International Journal of Innovative Computing, Information and Control
Volume **10**, Number **1**, February **2014**

ICIC International ©2014 ISSN 1349-4198
pp. 183–210

SHARED SITUATION AWARENESS ARCHITECTURE (S2A2) BASED ON MULTI-RESOLUTIONAL LEVEL ARCHITECTURE

Hossein Parvar[1], Mehdi N. Fesharaki[1] and Behzad Moshiri[2]

[1]Department of Computer Engineering
Science and Research Branch, Islamic Azad University
Tehran, Iran
parvar@mshdiau.ac.ir; mehfesharaki@yahoo.com

[2]Control and Intelligent Processing, Center of Excellence
School of ECE
University of Tehran
Tehran, Iran
moshiri@ut.ac.ir

Received May 2012; revised September 2012

Abstract. *In virtual organizations, staff are members of a team who should have situation awareness (SA) on factors related to their work in complex adaptive systems (CAS). Therefore, if all team members achieve a high degree of shared SA, this helps enormously to achieve virtual collaboration, virtual organization and self-synchronization. This paper presents a new architecture for shared SA (S2A2) based on intelligent system architectures with multi-resolutional levels using data fusion and SA models. This unique architecture not only specifies the essential components of the system and their required interactions, but also determines thresholds as necessary conditions for team members to achieve higher degrees of shared SA. Regarding the interactive dynamics in architectural processes, the cooperation and synchronization bottlenecks are presented as threshold values. In addition to the required data and information for S2A2, these threshold values mostly deal with organizational culture and other cognitive parameters that are necessary for collaborative organizations. We have used Garbage Can Model (GCM) introduced in the organizational theory to determine these threshold values. The simulation results show some norms in this respect, which must be at least 50% efficient.*
Keywords: Shared situation awareness, Information fusion, ELF model, Garbage Can Model, Intelligent system architecture, Virtual organizations

1. **Introduction.** Situation awareness (SA) is a form of knowledge obtained from current information related to the surrounding environment which is essential in achieving any goal. The key factor in the process of decision-making is efficient SA and in doing so, mental models play a pivotal role [1,2].

Organizations based on team cooperation, including virtual organizations, are characterized by handling the dynamics of environmental changes using shared SA (SSA) for adaptation to these changes. In the most network centric organizations like virtual organizations which encompass a dynamic set of individuals defined around set of shared resources, conditions and rules, SSA means the same view shared by different individuals in relation to a set of specific activities including common concerns and requirements which vary in sociology, scope, structure, size and duration. In other words, individuals obtain the SSA by observing a shared operational picture and their own mental processes [3]. Hence, the SSA consists of identical or similar views by members of the team related to a situation [4].

High degrees of SA for all members of a team can only be achieved when they have the same perception, comprehension, projection and explanation of the situation. This increases the resilience and robustness of virtual organizations. Every member of a team has a sub-goal related to his or her role, which will lead to the final goal. On the other hand, SA in team members increases the operation tempo, decreases operation cost and risk, and boosts the team members general efficiency in virtual organizations. In many complex system applications such as healthcare, emergency management, disaster management, crisis management, emergency materials distribution, and logistics management where collaboration between team members is necessary to achieve to teams objectives, SSA is the key factor in proper decision making [5-7].

To achieve SA, the main objectives of a special series of works and the required sub-goals are defined. Specifying the required objectives, the SA for decision making is used by goal-directed task analysis [1]. In this process, SA requirements are defined by dynamic information related to the main objectives or the operator's sub-goal. For each sub-goal, there is a set of SA elements related to each member. Since sub-goals overlap, members will take the same course in reaching the goal. Therefore, each member shares a subset of his or her information with other members and this process forms the coordination of the team. Figure 1 shows the SSA in Goal-Directed Task Analysis (GDTA).

SSA is affected by environmental, team, and individual factors. Environmental factors include anxiety, stress, workload, exhaustion, team size, physical place, uncertainty, ambiguity, automation, capabilities of the system, and complexities of the interface. Team factors include communications, cooperation tools, shared mental models, team processes, team dimensions, trust and commitment. Individual parameters include mental models, memory, knowledge, cognitive sources, experience, exercise, perceptive and problem solving capabilities, individual skills, decision making skills, mental and physical state, perception beliefs, expectations and capabilities, and the intention of the operation [4]. Using GTDA, these factors are defined as threshold values for the fulfillment of the prerequisites in S2A2. GDTA defines roles for each team member based on the main goal and sub-goals. It also specifies required cognitive thresholds for the coherence of the operational process of the organization. Therefore, based on the analyses pertaining to individuals training and capability, leadership, doctrine, organization, materials and facilities, roles are defined, and cognitive thresholds are formed based on individual, team and environmental factors.

Concerning environment changes with time, users requirements change in different situations. Therefore, different components such as dynamic community of human observations, available and sensing resources [8] should be added or removed without disrupting or stopping the system. The mobility of components is another characteristic of virtual

Figure 1. Shared situation awareness (SSA) in goal-directed task analysis (GDTA)

organizations. Therefore, lifetime, connection bandwidth and the energy of the sensing resources used by different components in the system vary [3]. A formidable challenge in such systems is the limited bandwidth whereas a flood of received data require extra bandwidth. As a result, the required components of SA may not be obtained in the specified time [4] and the virtual organization resilience will be threatened by the lack of timely SA. The reason for this is that if only one member of the team has a weak SA, a critical error occurs at a critical moment. As a result team's success will be compromised.

In our previous works, in order to achieve mutual SAA for agents interaction, we modeled trustworthiness as a key feature in virtual organizations robustness [9]. The impact of trust, intention based trust and norm information sharing are examined as cognitive parameters. These parameters are used for governing the New Product Development (NPD) relationship as a virtual organization [10,11].

We aimed to present SSA as a system architecture, providing a technical system in support of on-line collaborative decision making for various real-time network centric environments. Furthermore, it plays a key role in the evaluation of collaborative decision-making, with simulations performed to support this evaluation. Though there are many works with SSA as the main theme, few were dedicated to its operationalization.

The major contribution of this work compared with other models and architectures is considering the dynamics of awareness in which planning for achieving SA is carried out in an online manner complying with the intention of decision-makers. This work seeks to design processes presenting SA for individual team members and SSA for the whole team. Therefore, an architecture has been proposed for SSA (S2A2) using multi-resolutional levels and an intelligent system model called Elementary Loop of Functioning (ELF) [12-14]. In the presented architecture, based on specific goals and objectives, decision-makers achieve SA regarding their decisions and SSA concerning their team goal.

Another contribution of this paper is to quantify the parameters affecting the coherence of SSA for resilience of the virtual organizations. For this purpose based on desired operational processes, a series of thresholds is specified by the designer as necessary conditions. Using a normative model named Garbage Can Model (GCM) [15], interactions between processes in S2A2 are modeled and simulated. Therefore, the cooperation and synchronization of the processes in the system are considered as threshold values.

The remainder of this paper is organized as follows. Section 2 explains the information fusion and SA models. Section 3 describes the intelligent system architecture based on multi-resolutional level representation. Section 4 proposes SSA architecture (S2A2). Section 5 illustrates Garbage Can Model and simulation results. Section 6 presents discussion and comparison. Section 7 demonstrates practical example to show the effectiveness and efficiency of the main results. Finally, Section 8 draws the conclusion.

2. **Information Fusion and SA Models.**

In order to obtain SA, data fusion techniques are used. Presenting flood of information can only be useful if they are fused successfully and in a timely manner. In general, all decision support systems make decisions using existing observations and knowledge, and make a better decision by synchronized exploitation of the knowledge. Therefore, such systems use data and information fusion. Different definitions have been presented for data fusion in different articles, but they generally refer to it as a synergistic combination of data at a certain time in a way that the information obtained from data fusion is thorough and can represent those aspects of the environment that are in question. Data obtained from data fusion is more accurate, precise, and robust both quantitatively and qualitatively compared to data obtained from a single source.

2.1. **Literature review on data fusion, information fusion and SA models.** Since 1987, there have been different models for data fusion, information fusion and SA including OODA [16,17], JDL [18-22], Intelligent Cycle [16,17], Waterfall [16,22], Dasarathy [22,23], Omnibus [16,22], Object-Oriented [17,23], Frankel-Bedworth [17,23], Extended OODA [24], λJDL [25], DFIG [3,26,27], combined CECA and JDL [28], STDF [3,29-31] and Salerno [3,27,32-34] models. Designers of data fusion models focus on finding an automatic method for object assessment, situation assessment, and impact assessment. However, the former process is at the center of their attention, and several technologies have been produced in this regard. With further development of models, especially from 2000 onwards, approaches for situation and impact assessment come to the forefront of model designs. Endsley proposed a model for SA [1,4,33,35,36] and based on Popper's thought, CECA was proposed for human key activities in decision making [37]. All the above models determine awareness and/or decision components and do not make any claims regarding architecture, system, software application or instantiations of physical values. This paper proposes a software architecture inspired by JDL, STDF, Endsley and CECA models. Furthermore, these models do not offer any solution for SSA.

