editor

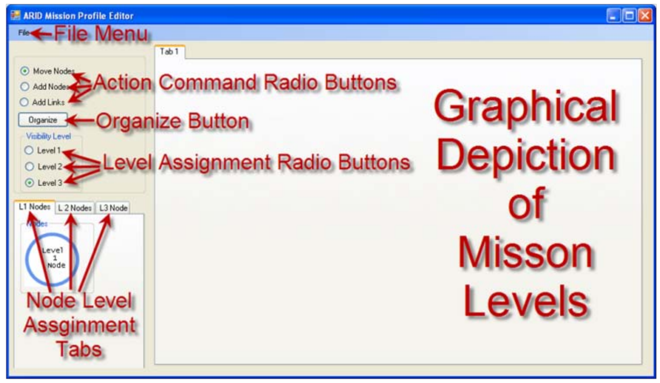

Figure 13. Annotated mission profile editor

Our reference implementation included a prototype GUI tool for authoring mission profiles. An annotated screen shot is shown at right in Figure 13. Using this tool, the mission profile can be built using the same layered approach as discussed above.

In the context of the editing tool, goal and subgoal nodes are represented as blue bordered circles, while situation element conditions are shown as blue bordered rectangles. Agents are represented as green bordered ovals. Influence arcs between nodes are equipped with a GUI widget which allows the author to drag the mouse to set the influence on the arc. Along with the visual representation of the degree of influence, a limited set of textual descriptions is also shown. Figure 14 shows a tear-out of an example profile that is in the process of being authored.

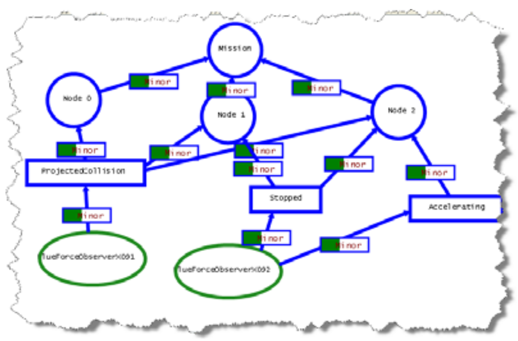

Figure 14. Profile in process of editing

5 Belief Fusion Pipeline and the Fusion Engine

In an open architecture, multi-agent system such as that of the proposed system, we expect situations where two or more agents are making inferences about the same relationship. Since each agent encapsulates a relatively small amount of knowledge, there are bound to be differences of opinion among them as each agent approaches the situation from its own perspective. The emitted output of the proposed system must be a set of coherent judgments. Thus, after the agents have reasoned over the incoming information, the next problem is to fuse the various, possibly conflicting opinions and judgments into a coherent whole.

In Figure 15, we show a high-level overview of how agent inference propagates through the proposed system. The fusion engine listens in the SA Room where the products of the agent pool are exchanged. These products include opinions, based on heuristics operating on low level information, and judgments derived from the opinions and judgments of other agents. Each agent operates asynchronously, reasoning opportunistically from the flow of information from the GIG as well as the products of other agents. The fusion engine passively collects all of these inference products as well as any raw data, accumulating them into a local memex.

