Performance of a Simple Cooperative Individual Situation Assessment (CISA) with Respect to Information Sharing Strategy Metrics

Henry Hexmoor and Elisabeth Cuddihy
Department of Computer Science
and
Center for Cognitive Science
State University of New York at Buffalo
Buffalo, NY 14620

Abstract

When situation assessment is performed individually by cooperative, friendly agents working toward a commonly held goal, optimal strategies for sharing information are needed. The best strategies are often domain dependant with respect to constraints imposed by the domain itself. Appropriate strategies for information sharing can be selected by looking at the overall performance of a strategy based on a set of predefined metrics measuring qualities such as information consistency across the agents, information completeness among the agents, and timeliness of information availability to each agent. We have developed a prototype system, SAM-ISS, within a Theater Missile Defense scenario that we have used to investigate the utility of a set of metrics for determining the quality of three different information sharing strategies (ISS).

1.0 Introduction

The process of situation assessment is a common human activity. Within the cycle of perceiving one's surroundings, cognitively processing the input with other relevant information, followed by producing an appropriate action, situation assessment involves the processes of perception followed by understanding. These are difficult cognitive processes to model computationally yet these are essential components of an autonomous agent, be it human, an animal, or artificial.

Smith and Sage (1991) define situation assessment as "the process of detecting and defining an opportunity or problem." Although this definition alludes to the necessity of cognitive processing to realize the situation, the definition does not take into account the act of performing data aggregation and fusion over input from an array of different modalities. For this reason, we will define situation assessment as **the mental process of aggregating sensory, non-sensory, and a priori input to construct a mental representation of a situation**. A situation is any meaningful abstraction of personal, group, and environmental information relevant to the agent's set of goals.

Situation awareness is closely related to situation assessment. In general, assessment requires more direct attention as compared to awareness. For example, I can be aware that there are other people driving cars behind mine while on the expressway but I rarely need to assess how this affects me. Endsley (1988) defined situation awareness as "the perception of elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future" (p. 97). The main difference between situation awareness and assessment seems to lie within the level of attention that they require. Situation assessment requires a more focused management of attention in order to understand how the perceived elements are immediately relevant to any goals held by the agent. Situation awareness only requires an understanding of the perceived elements but does not require that this understanding be extended to surmise how these elements can effect the agent.

Many factors influence the process of situation assessment within an agent. A few key factors are: (a) the temporal relations between perceived sensory data; (b) the agent's expectations and prior experiences about the meaningfulness of the sensory data; (c) the agent's attention capacity; (d) the agent's mental attitudes such as beliefs, wishes, desires; and (e) the agent's emotions such as fear, anxiety, and joy. When processing or formulating a situation assessment, the agent's goals and objectives are important influencing factors that affect how available information is used to construct situations. As I am driving on the expressway, I maintain many goals, one being that I keep myself and my car safe and free from accidents. If it suddenly starts to rain while I am driving, my goal of automotive safety will be an important factor, coupled with my perception of sudden rainfall, will greatly influence by assessment of the current situation: a potentially hazardous driving situation.

In this paper we assume a commonsense model of agents involved in situation assessment. In this model, situation assessment is one of many things that an agent can do while in pursuit of fulfilling its own goals. We are making the assumption that these agents hold a friendly attitude toward each other, thus making them generally cooperative, but that they are also resource-bound and therefore, must be careful about their information sharing. Therefore, although their friendliness would tend to increase the amount of information that agents could share, resourcefulness with respect to timely accomplishment of individuals goals will tend to constrain the amount of information agents wish to send and receive.

We call this model Cooperative Individual Situation Assessment (CISA) between a group of agents. We define CISA as the exchange of information between friendly agents in order for each agent to individually perform unique situation assessment within its own "mind." This differs from other information sharing scenarios that could arise in a multiple agent setting by placing a certain constraints on the types of interactions agents can expect when sharing information. First, all agents can assume that the information received from other agents is qualified with respect to the task at hand rather than consisting of raw data. Secondly, all agents can assume friendly, rather than adversarial, interactions with other agents. By friendly interactions, all agents can assume that their communications with each other will follow the Cooperative Principle described by Grice (1975) in the following set of supermaxims:

1) the **quantity** of information provided is neither more or less informative than needed;
2) the **quality** of information does not include anything that the sending agent believes to be false or lacks adequate evidence;
3) the information exchanged is **relevant**; and
4) the **manner** in which the information is given avoids obscurity, ambiguity, and is brief and orderly.

This model makes no assumption of all the agents seeking a collective goal although it does allow for agents to do so if each agent individually holds the goal and decides to work toward it.

CISA relies on two factors. The first is the ability of each agent to individually perform situation assessment and the ability of all agents to share information. The first ability is required by the definition of our model. Without this ability, the model would reduce to that which has a central agency responsible for performing situation assessment based on information collected by other agents. The second ability is necessitated by the fact that some goals exist at the level of a group, community, or culture rather than within a single "leader" agent. In these situations, an agent is required to contribute its efforts toward the goal and by doing so, the sharing of information is required. Such examples can be found in flying a commercial airplane or team sports such as soccer

Central to information sharing among agents is the identification of strategies. We use the word strategy to have its commonsense meaning of *planned organization of methods*, or simply, *the plan for solving a problem*. Information sharing strategies are plans that use a set of methods for information sharing. As in a computer program, this program is a set of instructions, rules, procedures, and objects. Also, just as a computer program can be specified, the program of a strategy can be specified.