In contrast with other models and architectures, a data fusion architecture is proposed in [38], focusing on perception level while providing no solution as to comprehension and projection level as well as their cognitive parameters. Lambert offered a framework for higher-level fusion systems based on STDF model [29-31], but he does not mention anything about providing SSA in addition to on-line planning to prepare the required SA. Although Salerno proposes a model and framework for resource management as the essential component for obtaining SA [36,34], the methods for obtaining SSA and also dynamic resource allocation are not explained. Furthermore, the DFIG model realizes control functions based on types of sensors and user's information need to supply action for achieving the mission goals; hence, it is of limited value in obtaining SSA [26]. In [39-41] an SSA architecture is proposed. This architecture addresses the way of consuming data, data dissemination time, and place of accessing data. Moreover, it focuses on the general process of real-time distributed information fusion in dynamic network environments for military purposes especially for fighters. However, they mentioned nothing in regard to SSA development for other network centric environments.

The above mentioned models have all failed to fully achieve our purpose owing to certain drawbacks. In previous models, SA and information fusion are obtained based on individual operators demand for decision making. These processes is accomplished independent of operators' perspectives and particular goals. These models mainly focus on data and information fusion processes the results of which regarding object, situation and impact assessment processes vary in details. To sum up, all previous works did not dynamically change the plan based on new events in the environment; neither did they describe the necessary conditions for effectiveness of obtaining data/information fusion, SA and SSA for all type of network centric environment.

2.2. **Endsleys model of SA.** According to Endsley's model of SA [1,4] (Figure 2), SA is considered at three levels of perception, comprehension and projection. At the first level or perception, objects and events, states and values are obtained through sensing, detecting and identifying processes. At the second level, comprehension, the meaning of the critical factors obtained at the level of perception, implications and the kind of situations are specified. The meaning of factors should be understood based on the operator's goals. Furthermore, interpretation and synthesis also occur at this level. Then, the third level projects future scenarios, possible outcomes and results. Prediction and simulation processes are performed at this level. Later, a fourth level, resolution level, was

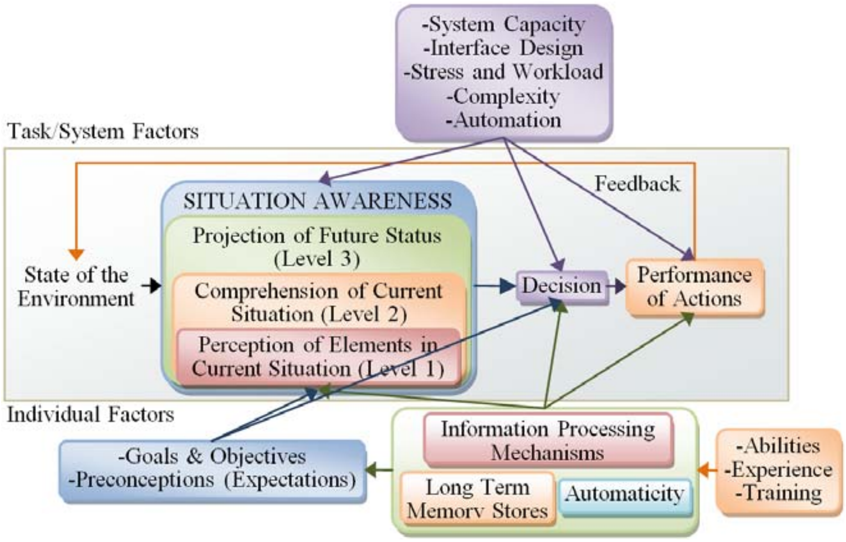

Figure 2. Endsley's model of situation awareness [1]

added to this model. Intention, the course of actions (COA) and cooperation, also occurs at this level, including planning and decision-making [35]. Resolution level provides the awareness required for the best result based on the current situation. The first three processes are considered from the internal perspective and the fourth from the external view. As seen in Figure 2, there are lots of parameters affect SA called in this paper as threshold values.

At the level of perception, we deal with the question of "what the current realities are?". The comprehension of the current reality answers the question of "what happens?" and the projection of state and characteristic answers the question of "what happens if?". The question of "what is to be done?" is dealt with at the level of resolution. In fact, at the resolution level, sense making occurs based on what SSA has achieved. Then, decision-making and planning can be done.

There are different factors involved in gaining SA. Capabilities of each team member, their interactions, and the environment in which interactions occur are important factors affecting shared SA. However, these criteria are themselves affected by such factors as geographic distribution, team leadership, coordination tools usage, network neighborhood, similar experiences, and the level of familiarity which in turn affect SSA [4]. In Endsley's model of SA, in addition to the environment, system factors and individual factors also affect SA, decision-making and the evaluation of the performance of actions.

The main approach of this paper is to plan an architecture for realizing SSA based on Endsley's cognitive model of SA. Therefore, the cognitive processes of Ensley's model are mapped on a multi-resolutional architecture. In this regard, the realization of all mentioned factors in the SA and SSA are defined and considered in the architecture as threshold values.

3. **Intelligent System Architecture Based on Multi-Resolutional Levels.** Presenting an architectural reference model is essential to the design and engineering of intelligent systems. The architectural reference model named Elementary Loop of Functioning (ELF) is used for designing and building an SSA system [12-14]. ELF model considers intelligence as a computational phenomenon and it is said that any system with the following four main processes is intelligent: sensory processing (SP) or perception, world modeling (WM), value judgment (VJ) and behavior generator (BG) is intelligent

[12]. The processes in ELF model specify sensor information, building, maintenance and application of knowledge-bases, goal selection, and response to sensor input and action control. The system architecture which is based on ELF, organizes these four processes the intelligence elements- by creating functional relationships and information flows.

SP includes functions responsible for focusing attention, identification and classification of characteristics, recognition of features, comparing of observations with expectations, objects and events detection, and situational analysis. WM includes the knowledge-bases about the world and builds and maintains events, entities, relationships and situations. In this process, predictions, expectations, belief and estimations of the probable results of future actions are also provided. Therefore, learning is formed in WM through storing planned behaviors and perceptions. The process of VJ estimates the costs, benefits, risks and expectations of the plan and assigns them to obtained objects, events, and situations. In other words, VJ determines the level of importance, punishment or reward, and degree of certainty of what has been provided by world modeling. Finally, BG chooses the goals and divides tasks for smaller departments. Furthermore, this process produces plans for achieving the goal and coordinates of the required actions accordingly, and also controls these actions [12].

Figure 3 illustrates ELF model [12-14]. The connection between the mentioned processes is formed in a way that a control loop with computational feedbacks characterized by bandwidth and latency is created. This internally closed loop is placed in the intelligent system from SP to WM, and from VJ to BG. Input and output in the internal part of the system are done by sensors and actuators, which are used as its external part. In organizations, sensors and actuators are individuals and groups. Therefore, outside the intelligent system, this loop is closed by actuation in the environment and sensing the environment. Goals enter this computational loop from above (BG) and sensor signals from below (SP). In an intelligent system, computational loops from sensors to action, from WM to SP and from BG to WM and then to VJ are constantly repeated. Based on sensor variety and action capability, these loops are repeated until the required thresholds of the tasks are achieved. In other words, repetition is terminated when information units in all sub-systems come to a consensus about entities, events, and situations, moreover, tasks for reaching the final goal are divided into specific tasks for command creation.

In the internal loop, sensed results are compared with final goals, and plans become regulated. Then, control is applied to the world so that the results of the plans are provided. In fact, intelligence is provided by the interaction between top down goals and bottom up feedbacks from sensors. Furthermore, there is an internal loop between SP and WM that provides predictions, compared with sensory observations. In this closed loop, estimation, predictive filtering or the placing of events stored in the short-term memory are done in a recursive manner. Besides, there is a connection between BG, VJ and WM that simulates and assesses elementary plans in the internal planning loop before the plan is chosen and carried out. In each loop, a knowledge-base is kept at a certain resolution

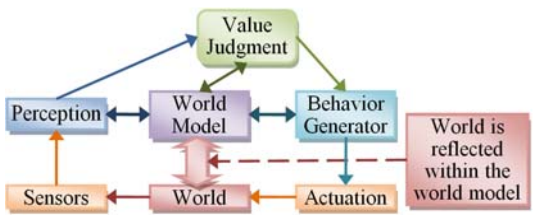

FIGURE 3. Elementary loop of functioning (ELF) [12]

and in a certain zone in the WM. At every level, plans are designed based on different plan horizons, and short-term memory follows sensory experiences stored in different historical intervals. Figure 4 illustrates the functional relationship between modules of ELF. The detailed explanation can be found in [12,42,43]. These processes are consistent with the models presented for data fusion and SA. That is why in S2A2, the Endsley's model and data fusion models can be mapped onto ELF.