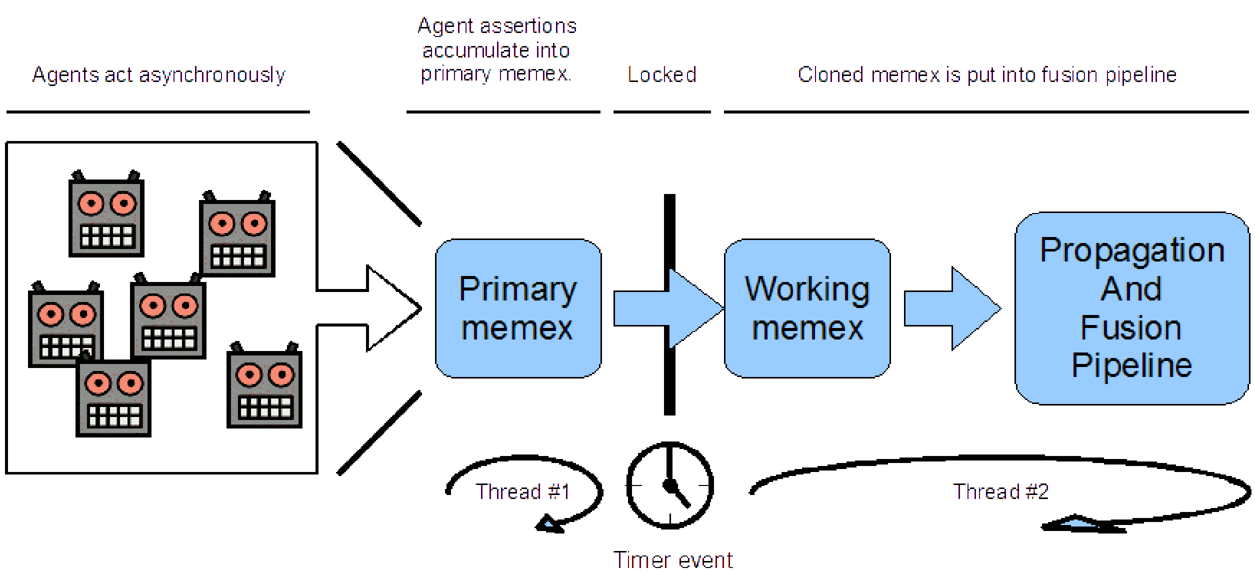

Figure 15. Overview of flow of inference products

In our reference implementation, at regular time intervals, a time event locks the primary memex and clones the primary memex into a working copy. Once the clone operation is complete, the lock is released and the primary memex is again free to listen to the agents and gather updates to its state. The working memex then enters the propagation and fusion pipeline in a thread independent of the primary memex. Rather than using a discrete time, alternate implementations may drive the fusion process by transaction events.

5.1 The Fusion Pipeline

The propagation and fusion pipeline is shown in Figure 16. As a prerequisite step, the working memex is first joined with the mission profile, as discussed in a previous section. The result of this prerequisite step is the addition of Propagation (an element of our COSAUS-KR SO) frames to the working memex. These frames describe the propagated influence of inference products on the overall mission.