One particular method for specifying information sharing strategies is through the use of domain independent teleological questions. Such questions include

a) data about the quality of the information, such as:
   - Is the information reliable?
   - Is the information current?
b) data about who to share the information with, such as:
   - Should sharing be only with those who request information?
   - Should sharing focus on informing those who probably don't have access to the information?
   - Should sharing only be between a selected group of individuals?
   - Should sharing be focus with agents who are least busy?
c) data about when to share the information, such as:
   - Should sharing be done only upon receiving a request?
   - Should sharing be done whenever something of interest has happened?
d) data about how to share the information, such as:
   - Should sharing be expected every time a request is received or can requests be denied?
   - Should sharing be done through mediators?
e) data about what parts of the information to share, such as:
   - Should only information relevant to parts of unsolved goals be sent to others?
   - Should updates to previously sent information be sent out?
   - Should information sent only be that which is not otherwise available?

Such questions as above can be used to construct strategies for filtering information to be sent and received. By doing so, redundant, unneeded, and out of date information can be reduced in the process of communication. In order for agents to successfully filter information based on teleological questions, each agent needs to construct a mental model about the agents it communicates with. This requires the agent to assess the needs, requirements, and intents (i.e., goals, plans, desires, deadlines) of other agents plus maintain such awareness about itself as well. Likewise, the agent also needs to be aware of domain specific constraints in order to answer these types of teleological questions. If the domain imposes a high cost on message passing, agents may seek to limit what they share and how often they share to the most minimal acceptable level. In a domain that requires global consistency of information across the agents, they will need to seek a strategy that maximizes their ability to get updated information when their information becomes outdated.

In order to determine the effectiveness of an information sharing strategy at meeting domain constraints, we need to define a set of metrics that can be objectively measured during the process of agent communication. Such metrics that we have examined in our current model include the timeliness of information availability, the consistency of pieces of information across the agents, and the completeness of the set of information across the agents. It is our belief that an effective set of performance metrics can be used to automate the selection of an appropriate information sharing strategy.

2.0 Background and Motivation

Situation assessment is studied in defense systems [Wesson, et al, 1981, Smith and Sage 1991, Cohen, et al 1994] and Wesson , et al, show how SA is improved by information sharing. Situation Assessment among cooperative agents can be considered to be a process of Cooperative Distributed Problem Solving (CDPS). CDPS involves an informally organized group of problem solvers, solving problems that are beyond the scope of a single problem solver Decker (1995). Approaches to CDPS differ amongst methods which employ negotiation among nodes, functional accuracy, use of communication patterns to aid cooperation, and reasoning about other agents, among other techniques. Prior work in CDPS by in large specify a single technique for information sharing. For example, in the blackboard approach, each application specifies a fixed communication and information sharing protocol. The very nature of blackboard architecture restricts the information sharing strategy to use the blackboard as a common information sharing data structure. Our problem and approach loosely fits in CDPS. Among the aspects of CDPS, functional accuracy is a central focus for us. Functional accuracy is the idea that problem solvers will work despite inconsistencies in the information being shared. A related thread of work to SAM-ISS is DRESUN, Carver and Lesser (1995). DRESUN is an extension of DVMT project which models assessment of vehicle locations using distributed sensing.

In addition to strategies for filtering the information received as in Malone (1987), we are interested in strategies that filter information to be shared. We are after strategies that filter redundant, useless, old information from being disseminated. In order to determine whether/what/how/who/when each agent must assess the needs, requirements, and intents (i.e., goals, plans, desires, deadlines) of other agents and maintain such awareness for itself as well.

We have also drawn from the literature on mutli-agent environments. A solitary agent's ability to learn is limited by its experiences with its environment. As we want to eventually created a system of agents that are capable of determining appropriate ISS, we want to maximize single agent's experiences in knowing what ISSs work well under specific constraints. Agent collaboration for quicker learning has been examined among user interface agents (Lashkari, et. al. 1994). A key problem in information sharing among a large group of agents is making relevant information easily available to agents seeking it out. Foner (1995) has investigated means for improving inter-agent communication within a large society of agents. He proposes the use of a concept space with respect to types of information available which is grouped into clump spaces based on semantic relatedness.

Other approaches to information filtering have included evolutionary model approaches where the successful filters are evolved during direct usage. The *Amalthaea* system developed by Moukas (1997) is capable of doing data discovery and information filtering with respect to learning about a person's interests by observing their interactions with a World Wide Web browsing program, particularly, by looking at their hotlist, browsing history list, and keywords contained within URLs. Within the *Amalthaea* system, there exists a class of agents who are responsible for information filtering and a second class of agents that do information discovery. *Amalthaea* uses an ecosystem model with evolutionary strategies defined on the agents, allowing for an approach similar to a genetic algorithm for improving agent fitness and overall system success.