In architectures with multi-resolutional levels, bandwidth control, resolution of the information, WMs and memories required for registering events are decreased by moving from lower to higher levels. The architecture will be assessed by threshold values based on high and low level restrictions. High level restrictions dictate priorities based on a compromise between the requests and processing elements, and low level restrictions assess measurement or storing ranges based on resolution levels. The appropriate choice of high and low restrictions of the required scope, plus satisfactory resolution for representation, in addition to proper assessment of the architecture's elements are the conditions based on which functional loops determine representation values based on application capabilities. Hierarchical analysis will be completed by creating a proper list of performance loops, determining actions and values as well as and specifying all the cause and effect connections provided in these loops.

Grouping (G), focusing attention or Filtering (F) and Searching (S) are the initial algorithm elements in planning and control, and provide different resolution levels based

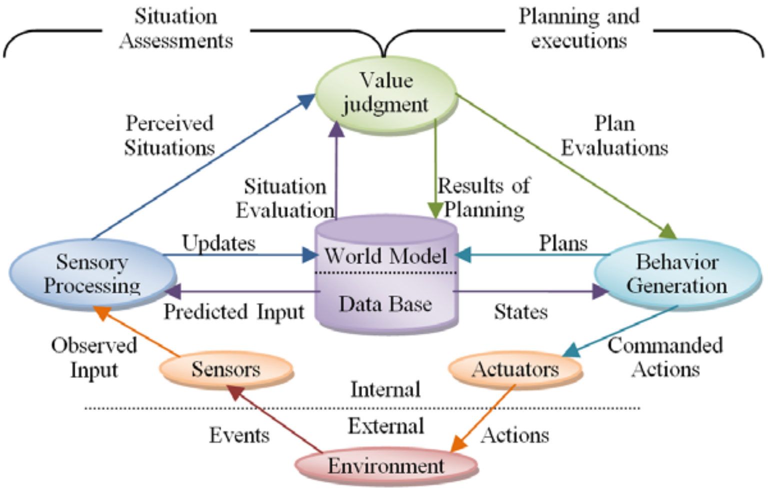

FIGURE 4. Functional relationships between modules of ELF [12]

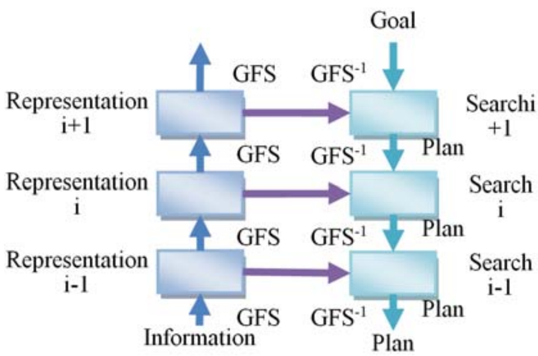

FIGURE 5. Relation between learning and planning [13]

on the mentioned thresholds. Figure 5 shows GFS for learning and GFS^{-1} for planning in architectures with levels [13,44]. The incoming information from the highest resolution level or the lowest level (i.e., level $i-1$) is aggregated and generalized by GFS. The results of this effort are stored in the world model and generalized in the next GFS phase with the lower resolution level (learning occurs). The generalization process continues until reaching the lowest or the highest resolution level. In the right column, by applying GFS^{-1} on the goal or the task at the lowest or higher resolution level, (i.e., level $i+1$), behavior is produced and sent to a higher stage. This process is continued until the action is produced at the highest resolution level. Therefore, using the GFS^{-1} and based on the scope of the operational process and planning horizons, tasks are decomposed and expanded by changing the levels from high to low. Using GFS and moving from low levels to high, SP integrates similarities and differences between observations and expectations and recognizes features, objects and relationships in the world. Based on the interrelation between planning and learning, any change in the environment will lead to the alteration of information and as a result plan changes for the new situation.

4. **Shared SA Architecture (S2A2).** S2A2 is based on ELF model in which the plan is provided from high to low levels. Furthermore, based on the human key activities' model for decision-making or CECA [37] in a layered manner, each level produces a behavior and gives it to lower levels until an action is produced in the environment at the lowest level (the level with the highest resolution). Moreover, SP carries out perception from the lowest level, and each perception goes to higher levels until it reaches the highest level with the lowest resolution and WMs are formed at each level based on aggregation processes presented in STDF [29-31] and JDL [3,18-21] models. In addition, at each level, SP, WM and BG processes are evaluated by VJ as well and as a result will reduce the processing complexity [12].

Most of the mentioned processes in SSA are mental processes; therefore, we need an architecture based on abstract levels of human mental processes. Based on these processes, four abstract levels are proposed for human data fusion: The sensory-motor calculation of differences, a categorical calculus of objects and relations, a modal calculation that combines concepts into deductive, inductive and abductive propositions that are falsifiable and finally, an ecological calculus of interacting systems [17]. Comparing these four levels including perception (object), comprehension (relationship comparison), projection (normative or normalizing level for concepts and the possibility of cancellations as a result of projection) and the ecological level or Resolution of interaction between systems with Ensley's SA model, they can be mapped onto perception, comprehension, projection and resolution levels respectively.

Depending on the type of process, human agent could be an individual or a team that has the final task of decision-making. This human agent gives his required information in a certain format and structure to the system under his supervision. Afterwards, the system gives back the results and the information required for final decision-making to agents. Figure 6 shows S2A2. In this figure, each decision-maker obtains SA at different resolution levels and shares his or her awareness with others based on the main goal resulting in SSA [45].

Shared VJ (SVJ) encompasses cognitive thresholds which are necessary for coherence in virtual organization's operational processes. These thresholds emanate from the culture of the organization and achievement regarding goal-directed task analysis (GDTA). In other words, shared VJ is the culture of the organization [46]. SA is achieved at the first three levels whereas SSA is formed at the fourth one.

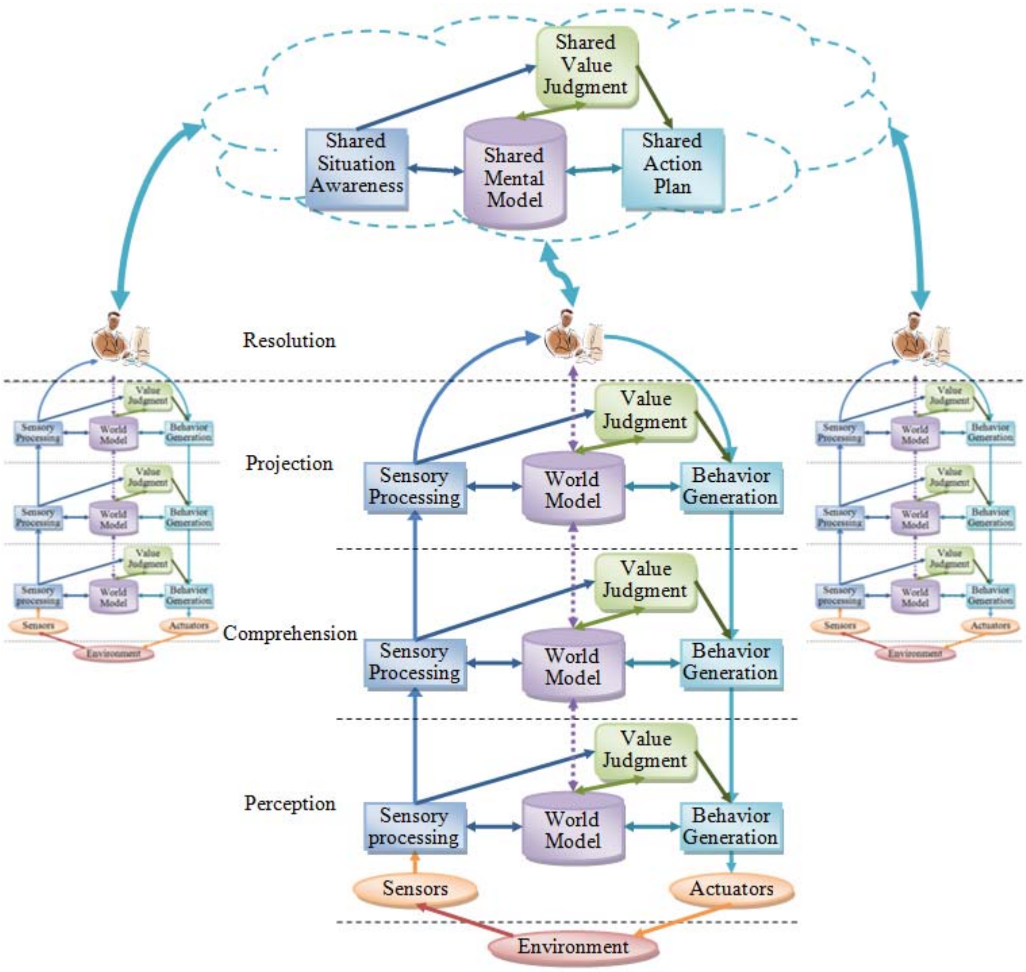

FIGURE 6. Shared situation awareness architecture (S2A2)