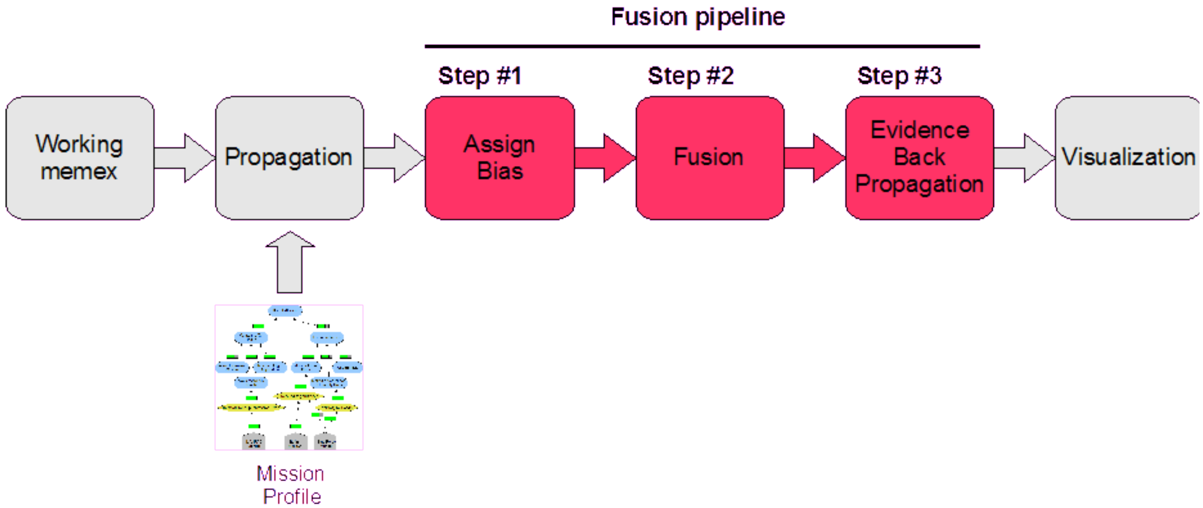

Figure 16. Propagation and fusion pipeline

5.1.1 Step 1: Assign Bias

With the Propagation frames already embedded in the memex, the first step of the fusion pipeline is to assign a bias to all the SituationElements (a core element of our COSAUS SO) that might be of interest. The default bias is the vacuous opinion, which signals that the situation element is acknowledged but the system has no reason, as of yet, that the element is of importance to the operator. By definition, in subjective logic, the conviction of the vacuous opinion is defined as the *tuple* (*belief* = 0.0, *disbelief* = 0.0, *uncertainty* = 1.0, *atomicity* = 0.5), which also fits our intuition into the nature of a lack of evidence.

While the default bias is vacuous, an implementation of the proposed system can adjust the bias by reallocating small increments of uncertainty to belief. Similarly, the bias can be decreased by first decreasing belief back to vacuous, then increasing disbelief by reallocating uncertainty. From the perspective of Step 1 of the fusion algorithm, bias is a parameter and the algorithm does not change under different variations of bias.

The Step 1 algorithm, described in pseudo-code, is given in the Algorithm 1 frame, below. Note that this step introduces the Impact frames into the memex. The Impact frames are COSUAS-KR elements and special cases of the more general Opinion class. Impact frames will act as accumulators of influence as the memex moves through the pipeline.

```
for each SituationElement, e:
        instantiate Impact statement, i, that refers to e.
        instantiate Conviction statement, c_i, set to (the default) ignorance (b=0,d=0,u=1.0)
        set i to refer to c
```

Algorithm 1: Assign bias to situation elements

5.1.2 Step 2: Fuse Propagation Paths

The prerequisite step of propagating influence through the mission profile results in embedding Propagation (from our COSAUS-KR SO) frames in the memex. As discussed in the previous section, for each situation element, there may be multiple propagation paths through a mission profile. Consider, for example, a mission profile with two leaf-node subgoals "Monitor Hostile Intent" and "Monitor Buffers" which are both sensitive to ProjectedBreach frames. In the event of the production of a ProjectedBreach frame, there will then be at least two paths from the Condition node up through the profile to the root of the tree, as depicted in Figure 17. The mission profile propagation algorithm will associate a Propagation frame to the ProjectedBreach frame for each of the paths, resulting in one situation element (the ProjectedBreach frame) with two Propagation frames. In this step, we use the subjective logic consensus operator to fuse Propagation nodes into their relevant Impact accumulator nodes. The pseudocode for this step is given in the Algorithm 2 frame.

for each Judgment j (where the author of j is not "ARID"):
  if $j.about$, is of type SituationElement; then...
    get SituationElement e to which $j.about$ refers
    for each Propagation p that refers to e:
      get the Impact, i, associated to e
      let c_p be the Conviction of the p
      let c_i be the Conviction of the i.
      accumulate c_p into c_i by consensus: $c_i \leftarrow c_i \oplus c_p$