3.0 The SAM-ISS Testbed

In order to examine the performance of our set of basic information sharing strategies for CISA, we developed a prototype testbed in Microsoft Visual Basic on an MS Windows platform. This allowed us to rapidly prototype a user interface and the basic agent interactions. The prototype utilizes a model from a Theater Missile Defense scenario involving the exchange of information pertaining to the identification of Surface to Air Missile (SAM) units comprised of three specialized vehicles.

Within our testbed, the SAM-ISS prototype, we have implemented three modules: a domain simulator, a ISS performance analyzer, and a graphical user interface (GUI) as shown in Figure 1. Two of the three modules, the simulator and the GUI rely on the SAM identification model. The simulator implements the behavior of the model and of all agents and their communications within the model. The GUI provides a visual display of the model and allows the user to interact with the simulator and the ISS performance analyzer. The ISS performance analyzer collects information pertaining to the success of each ISS strategy used within a scenario run.

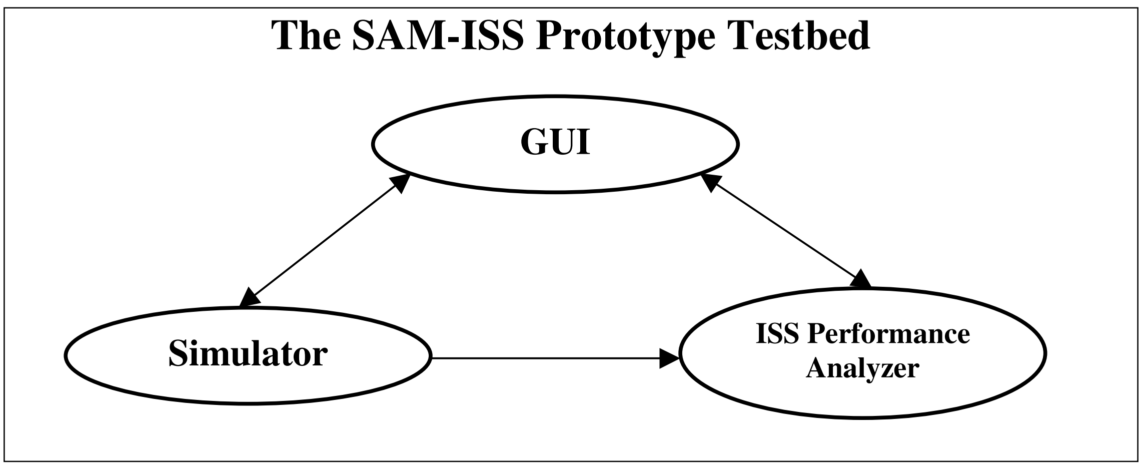

Figure 1: The components of the SAM-ISS prototype testbed.

3.1 The SAM Model

The testbed is currently modeled around an information sharing problem within the domain of Theater Missile Defense. This problem consists of three stationary sensing nodes on a battlefield that are each on the look out for Surface to Air Missile units (SAMs) comprised of three specific ground vehicles: a power vehicle, a communications vehicle, and a transporter vehicle.

Each sensing node is able to see only a portion of the battlefield, in a conical cross-section with the apex at the node, extending outward across the field, away from the node. This is marked on the battlefield as three triangles of visibility extending out from each of the three nodes. Due to the spacing of the nodes, some areas of the battlefield are out of view of all nodes, some areas are covered by one node, and other areas have redundant coverage. Band-shaped regions on the battlefield corresponding to the amount of node visibility coverage have been marked as low, medium, and high visibility. The high visibility region has complete and redundant coverage throughout the region, with over half of the region covered by 2 nodes. The medium visibility region has incomplete coverage with small pockets of redundant coverage and small pockets of no coverage. The low visibility region has large areas totally just over half of the region, with no visibility coverage; the other areas in the low visibility region have coverage only by a single node (see Figure 2).