4.1. **Achieving SSA.** At resolution level, users share the obtained SA and update their mental models and consequently their SSA is formed. Figure 7 shows the functional relationships between modules at resolution level. Notice that SVJ and shared action planning in shared planning and execution stage of Figure 7 are performed based on the context aspects of the operational process which is deliberated in GDTA. Any operator has a role in virtual organizations in contrast to classical ones; moreover, any operator specifies its task for playing the related role. SVJ assigns cognitive threshold values, which are necessary for the coherence in shared action plan (SAP). These values are derived from the culture (i.e., trustworthiness, intention based trust and other cognitive parameters) of the virtual organization. Dividing the task into the sub-task based on organizations objectives is performed by behavior generator for shared action plan. When team members obtain shared mental model (SMM), they decide to perform shared actions to achieve the determined goal. Hence, shared action plan is transferred to the BG for issuing commanded shared actions to actuators.

As can be seen in Figure 7, SSA processing receives all obtained SAs from projection level of all SA systems in addition to the required SSA defined by SMM based on SAP. This process provides SSA based on perceived SAs and the required SSA, and transfers it to SVJ for evaluation. If SSA meets thresholds, the updated SMM will send the required SSA, parameters and thresholds to SSA processing.

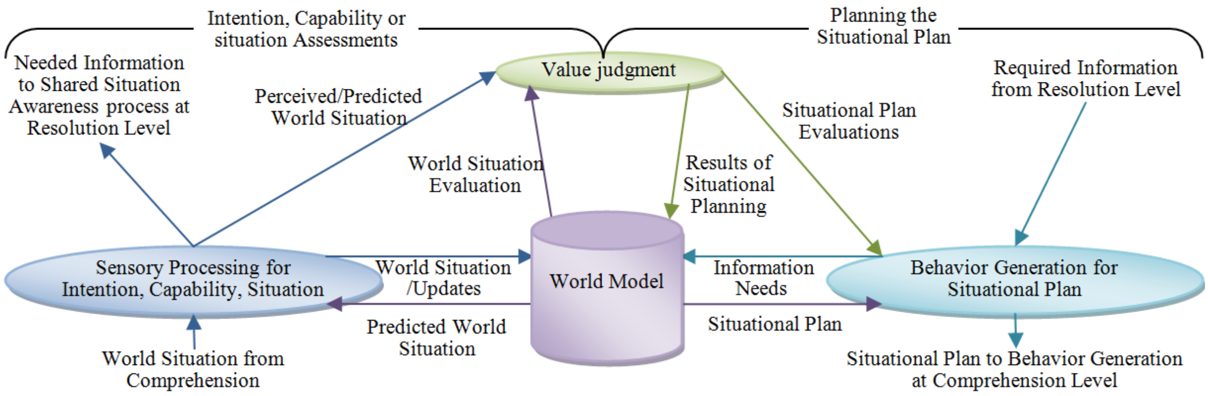

Figure 7. Functional relationships between modules at resolution level

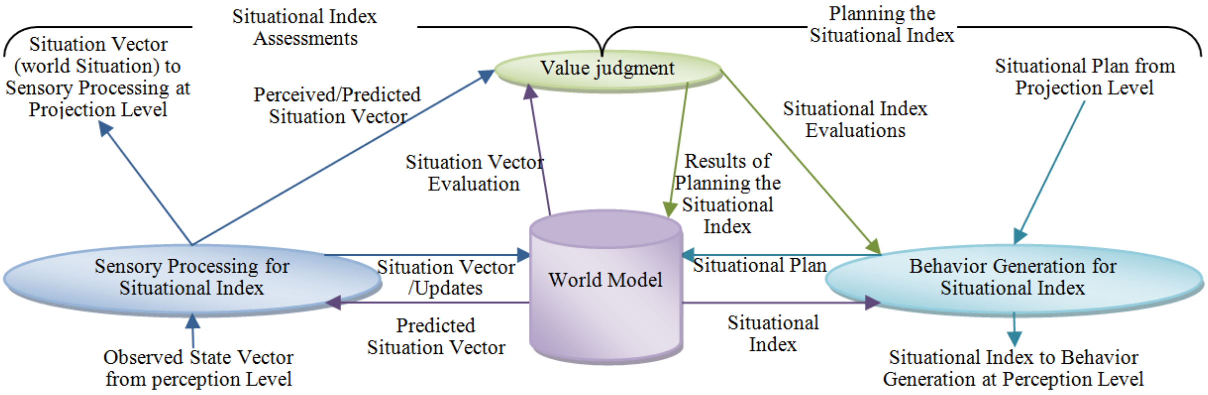

Figure 8. Functional relationships between modules in SSA processing

Figure 8 indicates the functional relationships between modules in SSA processing which is called SSA system. This process also provides required information based on operational process goal for BG process which users or team members enter to projection level of SA systems.

As Figure 8 illustrates, BG process receives the required SSA and sends them to shared WM (SWM) process as predicted information needs. SWM process retrieves the list of initial parameters and information such as the weighing coefficient required for environmental, team and individual factors, and also the degree of commitment or the ability to perform tasks by individuals, along with the benefit or value of exchanged information. It then assigns required information plan based on objectives of the operational process and required parameters. Afterward, SWM process transfers required information plan along with retrieved parameters as thresholds to the VJ process. In VJ, the cost, benefit and risk of the information plan are compared. If the result with thresholds is not met, the information plan returns to the BG to modify the information plan or even the required SSA. If the results meet the thresholds, constraints and preferences are assigned to required information plan and sent to the BG process. Cognitive or human agents as team members in the BG enter the information needs to the projection level of their own SA systems. They also set or update the threshold values and required knowledge and models in the WM processes of each layer in their SA system.

As seen in Figure 8, all SAs obtained from projection level of each operator's SA system enter to the SP of SSA processing. SP algorithms for SSA integrate all SAs based on common parts of all obtained SAs. Furthermore, the SP sends the perceived SSA to SWM. SWM generates the predicted SSA based on required information planning. Afterward, the perceived SSA and predicted results are transferred to VJ for evaluation. The VJ process computes costs, risks, and benefits, both of perceived SSA and predicted results based on planned activities and list of parameters such as the weighing coefficient required for environmental, team and individual factors. It also computes the probability of correctness and assigns belief and uncertainty parameters to the SSA. If the results of the obtained SSA meet the threshold, perceived SSA will be transferred to SVJ and SMM after updating SWM. Otherwise, the control is given back to the BG process to modify the required information plan.

To achieve the SSA, the VJ process should consider two perspectives for exchanged information. From the meaning perspective, the degree of commitment or the ability to perform tasks should be determined. Since intention equals multiplication of purpose by commitment [47], in the context aspect, the information exchanged between agents is considered. This information can be presented for doing something or updating information or it could be raw information transferred without any specific goals. However, in order to assign threshold values, the benefit or value of exchanged information should be considered [12]. From the agent's viewpoint that uses the received information from other agents, the benefit or value of information exchanged is almost equal with the value of received information multiplied by agent's capability to understand and act on, multiplied by the importance of action for reaching the goal [12]. The weights of these factors are assigned by theory of information entropy for better rationality and a higher practicability [48]. Therefore, an increase in value in each information unit can be a criterion for an increase in knowledge and thus bring about better SSA for achieving the goal.