Algorithm 2: Fuse propagation paths into Impact accumulator

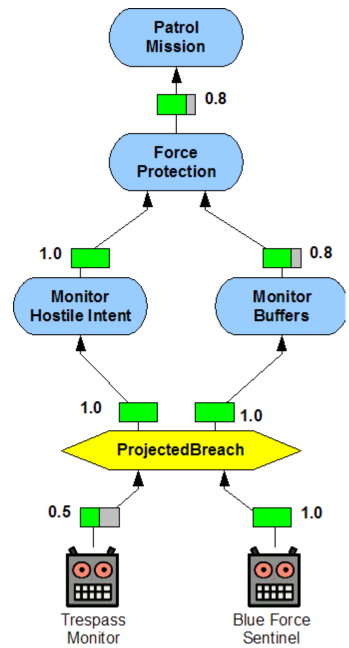

Figure 17. Mission profile

5.1.3 Step 3: Evidence Back Propagation

When an agent asserts some product of its reasoning, it associates to that product a Judgment frame and any associated Evidence frames that cite other SituationElement frames that the agent believes are relevant to its judgment. Should the mission profile be sensitive to the production, the impact of the production has already been computed in Step #2. The associated Evidence frames, however, imply that the cited SituationElement frames are indirectly influencing the mission via support for the reasoning product. The last step in the fusion algorithm is to “back propagate” the impact of reasoning products to the evidence that supports those products. The pseudocode for this step is given in the Algorithm 3 frame.

for each Evidence statement, *s*:
    let *j* be the Judgment that *s* supports
    let c_j be the Conviction of *j*
    let SituationElement *e* be evidence of *s*
    let *i* be the Impact of *e*
    let c_i be the Conviction of *i*.
    let *d* be the strength of *s*
    accumulate c_j into c_i: $c_i \leftarrow c_i \oplus (d \otimes c_j)$

Algorithm 3: Back propagate influence to supporting evidence

6 Example

In order to better illustrate the evolution of the memex as it proceeds through the fusion pipeline, in this section, we give a brief example including a focused view of the semantic network within the memex. In Figure 18, we see a simplified representation of a subgraph of the semantic network in a memex. Here, we have some ProjectedBreach with an associated Opinion (a subclass of Judgment) that is supported by two elements of evidence, a MobileObject, which would be the subject of the ProjectedBreach, and a Region, which would be the object of the ProjectedBreach.

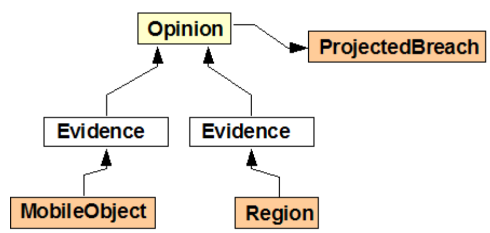

Figure 18. Initial memex state

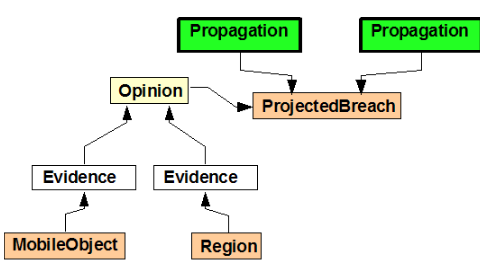

Figure 19. After computing propagations

Given the initial memex as a thread-safe working copy, we fit the memex to the mission profile and apply the propagation algorithm that computes the influence of each SituationElement on the mission. Using the mission profile shown in Figure 17, the ProjectedBreach has two propagation paths, resulting in two Propagation frames now linked and embedded into the working memex. Note that the mission profile was not sensitive (via a matching Condition node) to the MobileObject and Region frames. The state of the memex after the propagations is shown in Figure 19.

After the prerequisite step of computing the propagation paths and linking the Propagation nodes into the memex, we then execute Step 1 of the fusion algorithm and associate a default Impact bias to each SituationElement. The Opinion, Evidence, and Propagation frames are all part of our COSAUS-KR SO knowledge representation elements and are themselves not instances of the class SituationElement. ProjectedBreach, MobileObject, and Region are all members of our reference DO and inherit from the SituationElement class and therefore are given an associated Impact frame, as shown in Figure 20.