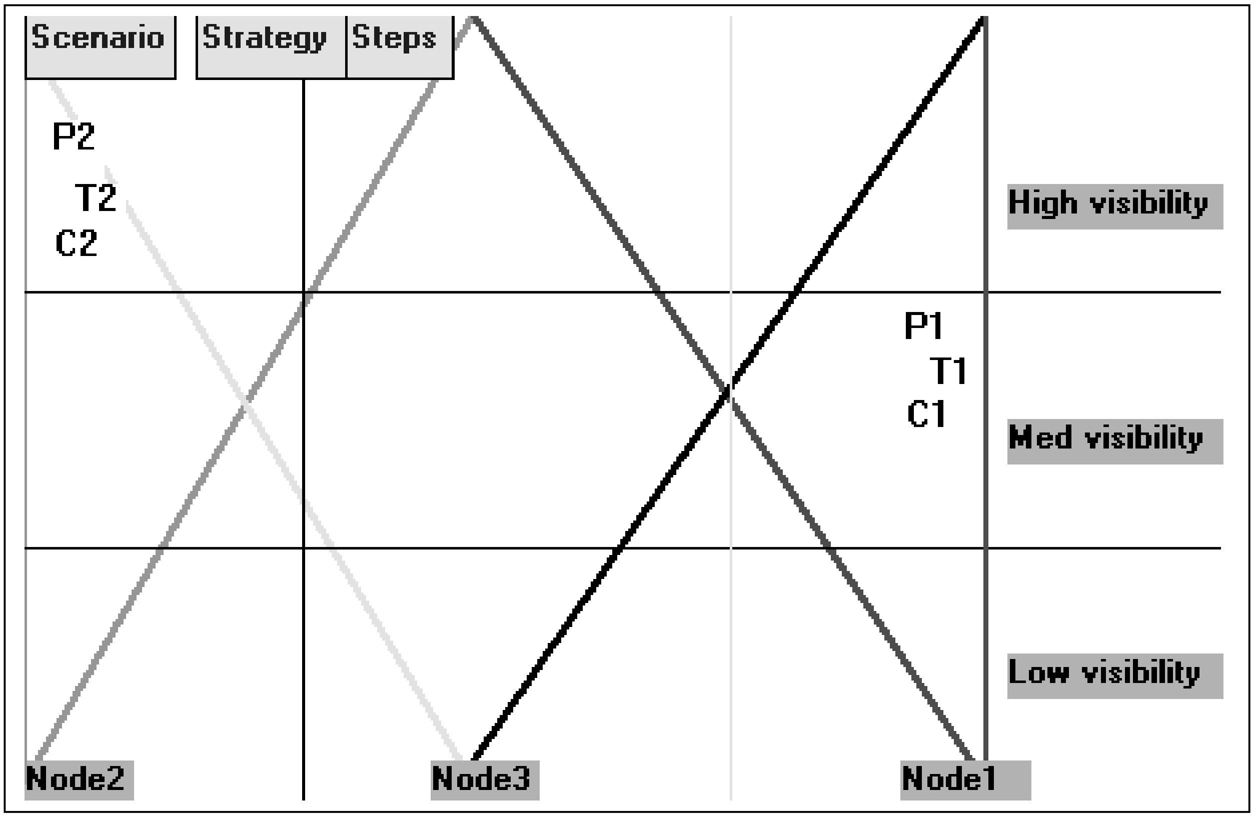

Figure 2: Battlefield display showing nodes at the bottom of the field and the range of the node's sensors extending up from each node. Three visibility coverage regions have been marked on the battlefield based on the percentage of node sensor coverage within that area. These horizontal coverage regions have been marked as high, medium, and low visibility. Two groups of three vehicles, marked C1, T1, P1, and C2, T2, P2 are also shown. These represent the vehicles within two separate SAM units.

The six vehicles move through the field in accordance to one of several predefined scenarios. In each scenario, the vehicles travel in two groups of three, where each group is comprised of a transporter-erector-launcher vehicle (marked T), a communications vehicle (marked C), and a power vehicle (marked P). Such a grouping is called a SAM, a surface to air missile unit. These two SAMs can be positioned anywhere in the field prior to running a simulated scenario. As the vehicles move through the field during a scenario, the nodes detect the vehicles that exist within their field of view. If a node detects three vehicles moving in close formation such that the unit is comprised of a T, C, and P vehicle, the node will surmise that it is viewing a SAM unit.

A SAM unit can be in one of three modes while the simulation is running. These modes are moving, arming, and launching. When a SAM is arming or launching, that SAM is said to have had an event. Nodes that have detected a SAM are able to detect SAM events. Within this model we assume 100% sensor accuracy. If a vehicle is in a visibility region for a node, it will always be seen. Likewise, if the three vehicles comprising a SAM are detected together by one node within its visibility region, then the node realized that it has spotted a SAM.

3.2 The Scenarios

Three scenarios for vehicle movement have been defined within this testbed, two of which have been used for the initial ISS analysis described within this paper. The first scenario is one in which one of the SAM units travels horizontally from left to right across the battlefield while the other unit travels horizontally from right to left. For this scenario, the SAM units can be placed within the low, medium, or high visibility bands, or in any combination of two bands. Within this scenario, a SAM unit stays within a single visibility band throughout the course of the scenario. The second scenario places one SAM unit such that it travels vertically through all three visibility bands, from low to high while the other SAM unit travels horizontally across the battlefield within one of any of the three visibility bands.

These two scenarios allow for ISS tests in circumstances where this is little information to be shared up through scenarios where information could potentially be shared continuously.

3.3 The ISS Strategies

In our implemented system we used the following specification as a specialized production rule:

```
IF < sender-filter>
THEN <sender> GENERATE
           FOR < recipient> THAT MEET <filter-potential-
      recipient>
SEND <filter-recipient>
```

The <sender-filter> is the set of conditions under which the message source will send a message. These conditions filter messages that the sender may send out. An example of specific condition *is the range data for TEL2 is less than 2 miles and gathered less than 2 seconds ago*. A more abstract example is *target must be identified*. In general, these are the (a) temporal conditions/constraints for communication between agents, (b) atemporal conditions/constraints for data to be communicated between agents, e.g., reliability-threshold, information-age, availability. <sender> is the source of information, i.e., who sends the information. <message> is the composition format of the message, e.g., id:sender. <recipient> is the receiver of the message. It can be one or more sites. <filter-potential-recipient> are the conditions for selecting recipients known by all agents (i.e., common knowledge) on the recipients, such as the nodes that needs the info most, lacks the info the most, respond to solicitation, relevance-to-agent, e.g., no*t busy* or *interested*. <filter-recipient> are conditions that the recipient might impose such as *no messages on TEL after 10pm*. The conditions in the rule encoded ISS teleologies.