4.2. **Toward SA.** In order to obtain SA, first the information required by a cognitive or human agent, including the intention of the operation, capability or SA, is fed into projection level whose task is to plan the required situation based on the goal and to clarify the difference between expected and present situations according to CECA decision-making model. Figure 9 shows the functional relationship between processes at projection level. In order to clarify the current situational plan, the BG process sends the received required information to the WM process. The WM process retrieves the course of events (COE) for possible courses of action (COA) based on assumed course of intents (COI) for agents; moreover, WM retrieves capabilities of the agent in regard to the defined operation. Then, the required situations and priorities of COAs and COEs are assigned for the expected situation. After assignment, the WM process assigns a new situation based on assumed

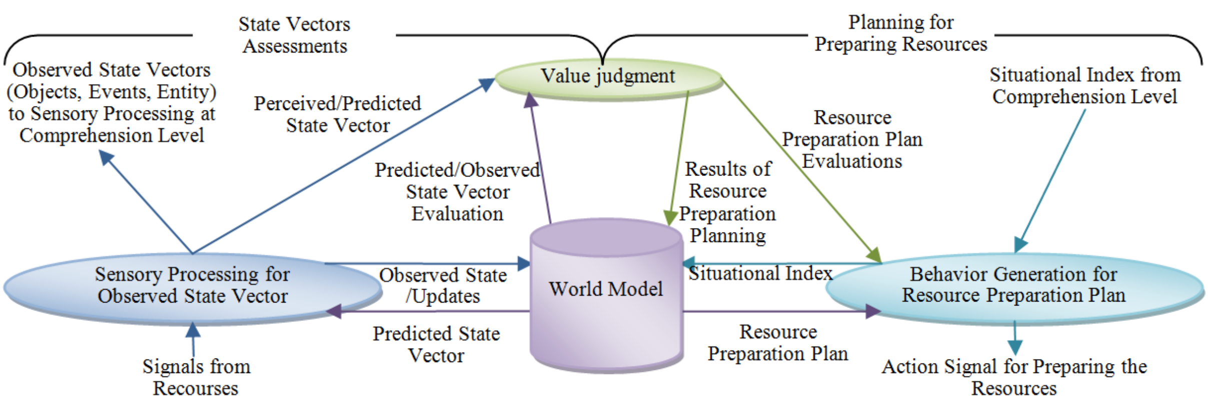

Figure 9. Functional relationships between processes at projection level

situations. Then, the new situation is sent to the VJ along with threshold value parameters as threshold vectors for comparison. In the VJ, the cost, benefit and risk of the new situation are calculated, and the situational plan is formed after considering constraints and preferences. Afterwards, the VJ compares the obtained situational plan with the required situation, and if it is higher than the threshold, constraints and preferences are assigned to the situational plan and transferred to the WM process. The WM stores the obtained situational plan and sent it to the BG for transfer to comprehension level. If threshold values are not met in this comparison, the plan is sent to the BG at projection level so that values are revised to meet the threshold values or the user is asked to set new threshold values.

As can seen in Figure 9, the SP receives the world situation from comprehension level and carries out semantic registration, converts the received situation vector to a certain semantic format, and provides a predicate expression of the world situation. Each agent can carry out an action in the environment and affects the environment based on its intention, capabilities and awareness. Therefore, COE, COI and COA should be determined for every agent in order to estimate the assumed COE and possible COA. This estimation is done in predictive assessment module. As a result, the SP sends the world situation to the WM at projection level. The WM retrieves the previous situation vector from the knowledge base and estimates possible COEs for possible COAs. In other words, the WM process carries out predictive assessment or impact estimation. Furthermore, expected and related COAs for the expected future situations are identified in the WM and stored along with the current situational vector. In the VJ, the current situation vector is compared with one or more previous expected situation vectors, and if they are the same, previous situation vectors are updated from the current situation. If the state of the current situation vector is different from the previous state, the new situation vectors that will begin from the current situation are initiated, and impact assessment is repeated based on the present situation vector. Finally, the VJ assigns belief and uncertainty parameters to the obtained impact or situation vectors and compares them with the threshold. If the obtained impact does not meet the threshold, VJ changes the situation plan, and control is given back to the BG, unless the model is updated and the information required by the cognitive or human agent, including intention, capability or SA are given to the them at the sensory processing for SSA at the resolution level through sensory processing at the projection level.

Figure 10 indicates the functional relationship between processes at comprehension level. The BG process at this level receives the obtained situational plan from projection level and sends it to the WM process. It then assigns the current situation based on the retrieved general understanding of the operation and the received situational plan. After assignment, situational indexes including objects, entities, and events related to the situation and the relationships between them at a certain time and place are determined and sent to the VJ along with the desired threshold vector. The VJ process determines the priority, cost and risk of situational indexes and weighs them. After comparison with the desired threshold and meeting the threshold values, constraints and preferences are assigned to situational indexes, and they are then sent to the WM process. The WM stores the obtained situational index and sends it to the BG for transfer to perception level for preparing resources. If situational indexes do not reach the specified threshold, the operation is sent back to the BG for revision of the situational index or the user is asked to set new threshold values.

As Figure 10 reveals, the SP receives the observes state vector from perception level and carries out semantic registration on observed state vectors, converts them to objects with specific semantic formats and finally provides predicate expressions for world observations

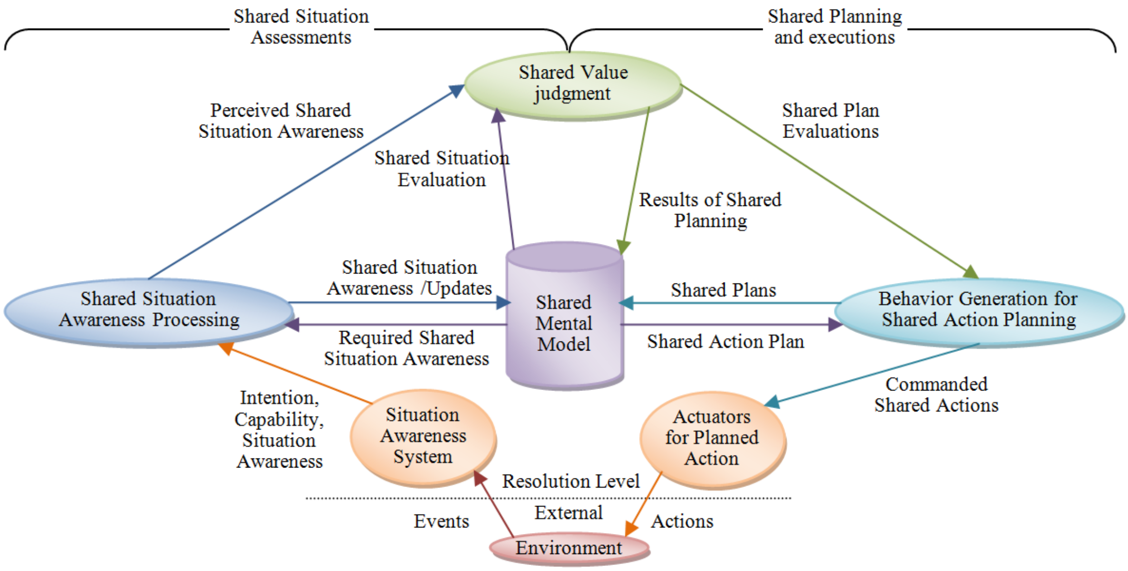

FIGURE 10. Functional relationships between processes at comprehension level

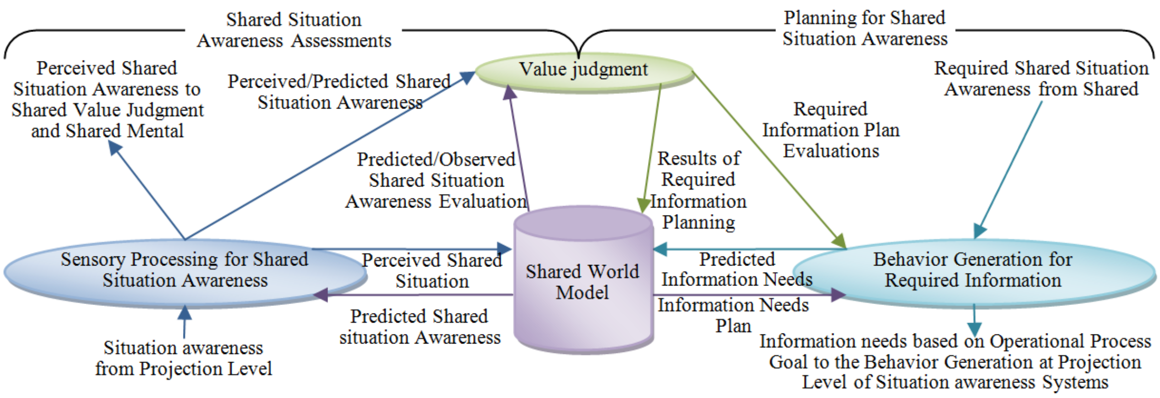

FIGURE 11. Functional relationships between processes at perception level

or situation vector and gives it to the WM. The WM process retrieves the previous situation, predicts and estimates the current situation based on predictive assessment [31], and stores predicted and expected predicates through the process of expectations. Then, the predicted and estimated predicates are sent to the VJ process for comparison. During the VJ process, if the observed predicate is consistent with the estimated predicate, the new situation vector is produced and if the observed predicate is not consistent with one or more expected predicates, the new situation vector is updated. Finally, the VJ assigns the parameters of belief and uncertainty to the new or updated situation vector. If the obtained situation vector meets the threshold, after updating during WM, the new situation vector is sent from the SP at comprehension level to the SP at projection level as the world situation. Otherwise, the VJ should change situational indexes based on the required situational plan.