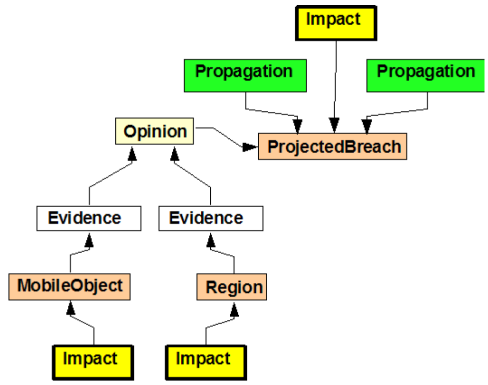

Figure 20. After fusion step #1

After the algorithm has created the necessary Impact frames and linked them into the memex, the next step is to accumulate the Propagation paths into the Impact frames. In this case, our figures are slight oversimplifications. The Opinion, Propagation, and Impact frames are all subclasses of Judgment and rather than store the subjective logic belief mass directly, they each have a one-to-one linkage with a corresponding Conviction frame. These Conviction frames are assumed in the figures and are omitted for purposes of clarity.

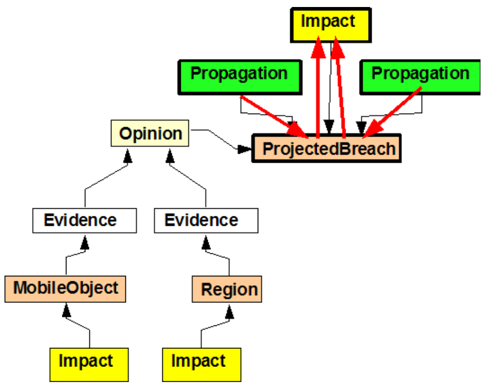

Figure 21. Accumulating propagations

In Figure 21 we see a graphical representation of the accumulation of Propagation belief masses into a single Impact frame by indirection through the relevant SituationElement (in this case the instance of ProjectedBreach).

As of yet, both the MobileObject and Region frames have only the default Impact associated to them. Unless the operator has increased the bias of the fusion, the impact of these frames on the operator is essentially unknown. However, both of these frames are believed to be highly relevant to the ProjectedBreach and that frame does have some already computed impact on the operator's mission. Here, we apply the last stage of the fusion algorithm and “back propagate” some of the impact of the higher level ProjectedRegion frame back to the frames that are cited as evidence for its impact on the operator.

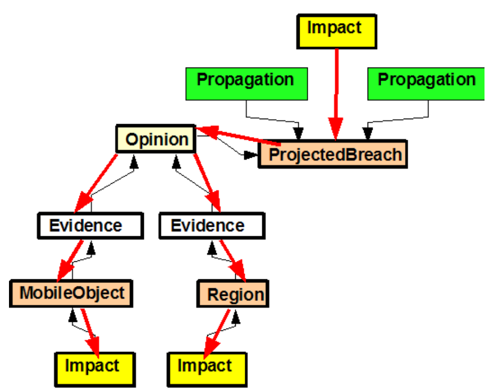

Figure 22. Back propagation

The flow of influence is shown graphically in Figure 22, depicting all of the indirection that eventually provides a path by which the higher level impact can provide influence for the supporting evidence. Note that Evidence frames can discount the back propagation, as may be the case when some supporting evidence has only a small influence on the production of the higher level frame with direct impact. Note also that the belief mass of the higher level frame, in this case the ProjectedBreach, is not diminished by back propagation. After the computation of the back propagation, the memex has completed the fusion pipeline and now contains a single, coherent Impact frame for each SituationElement.

7 Conclusions and Future Work

The proposed multi-agent system is designed for increasing situation awareness. This goal is achievable by inferring higher-level relations from low-level information and filtering the resulting superset of objects and relations to a core subset of elements that have a traceable impact on an operator's declared mission goals. Both the inference of higher-level relations and the filtering of the situation have the potential to reduce the cognitive load on the operator and may be of value in applications where an operator's cognitive resources are heavily burdened.

At the core of the evaluation of situation elements for relevancy to goals is a mission profile. The mission profile described here is a network structure that allows a profile author (likely the same person as the end operator) to declare goals and then decompose those goals into subgoals of arbitrary granularity. We believe that the graphical structure of the mission profile allows for easy visualization that is more intuitive to understand than collections of textual if-then rules. Looking to the future, we expect that mission profiles can be shared and re-used using GUI tools that can search and synthesize variations on profiles or wholly new profiles.