Three information sharing strategies have been defined among the nodes. These have been named: broadcast, event-driven, and unique. When using broadcast ISS, each node, on each time step, broadcasts all information it currently detects to all other nodes. Such information includes both observation of a type of vehicle and overall observation of a SAM unit. During event-driven ISS, a node will only broadcast information to all other nodes when a SAM is observed in the state of arming and launching. Both broadcast and event-driven ISS send information to all other nodes. Unique ISS differs from these by sending information only to predefined neighboring nodes. Within this testbed, there are three nodes where both end nodes are defined to only have one neighbor, the center node, and the center node is defined to have two neighbors. In unique ISS, information is shared with a neighbor when a node determines that the information is unique to itself, and therefore outside of the neighbor's field of view.

The teleological polices used by these three information sharing strategies are outlined in Figure 3. All three strategies share the same set of information: identification of each vehicle it is able to see, whether or not it has detected any SAMs, and whether or not any detected SAMs have been observed to be arming/launching. This information is always shared in an unsolicited manner such that an agent will share information without requiring a request from other agents. What differs between these three ISSs are who and when the information is shared. For both broadcast and event-based, the information is shared with everyone. For unique ISS, it is only shared with the predefined neighbors, limiting the number of messages sent and received during each phase of communication. Broadcast and unique ISS always share information any time anything of interest is noticed. Thus, each time an agent observes the battlefield, if a vehicle is identified, a SAM is detected, and/or an arming and launching event is observed, this information is shared. Event-based differs by only sharing information when SAMs are observed having an arming and launching event, limiting the communications only to times when information of highest importance needs to be communicated.

	WHO?	**WHEN?**	**HOW?**	**WHAT?**
Broadcast ISS	Everyone	when anything is detected	share information unsolicited	- vehicle detected - SAMs detected - events noticed
Event-based ISS	Everyone	only when an event is detected	share information unsolicited	- vehicle detected - SAMs detected - events noticed
Unique ISS	Predefined neighbors	when anything is detected	share information unsolicited	- vehicle detected - SAMs detected - events noticed

Figure 3: The teleological-based policies for the broadcast, event-base, and unique information sharing strategies.

3.4 The Domain Simulator

The simulator is responsible for managing the microworld in which all testbed agents exist. This includes a domain simulator for the microworld, an agent simulator, and a communications si

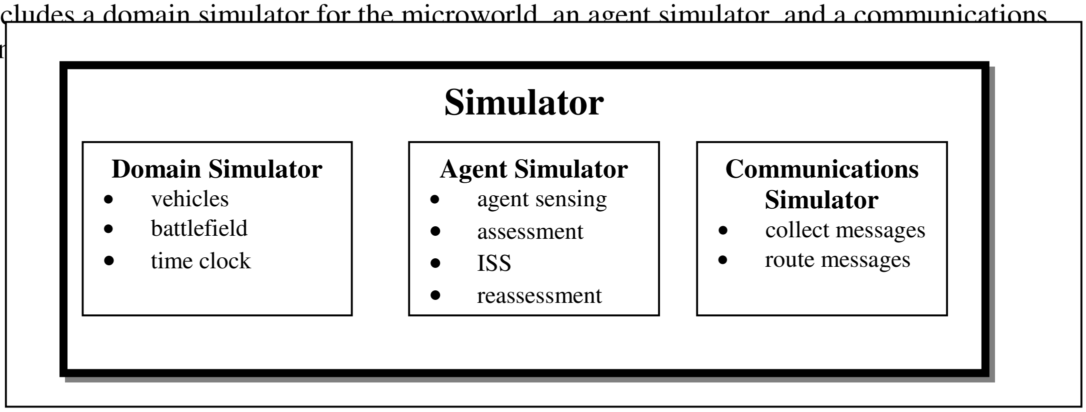

Figure 4: The simulator, composed of a domain simulator, an agent simulator, and a communications simulator.

The domain simulator is responsible for maintaining a global time clock for the microworld. This time clock is used in determining the actions of objects and occurrence of events. The domain simulator is also responsible for maintenance of information pertaining to the particular domain being simulated. Within our current prototype for Theater Missile Defense, the domain simulator is responsible for maintaining the vehicles and the battlefield. The domain simulator maintains a set of three different scenarios in which the SAM units participate. Each scenario contains a movement pattern for the SAM units. The domain simulator also maintains the occurrence of events among the SAM units. A single SAM is capable of having an arming and launching event. The simulator currently has each SAM create an event once every set number of time clock cycles.