Figure 11 shows the functional relationship between processes at perception level. The BG process receives situational indexes and sends them to the WM for planning and preparing resources. In the WM at perception level, based on constraints and knowledge from resources, assignment is done so that the measure required for observing situational indexes are determined. Then, in the VJ, resources are assessed and determined and actions required for specified resources are determined based on spatial and temporal constraints, geographical constraints and risk. Next, based on what has been obtained in the VJ, if resource preparation based on the required threshold is done successfully, based on determined commands by value judgment, BG process sends action signals to actuators. If resource preparation does not reach the specified threshold, the operation is sent back to the BG for revision of the resource preparation or the user is asked to set new threshold values. Actuators activate resources based on the received action signal so that these resources send the responses to the SP at perception level.

As can be seen in Figure 11, the SP receives signals and processes them, carries out registration and data alignment, converts them to a certain spatial-temporal format or observed state vector, and finally sends the observed state vector to the WM process. The WM retrieves the previously stored simulated states, predicts and estimates the current state and sends estimated and observed states to the VJ process. In the VJ, if predicted and observed states are identical, the predicted state is updated, otherwise a new state is initiated. The VJ assigns parameters of belief and uncertainty to obtained states as an object, event or entity, and if it meets the threshold vector, the model is updated in the WM, and the observed state vectors are transferred to the SP at comprehension level. Otherwise, the required resources should be determined and prepared again in the BG at perception level based on situational indexes.

In this architecture, WM processes at different levels interact with each other in order to update models required by other processes at each level. Furthermore, data mining process analyzes dynamic community of human observes and archived resources and provides new models at different levels or updates existing models.

5. **The Model Required by S2A2.** Though the major contribution of this work is representing a software architecture for developing various SSA systems required for virtual organizations, it is important to right the necessary conditions for the effectiveness of this architecture (i.e., coherence, resilience).

In S2A2, based on the operational process, a number of threshold values, constraints, and preferences should be determined for every level based on existing computational models and algorithms. Constraints dictate innate limitations to each level of abstract, and preferences specify preference characteristics of the system. Therefore, it is important to assign these values to the situation required for decision-making compatible with the goal. If these preferences and constraints are not well-defined, the system will not achieve the intended goal [15,49]. Furthermore, the trial-and-error actions should be avoided as far as possible. Since the cooperation of agents in different time periods and their level of activity in different areas are different, there will be an increase in uncertainty and changes. Due to agents not being able to carry out a proper action at a certain time, such systems will face problems or breakdown, so interactions should be designed in such a way as to satisfy performance criteria according to time sensitive changes. The bottlenecks in cooperation and synchronization of the system processes in question will be assessed as threshold values. Garbage Can Model (GCM) [15] is used for determining threshold values to quantify parameters affecting the coherence of SSA in virtual organizations.

5.1. **Garbage Can Model (GCM).** This model has been introduced in organizational theory for describing disorganization in organizational behaviors. Organizational processes are pre-defined tools for solving problems. These processes include procedure sets that should be used in dealing with problems. In other words, this model is used for simulating systems in which various participants, unclear decision-making technology and problem setting exist [50,51]. Since in S2A2, mentioned processes are carried out by human or computational agents, GCM is suitable for analyzing interactions among processes in S2A2.

This model defines the organization with four interactive elements known as Participants, Opportunities, Solutions and Problems. Participants are decision-makers in the organization and use Opportunities for problem solving. Problem solving capability and problem complexity affect energy values for Problems and Participants. Solutions are dependent and are multiplied by the energy of all Participants. In fact, this model introduces four independent classes of agents (Participants, Opportunities, Solutions, and Problems) with simple interaction rules.

In order to understand organizational processes, Opportunity is seen as a garbage can in which Participants throw different Problems and Solutions, as soon as they are produced. Problems, Solutions and Participants are independent of each other and may be added to or reduced from in different phases of decision-making regardless of time. Solutions use current options for problem solving instead of looking for new ones. In dealing with environments with high levels of uncertainty, decision-makers create a coherent image based on reshaping of their previous experience in order to use for new states' solutions that worked well previously.

In GCM, there are three methods for decision-making: Resolution, Oversight, and Flight. Decisions by Resolution are made when Participants are capable enough or when they are provided with enough Solutions or when Problems they are solving are simple enough. In other words, in this type of decision, Problems are actually solved. Oversight decisions are those made without considering existing Problems, and they solve no Problem in the organization. Flight decisions are not decisions per se, but they are Flights from one Problem and connecting it to another Opportunity so that it will be solved by others. Therefore, Flights happen when Resolution or Oversight decisions are impossible.

In this model, Participants are characterized based on energy or decision-making capability. Solutions have different levels of effectiveness, and Problems are determined by energy, which shows their level of difficulty. If Participants, Opportunities and Solutions are present with no Problem, the Oversight decision will be made. If all four elements are presented and if energy or Participants' ability, which is weighted based on Solution efficiency, is more than energy or difficulty of Problems, Resolution decision will be made; otherwise decision-making will be blocked until the required conditions are prepared. In other words, Participants reach to Solution or Problems are sent to another Opportunity (Flight). After the occurrence of Flight, energy or difficulty of remaining Problems may be too much in which case decisions are blocked or remaining Problems can be difficult enough in which case Resolution decision is made. This process continues until no more Problems are left or Flight decisions are made.

In order to simulate S2A2, cognitive or computational agents which perform processes at each level of resolution are considered as Participants. The chain of decision-making activities for SA or what is done in VJ and WM as the Opportunity and agents' requests or what BG process receives from higher level and sends to lower level or what the SP process should provide at each level are considered as Problems. Solutions are also present in WM as algorithms. The simulation will be done by determining different parameters for Participants such as Problems, Solutions, Opportunities, their stopping time, decision and access structure, energy distribution of Participants and Problems, and Solutions' efficiency as inputs for the simulator.

5.2. **Simulation results of S2A2 based on GCM.** Concerning GCM, NetLogo as a platform for simulating multi agent systems, is used. It provides an environment with programming capabilities for simulation of social and natural phenomena. Models implemented using NetLogo are described as complex time-variable systems. The modeler in NetLogo issues needed commands to different running agents which are independent of each other.

In our modeling with GCM, all Participants, Opportunities, Solutions and Problems move randomly. Based on the services presented in S2A2, the number of Participants, Opportunities, Problems and Solutions has been determined. Furthermore, the energy needed for solving Problems and Participants' energy have been normalized and efficiency of problem solving has been considered between 0.1 and 1. According to multi-agent modeling [51], initial values have been multiplied by 10. Therefore, the initial number of Participants based on the number of processes in all levels of resolution will be considered 150 as a constant. Based on the number of tasks in VJ and WM, initial number of Opportunities is adjusted as 150. Since the Opportunity leaves after decision-making, entry rate of 4 per step is selected so that there will be enough Opportunities left in the environment for solving remaining Problems. Furthermore, initial number of Solutions has been adjusted as 400 with an entry rate of five Solutions per step based on the number of algorithms and methods that perform the appropriate task. As interactions between levels are performed using BG and SP processes and there is no interaction at the first time, initial number of Problems has been considered zero. Based on tasks in those processes, an entry rate of four Problems per step and nine Problems per 10 steps is selected. Moreover, no Problem will enter after 100 steps. The results of modeling after 300 steps show different parameters based on problem solving efficiency, which is interpreted as variations of threshold value. Since all agents move randomly in the environment, the dynamics of the environment has been considered in the model. Furthermore, learning has also been considered with the entry rate of Solutions in the simulated model.

Figure 12 shows the time analysis of GCM based on time thresholds of S2A2. In this figure, different durations such as problem latency or time period of activity of the Problem before being given to the Opportunity and problem binding or time period of Problems being unsolved, are shown based on variation of efficiency. Moreover, this figure indicates the Participant binding or duration of the Participant unable to decide in the Opportunity due to lacking enough energy, and waiting time or the time spent by Opportunities in an organization before the decision is made.