The mission profile is used directly by the proposed system as the core of a fusion process which unifies the opinions of the various agents attached to the system. The fusion process uses evidential reasoning techniques to combine evidence and influence into a complete and comprehensive summary which can then be emitted as output for ingestion into dependent systems that act upon the fused situation.

Moving forward, the proposed system will certainly profit from further theoretical improvements as well as lessons learned from experience or simulation. Specifically, the "back propagation" of evidence is intuitive, but somewhat *ad hoc*. The present method might be improved by a more formal treatment of supporting evidence that can retain the ease of authoring a mission profile. Also, our experience with our reference system is limited in both scope and size. Although we expect mission profiles to scale well, both in terms of usability and computation, a broader domain of application and a larger corpus of realistic profiles is needed before judgments on scalability can be conclusive.

8 Acknowledgements

The work presented here was supported by a Phase II Small Business Innovative Research award for Air Force AF06-031 under contract FA8650-07-C-6754. Any opinions, findings, and conclusions or recommendations in this paper are those of the author and do not necessarily reflect the views of the Air Force Research Laboratory.

9 References

[C88] I.D. Craig, Blackboard Systems, *Artificial Intelligence Review* 2(2), 1988.
[E88] M.R. Endsley. Situation awareness global assessment technique (SAGAT). In *Proceedings of the National Aerospace and Electronics Conference (NAECON)*, 1988.
[E95] M.R. Endsley. Toward a theory of situation awareness in dynamic systems. *Human Factors* 37(1), 1995.
[F98] J. Ferber, *Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence*, Addison-Wesley, 1998.
[NP01] I. Niles and A. Pease. "Towards a standard upper ontology," In *Proceedings of the 2nd International Conference on Formal Ontology in Information Systems* (FOIS-2001), 2001.
[PNL02] A. Pease, I. Niles, and J. Li. The suggested upper merged ontology: A large ontology for the semantic web and its applications. In *Working Notes of the AAAI-2002 Workshop on Ontologies and the Semantic Web*, 2002.
[G02] A. Gangemi, N. Guarino, C. Masolo, A. Oltramari, and L. Schneider. Sweetening Ontologies with

DOLCE. In *Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management* (EKAW 02), 2002.
[B02] K. Baclawski, M.K. Kokar, C.J Matheus, J. Letkowski, and M. Malczewski. Formalization of situation awareness. In *Proceedings of the 11th OOPSLA Workshop on Behavioral Semantics*, 2002.
[J97] A. Jøsang, Artificial Reasoning with Subjective Logic. In *Proceedings of the 2nd Australian Workshop on Commonsense Reasoning*, 1997.
[J09] A. Jøsang, Conditional Reasoning with Subjective Logic. *Journal of Multiple-Valued Logic and Soft Computing* 15(1), 2009.
[LP07] E.R. Lindahl and P. Petrov, Situation Awareness and Dynamic Group Formation of Agents and Humans Using Jabber. In *Proceedings of the 12th International Command and Control Research and Technology Symposium (12th ICCRTS)*, 2007.
[LZ07] E.R. Lindahl and Q. Zhu Epistemic Belief Frames in Distributed Effects-based Reasoning. In *Proceedings of the International Conference on Integration of Knowledge Intensive Multi-Agent Systems (KIMAS 2007)*, 2007.
[MKB03] C. J. Matheus, M. M. Kokar, and K. Baclawski. A core ontology for situation awareness. In *Proceedings of the 6th International Conference on Information Fusion*, 2003.
[M05] C.J. Matheus, M.M. Kokar, K. Baclawski, J.A. Letkowski, C. Call, M. Hinman, J. Salerno, and D. Boulware. SAWA: An assistant for higher-level fusion and situation awareness. In *Proceedings of SPIE Conference on Multisensor, Multisource Information Fusion: Architectures, Algorithms, and Applications*, 2005.
[KMB09] M. Kokar, C. Matheus, and K. Baclawski. Ontology-based Situation Awareness. In *Information Fusion*, 10(1), 2009.