The second part of the simulator is the agent simulator. Within our current testbed, this is responsible for simulating agent sensing and situation assessment. An agent's sensors in the current simulator always give correct readings, identifying vehicles on the battlefield as either power, communications, or transporter vehicles and identifying the occurrence of an arming and launching event. After an agent senses, it assesses the situation by deciding if it has seen a SAM and if the SAM is in the process of arming and launching. The agent then determines if it has any information to communicate, based on the ISS being used. If so, communication takes place. If the agent receives any additional information during the round of communication, it reassesses the situation.

The final part of the simulator is the communications simulator that is responsible for the passing of messages between agents. This part of the simulator delivers messages from one agent to another and simulates agent's assimilation of messages.

The overall algorithm followed by the simulator is summarized below in figure 5.

```
For i=1 to total number of clock cycles
      Simulate vehicle movement according to scenario
      Simulate sensing for each agent
      Simulate agent’s situation assessment
      Simulate agent’s determination of what information to share
based on          current ISS mode
      Simulate interagent communications
      Simulate agent’s situation reassessment
```

Figure 5: Summary of the simulator algorithm.

3.5 The ISS Performance Analyzer

The analyzer module of the testbed is responsible for assessment of communication strategies such that a strategy can be analyzed both during and after a completed scenario. Analysis during a running scenario can be used to do real time switching of information sharing strategy such that a better level of performance is achieved. Analysis after a scenario provides a set of metrics collected during that scenario.

Our current implementation only includes a basic set of five ISS metrics that are gathered throughout the course of one running scenario and then displayed to the user. These metrics are:

1. Timeliness - the amount of time before a node becomes aware of a certain piece of information;
2. Consistency - the number of nodes aware of a certain piece of information;

3. Completeness - the number of pieces of information that a node is aware of as a function of time;
4. Cumulative consistency - the percentage of nodes aware of individual pieces of information as a function of time;
5. Cumulative completeness - the average number of piece of information each node was aware of over time.

Within this testbed, the information measured with respect to timeliness was the amount of time from the beginning of the scenario until a particular node first became aware of a particular SAM complex. The information measure with respect to consistency was knowledge of SAM unit; thus consistency measure the number of nodes aware of each SAM. Completeness was with respect to the total number of vehicles each node was aware of. Therefore, cumulative consistency measured the overall percentage of time during the scenario that each node was aware of either SAM unit 1, SAM unit 2, or both. Likewise, cumulative completeness measured the number of vehicles that each node was aware of time as a percentage of time with respect to awareness of a number of objects between 0 and 6, inclusive.

3.7 The Graphical User Interface

The user interface provides a display of the microworld, the objects within it, controls for both manipulating the microworld and controlling the simulator, and for displaying the metrics collected in the analyzer.

The visual display of the microworld provides a bird's eye view of the battlefield. The three communications nodes are fixed in stationary positions at the bottom of the battlefield, with their conical visibility areas marked on the field, extending and expanding upward. The vehicles are marked T1, C1, and P1 for the SAM unit 1 vehicles and T2, C2, and P2 for SAM unit 2. The user can place these vehicles anywhere within the bounds of the battlefield although the three preprogrammed scenarios are designed to have the starting location of the vehicles in certain regions of the field, and clumped together as distinct SAM units. As a scenario runs, the locations of the moving vehicles are displayed in the battlefield and the vehicles change their display when an arming and launching event occurs within a SAM unit.

The user interface allows the user to choose the information sharing strategy, which will dictate communications decisions between nodes; the scenario, which will determine the movement patterns of the vehicles during the scenario; and the number of time clock steps to be performed in the scenario. After a scenario has run, the user interface provides displays of the metrics collected during that scenario run.

4.0 Results

Data was collected for the first and second scenario using broadcast, event, and unique ISS such that all combinations of SAM travel within regions of visibility coverage where tested. The data collected was based on the five metrics discussed in section 3.5, information timeliness, completeness, consistency, plus cumulative completeness and consistency.

4.1 Scenario 1

In scenario 1, where both SAMs travel horizontally within one region of visibility coverage, 15 data runs where collected. These consist of: three runs for broadcast ISS, with the SAMs placed in high, medium, and low visibility regions; and six runs for both event and unique ISS each with all six combinations of SAM placement in the visibility regions.

For broadcast ISS, whenever a vehicle or an entire SAM unit is visible to at least one agent, this information is broadcast to all agents. Likewise, if an event is observed, all agents will know about the event after information is shared. With broadcast ISS, the percentage of time in which all nodes are informed of SAMs and SAM events is determined by the percentage of visibility coverage within the region. For the data run in which the SAMs were placed in the high visibility region, all nodes knew of both SAMs 100% of the time, corresponding to the 100% coverage available in that region. For the run with both SAMs in the medium visibility region, all nodes knew of both SAMs 63% of the time; for low visibility, only 11% of the time. Information between the nodes is always complete and consistent with broadcast ISS but at the cost of the highest possible level communication traffic between the nodes.