As seen from Figure 12, when the Problem is active before being given to the Opportunity, there is no significant different with an increase in efficiency. Therefore, efficiency increase does not affect the assignment of Problems to Opportunities much. The reason for the Problem not getting solved is that it does not have enough energy to be solved, so it is blocked in the Opportunity, or maybe it has come from another Opportunity

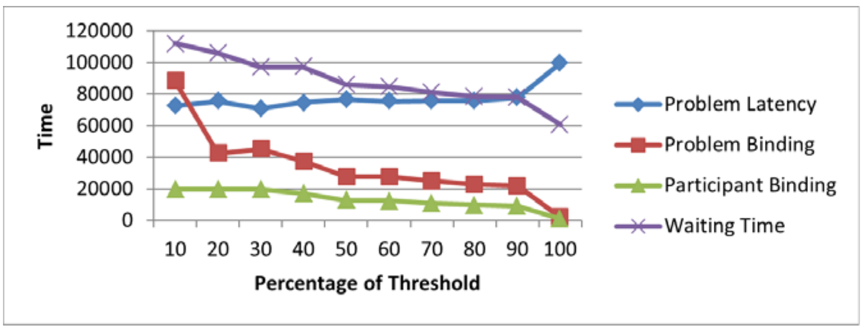

Figure 12. Time analysis of GCM

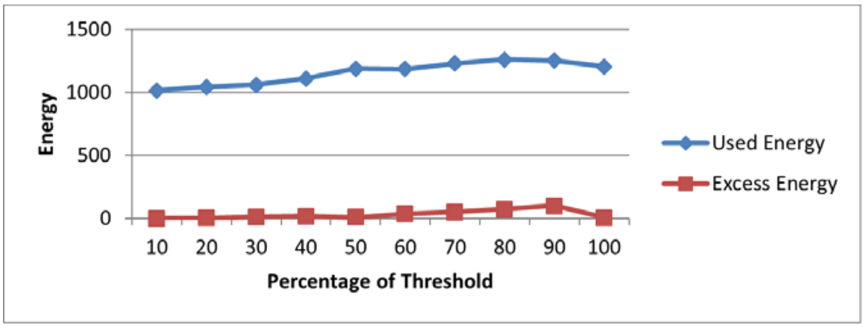

FIGURE 13. Used and excess energy for decision-making

(Flight). It can clearly be seen that an increase in efficiency decreases the duration of the Problem staying unsolved; in 50% efficiency there is a one third decrease and in 90% one fourth and in the highest efficiency it gets close to zero. Duration of the Participant not being able to decide in the opportunity due to lacking enough energy will also decrease with an increase in efficiency; in 50% efficiency to two thirds, in 80% efficiency to half and in the highest efficiency, it decreases to 5%. Finally, the time spent by Opportunities in an organization before the decision is made decreases with an increase in efficiency; in 50% efficiency to one fourth, in 80% efficiency to one third and in perfect efficiency, it decreases to half.

Analysis of the problem binding diagram shows that in S2A2, the time duration needed for the provision of information at each level is decreased one third by planning algorithms in BG and SP algorithms with 50% efficiency and one fourth with 80% to 90% efficiency as the threshold values of SP and BG processes. As planning and SP algorithms execute separately in each level, the problem latency diagram shows that there is no significant different with increasing efficiency before performing the algorithms. The participant binding diagram indicates that computational or cognitive agents in each level should have minimum thresholds of computational power and problem solving capability, which must be more than 50% based on complexity of the computational algorithms and processing load. Based on waiting time diagram, it can be said that VJ parameters for evaluating as threshold values along with WM should assign more than 50% in order to make better use out of the provided models and have faster VJ.

Figure 13 illustrates the diagram of the used energy or the energy spent by Participants for resolution and oversight decisions and excess energy or the total energy affecting the Resolution decision relative to efficiency increase. As it can be seen, the used energy and the excess energy increase with an increase in efficiency. There is an increase in 50% efficiency in both used energy and excess energy, and efficiency in 80% and 90% take the used energy and excess energy to its maximum respectively.

In S2A2, used energy represents the total amount of effective problem solving capabilities, which are performed by computational or human agents in the WM and the VJ processes. So the thresholds of problem solving capability and computational power of agents in these processes must be more than 50%. Furthermore, excess energy denotes the total effective algorithms in SP and BG processes. Hence, the threshold of effectiveness of these algorithms must be more than 50%.

Figure 14 shows total Flights or the number of times a problem goes from one Opportunity to another, and Participants' jumps or the number of times the Participant has exited the Opportunity and gone to the another Opportunity after spending some time in the first Opportunity based on variations of efficiency. As it can be seen, the number of Flights decreases with an increase in efficiency; in 50% efficiency to one third, in 80% and

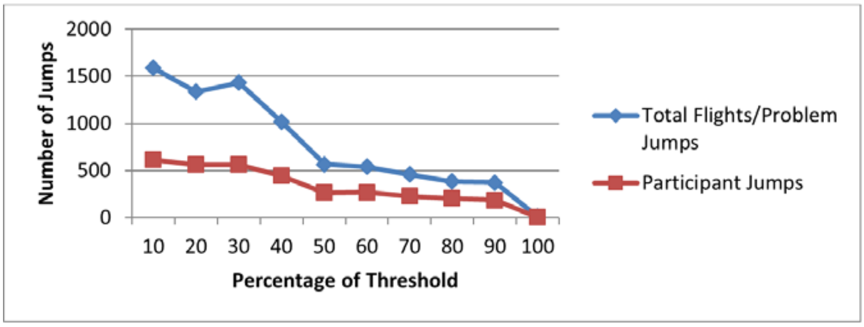

FIGURE 14. Total flights or problem jumps and participant jumps

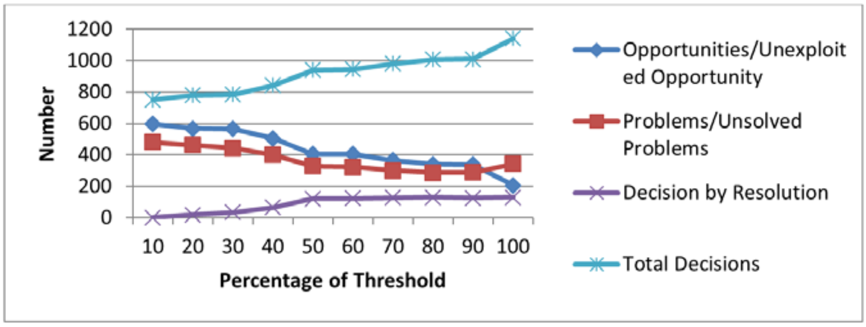

FIGURE 15. Number of unexploited opportunities, unsolved problems, resolution decisions, and total decisions

90% to one fourth and in 100% efficiency it decreases to less than one percent. Furthermore, the number of times the Participant exit the Opportunity to enter another one for solving problems after spending some time in the first one decreases with an increase in efficiency; in 50% efficiency it reaches less than half, in 80% it decreases to one third and in the highest efficiency, it decreases to one percent. The similar trend in the Participant and the Problem jump diagrams, indicates that computational power and problem solving capabilities of the agents and also the algorithms used in SP and BG, must be more than 50%.

Figure 15 shows the number of unexploited Opportunities, unsolved Problems, Resolution decisions, and all the made decisions at the end of simulation. It can be clearly seen that the number of remaining Opportunities in 50% efficiency is decreased to two thirds and in 80% and 90% to half and in perfect efficiency it decreases to one third. The number of remaining unsolved Problems is decreased with an increase in efficiency, and it reaches the minimum in efficiency higher than 50%. Furthermore, the number of Resolution decisions is increased with an increase in efficiency; as it can be seen, in 50% efficiency, there will be a significant increase in the number of the decisions by Resolution and for higher efficiencies, it remains almost constant. Finally, the number of all the decisions made whether the Resolution or the Oversight is increased with an increase in efficiency; in 50% efficiency it increases to 25%, in 80% and 90% efficiency to 35% and in 100% efficiency to at most 50%.