The data collected for event ISS showed the effects of limiting communications only to when a SAM event is detected by a node. By doing this, the consistency between all nodes is decreased to the percentage of observed events. A node's knowledge of a SAM is based on a function of the SAM's percentage of visibility and percentage of events performed by the SAM. For SAMs traveling in the low visibility region of the field, it is rare that more than one node ever knows about the SAM due to the infrequency of event observation. In such a scenario, nodes suffer a great delay in timeliness (approximately 50 steps) of being notified by other nodes of a SAMs existence. For SAMs traveling in the high visibility region, the timeliness delay is greatly reduced due to the SAMs' travel in an area that is completely monitored by the nodes. No more than ten steps where needed for any node to become aware for the first time of any SAM. None the less, all three nodes where cumulatively aware of both SAMs only 34% of the time. For 54% of the time, all nodes where aware of at least one SAM unit. Information between nodes in the event ISS will only be consistent when SAM events occur and are detected. If the frequency of detected SAM events is very low, consistency will suffer greatly. Despite this loss in consistency, the level of communication between nodes is reduced dramatically such that only the most essential information is being communicated. Unfortunately, by limiting communications only to event detection, the level of communication is too low for maintaining consistency across the nodes.

The data collected for unique ISS shows a marked increase in information consistency across the nodes due to the more frequent exchange of information. For SAMs traveling in the high visibility region, consistency is increased to 86% for all nodes knowing of both SAMS and 93% for all nodes knowing of at least one SAM. For SAMs traveling in the low visibility region consistency is much higher than that for event ISS although delays in timeliness for a node to first receive knowledge of any node is still great. Agents using unique ISS will communicate during every time step that they see something of interest, making unique ISS more like broadcast than event-based with respect to when communication is done. But, unique ISS limits the total number of communication messages sent by only sending information not available to neighboring nodes rather than broadcasting to all. Perhaps a noteworthy observation is that with unique ISS, detection is roughly 85% of broadcast ISS. We see that we have traded off 15% detection rate for a 30% saving in communication. The saving in communication is due to node 1 and 2 only communicating to node 3 and not to each other, i.e., 2 of 6 communication paths are blocked.

4.2 Scenario 2

In scenario 2, SAM 1 travels vertically upward from the lowest visibility region to the highest while SAM 2 travels horizontally within a single visibility region, from left to right. Nine data runs where done for this scenario, 3 for each of broadcast, event, and unique ISS such that all three visibility placements of SAM 2 where accounted for.

For broadcast ISS, the data is as expected. As soon as a SAM moved into a region of visibility coverage, all nodes would find out about its existence. Consistency was directly related to the percentage of visibility within the area the SAMs travelled in. As before, a high level of communication was required to maintain consistency of information between the nodes.

For event ISS and unique ISS, the performance was similar to that within scenario 1. Event ISS was not able to maintain a reasonable level of consistency and, in fact, sometimes an entire run will go by leaving at least one node completely unaware of one of the SAMs. For the low visibility run using event ISS, no node was ever aware of both SAMs, showing a degradation in detectability based on one SAM moving within the visibility range of only one or two nodes (and occasionally outside any node's visibility), rather than within all three. Unique ISS showed to improve in consistency as before.

5.0 Discussion

Broadcast ISS, due to its policies of always sharing information with everyone, is able to maintain completeness and consistency of information across all nodes with the most information timeliness possible but at the cost of extremely high communications traffic. In fact, many of the messages sent using broadcast ISS are redundant as a node will send out information during every clock cycle that it detects anything with its sensors. Thus, if a node sees a communications vehicle during the first, second, and third cycles, it will send out messages to the other nodes on all three cycles saying that it saw a communications vehicle, despite the redundancy. In our current model, the agents do not possess any memory so this redundancy is necessary but in a scenario with more complex agents that have the ability to remember information, this ISS method would tend to provide an overwhelming amount of redundant information. This method also puts a greater strain on the communications network. The amount of information communicated throughout the course of a scenario is a function of:

(cycles) * (no. nodes) * (no. nodes – 1) * (info packet size) / (% of visibility)

which is clearly dominated by the number of nodes participating in the scenario, giving a communications growth of N^2 where N equals the number of nodes.

Event-based ISS, which dictates that information is only shared when an event is observed, shows the performance of an ISS that is limited only to sharing information when something of critical importance occurs. This limits the amount of communications traffic considerably but at a cost to information timeliness, completeness, and consistency. Although all nodes are aware of critical events, when no critical events are observed, the nodes have neither a complete nor a consistent picture of the battlefield. If critical events are difficult to detect because the are occurring in an area with very low visibility coverage, the level of information completeness and consistency drops considerably. In the case of events occurring at every time cycle, event-based ISS will behave exactly as broadcast ISS. Yet, the assumption of event-based ISS is that critical events are occurring only over a fraction of the total time. Information completeness and consistency will depend on the percentage of time that critical events occur and can be observed.

Unique ISS manages to reduce the amount of communications traffic between agents from broadcast ISS while providing higher information timeliness, completeness, and consistency than found with event-based ISS. The amount of information communicated over the course of a scenario is reduced only by sending to a maximum of two neighboring nodes. This reduces the communications growth to a linear function of the number of nodes. Information completeness and consistency will fair as well over large numbers of nodes but if the domain constraints can be satisfied by allowing neighborhoods of nodes to maintain local completeness and consistency, than this ISS methods performs better than the other two with respect to maintaining lower communications network traffic.

Of the set of metrics used in analysis, cumulative completeness, measuring the number of objects that each node was aware of over time, cumulative consistency, and measuring the percentage of time each node was aware of each SAM unit served to be accurate indicators of an ISSs ability to distribute information across all of the nodes. Neither metric, as we have implemented them, are useful for identifying completeness and consistency over neighborhoods of nodes such as for measuring the effectiveness of unique ISS over a larger number of nodes. This would instead require cumulative measures over each of the neighborhoods in addition to, or as a replacement for, the global cumulative measures. Additional metrics for collecting the amount of overall network traffic would be helpful in determining the cost of communications. It is possible that this could be used in conjunction with cumulative completeness and cumulative consistency to create a system that was able to choose an appropriate ISS on the fly based on the current domain constraints.

Our timeliness metric measured the amount of time that passed from the start of a scenario before a node first becomes aware of a particular piece of information. When vehicles traveled in low visibility areas, timeliness scores where consistently poorer. This information can be useful in meeting domain constraints, particularly when higher levels of consistency and/or completeness are needed. In such cases, a strategy such as event-based ISS would not perform well and a system could use such knowledge to switch to another ISS that performs better for completeness and consistency when vehicles are harder to detect.

6.0 Summary and Conclusion

We presented a model and a testbed for situation assessment. Many parameters affecting information sharing and strategies for information sharing were discussed in the form of generic questions. Clearly we are moving in the direction of developing a general framework that is applicable to a large class of problems. Our testbed is missing a few key parameters. If communication and deliberation did not cost time and effort, communication should be kept at maximum. However, in the real world, time and effort are critical factor. We showed a tradeoff of 15% detection rate for a 30% saving in communication. Cost and other factors such as models of other agents need to be included in the testbed to make it more useful.

Our approach assumes the cooperative principle and Gricean maxims used in both teaching and analyzing human discourse. However, in order for our information sharing strategies to be modeled such, we must add a great deal of reasoning and resource-sensitivity to our agents. For example, we know what it means to be "relevant" but can not be modeled easily. Without attempting to solve the holy grail of AI we plan to (a) develop a more robust model of cooperative individual situation assessment, and (b) model simple forms of principles from language planning literature in linguistics such as in Tauli (1968) for ISS.

7.0 References

[Carver and Lesser 1995] N. Carver and V. Lesser (1995) The DRESUN Testbed for Research in FA/C Distributed Situation Assessment. In V. Lesser (editor) Proceedings of the First International Conference on Multi-agent Systems, pp. 33-40, MIT Press.

[Cohen, et al 1994] MS Cohen, et al (1994) A Cognitive Framework for Battlefield Commander's Situation Assessment, **US Army Institute for Behavioral and Social Science**, TR 1002.

[Endsley 1988] M.R. Endsley (1988). Design and Evolution for Situation Awareness enhancement. In 23rd annual **Human Factors Society**, Pages 97-101. Human Factors Society.

[Decker 1995] K. Decker (1995) Environment Centered Analysis and Design of Coordination Mechanisms, PhD Thesis, University of Massachusetts.

[Foner 1995] L. Foner (1995), Clustering and Informaton Sharing in an Ecology of Cooperative Agents, AAAI-95 Spring Symposium on **Information Gathering from Distributed Heterogeneous Environments**, Stanford University.

[Grice 1975] H.P. Grice (1975) Logic and Conversation, In P. Cole (editor) **Syntax and Semantics** Vol 3, pp. 41-58, Academic Press, New York.

[Lashkari et al, 1994] Y. Lashkari, M. Metral, and P. Maes (1994) Collaborative Interface Agents, Proceedings of **AAAI-94** Conference, Seattle, Washington 1994.

[Malone, et al 1987] T. Malone, K. Grant, F. Turbak, S. Brodst, M. Cohen (1987). Intelligent Information-Sharing Systems, In Computing Practices, pages 390-402, ACM.

{Moukas, 1997] A. Moukas (1997) Amalthaea: Information Discovery and Filtering using Multiagent Evolving Ecosystem, International **Journal of Applied Artificial Intelligence** (to appear).

[Smith and Sage, 1991] C. Smith, A. Sage (1991). A Theory of Situation Assessment for decision support. In Information and Decision Technologies, No. 17, pages 91-124. Elsevier Science Publishing, North Holland.

[Tauli 1968] V. Tauli (1968) **Introduction to a Theory of Language Planning**. Uppsala: L Almqvist and Wiksell.

[Wesson, et al 1981] R. Wesson and F. Hays-Roth, J. Burge, and C. Statz, C.A. Sunshine (1981) Network Structures for Distributed Situation Assessment, IEEE transactions on **Systems, Man, and Cynernetics** 11 (1) pp. 5-23 (Also published in Readings in Distributed Artificial Intelligence, Alan Bond and Les gasser (editors), pp. 71-89, Morgan Kaufmann, 1988).