Figure 16 shows the percentage of the decisions made at the end of simulation. The percentage of Oversight decisions made are decreased with an increase in efficiency; in 50% efficiency there is a significant decrease in the number of these decisions and in higher efficiency levels it remains almost constant. In 50% efficiency the Percentage of Resolution decisions made is increased significantly and in higher efficiency levels it remains almost

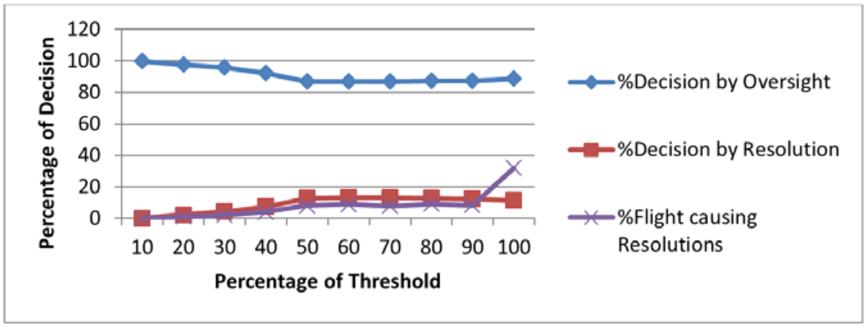

FIGURE 16. Percentage of the decisions made

constant. The percentage of Flights leading to Resolution decisions is increased with an increase in efficiency. As it can be seen, in the 50% efficiency this number is significantly increased and after that it remains constant and finally in the highest efficiency, there will be another increase.

Analyzing diagrams in Figure 15 and Figure 16, reveal that the algorithms and capabilities in four main processes of SP, WM, VJ and BG in each level of S2A2 must be more than 50% efficiency. Generally speaking, the results obtained from GCM show that the results are closer to the optimal level in thresholds over 50%. This threshold value states that fusion of sensors whose probability of correct detection is less than 50% yield worse results compared with the results of those sensors in isolation. Furthermore, in 100% efficiency there are leaps in the diagram. These leaps are caused by complete solutions and complete capabilities for solving problems. However, in real environments, there is no golden algorithm for solving the problem completely or there may even be no complete architecture for the organization [52].

6. **Discussion and Comparison.** Most papers are concentrating on data fusion, SA, software architecture, intelligent system architectures. However, few have been considering SSA realization in complex environments with dynamic situation and nonlinear behavior, in which non heterogeneous agents should cooperate to solve their problems efficiently and effectively. That is why S2A2 is proposed, which is a cognitive software architecture. It supports four tenets of complex systems [53]: distributive, decentralized, dynamics and diversity. Hierarchical levels with different resolutions support distributiveness. Cooperation of different agents based on their capability, supports decentralization. Dynamics are guaranteed by considering changes in system inputs. Finally, considering threshold values of the requirements of S2A2 and the relationship between heterogeneous agents guarantee diversity.

In S2A2, achieving the required information are broken into tasks using GFS-1 and mapped into general SSA internal processes. These processes are decomposed by designers using GDTA, and each task is determined from different perspectives beforehand or online (Figure 7). These tasks are based on the four processes in ELF model, and may be carried out by one or more agents [54-58]. In other words, it is necessary to determine who should carry out the tasks and how they are defined for agents. In hierarchical structures, these agents develop in a multi-scale relational structure [12]. In these organizations, agents appear in a tree-like manner with some connections between nodes of the tree at the same level. Based on the multi-scale relational structure, the cooperation between agents at a level provides the requirements of the agents at the higher level with lower resolution. Furthermore, agents of the same level can share their required data. These agents vary from simple reactive agents at high resolutions to cognitive agents or even humans at higher levels with lower resolutions. We can use any multi-agent architecture to facilitate the integration of agents, distributed services and applications to optimize S2A2 like the architecture presented in [59] for each level of abstract. In S2A2, the mentioned processes are done automatically as far as possible but under human supervision [60,61].

One of the main aspects of S2A2 is obtaining awareness in different contexts to continuing operations in very dynamic environments. There are different parameters such as belief and even scenarios based on the context of the operational process, stored in the WMs knowledge-bases, adjusted by human agents at different levels. The human agent gives his or her required information together with required threshold values to the SA system to obtain the SSA. These threshold values are determined in GDTA and are based on the culture, including individuals, team and environmental parameters. Thresholds are adjusted based on the context of the operational process and are stored in the WMs knowledge-base of each level based on the level of resolution. This system receives parameters in an online manner from human agents, and learning occurs as the system updates WM using GFS. Since all the process is not designed from the beginning and the human agent can modify system parameters while the process continues, this system sustains its operation in very dynamic environments.

Another aspect of this architecture is to achieve SSA not only at the resolution level but also all other three levels. For example, in some operations, users may only need to share the information at the perception level. Therefore, two other cognitive processes (i.e., the comprehension and the projection) are carried out by human.

One of the main questions which architectures must answer is computational modeling. Various computational modeling can be used for S2A2. For this purpose, in the proposed architecture, SP, WM, BG and VJ processes are decomposed to functions which perform specific existing algorithms and models. Receiving and organizing, clustering, generalizing, pre-recognition and pre-estimation computational models are useful for SP [18,30,62-64]. WM can make use of entity relational model formation, Knowledge base maintenance, estimation, focusing attention, combinational search, comparison and forecasting methods [26,65-67]. BG processes are implemented by task decomposition, scheduling, forecasting, comparison, selection and execution methods [14,37,63]. Finally, the VJ process encounters the uncertainty problem when performing object, situation and impact assessment. Computational modeling including Bayesian model, Dempster-Shafer theory, Dezert-Smarandache theory, fuzzy logic, intelligent neural networks, semantic information fusion, abductive reasoning, Ordered Weighted Average (OWA), and evolutionary algorithms are suitable models for solving these kinds of problems [23,68-73].

VJ as a task controller is the main component of the proposed S2A2 and instantiates a series of threshold values for computational models at each level for operational processes coherence. Human agents or BG processes at higher levels define these threshold values as the necessary conditions for the processes at lower levels. Migrating from each level to another depends on VJ or thresholds. If VJ or threshold as a task controller does not allow this to happen, we need to stay at that level and loop between SP and BG. It is why we are concentrating on modeling and simulating S2A2.

Simulating results based on GCM indicate that more effective algorithms for the mentioned processes decrease the time needed to provide information. Furthermore, higher threshold requires more processing power. Therefore, VJ parameters such as risk and cost-benefit in planning and belief and uncertainty parameters in situation assessment should assign more than 50% to themselves in order to make the best use of the provided models and have faster VJ. Difficulty analysis of the BG and the SP reveals that factor of the threshold related to high and low limitations at each resolution level should be at least 50%, so that the architecture becomes convergent and provides the SSA required for achieving the goal. Accordingly, computational agents at each resolution level should have computational power of over 50%; higher computational power leads to shorter processing time. However, it should be noted that increased problem solving capability could contribute to the complexity of the computational algorithm and processing load.

Four processes of ELF model are performed at three low levels by computational agents and controlled by human. Therefore, when there is not enough processing capability in machine or when there is a lack of proper algorithms based on defined thresholds for agents implementation, human agents are stranded in for ELF processes. As a result, human interference causes the SA system to update itself in unpredicted states. Furthermore, updating the models and modification of the threshold values are done online under human supervision or by human agent. However, threshold values should be applied to system stability condition, which is determined at the resolution level. These conditions are discussed in detail in the previous section.

It should be noted that thresholds are determined in VJ processes at each level. These values are in the form of a vector of threshold values because all information units exchanged between levels are in the form of vectors and include qualitative parameters. These threshold values are determined in GDTA and are determined by external level users or from the perspective of the external system in environmental, team and individual factors. At the resolution level, information should be transferred in a virtual organization and there should be interaction between cognitive or human agents, based on defined thresholds of communication functions which are determined in the GDTA phase. It means that, in order to ensure this communication, there should be a meaning-based communication between human or cognitive agents in addition to communication protocols among computational agents in the context perspective [65,66].

Another important aspect of S2A2 is the knowledge required for interpreting the outputs of SPs and supporting BG activities. This knowledge, which is named a-priori knowledge, is stored in the knowledge-bases of WM processes or knowledge grid [65,66] as ontology [74-78]. As to the operational process, a-priori knowledge is demanded by the virtual organization, and obtained by knowledge discovery methods at the GDTA phase. The obtained knowledge is validated and entered into the knowledge-base by human operators at the resolution level of S2A2 [12,34,35,76]. A-priori knowledge consists of a list of all possible world situations and clearly stated the meaning of the information obtained from world situations. Therefore, the resources that describe the world situation should be specified. In addition to determining the mentioned knowledge, a list of possible tasks and methods of performing them should also be described and stored in knowledge-bases. Hence, the WM process helps model and predict SP and BG processes at each level. Finally, a list of actuators for all possible methods of achieving the defined goals should be determined so that the decision related to the actuators behavior for changing the world can be produced using the BG process. Table 1 reveals the required knowledge stored in existing databases in the WM by levels.

Threshold values for data fusion and SA in other models are defined based on precision and accuracy of sensors and other information resources. In fact, in SSA, humans are exchanging information and therefore, cognitive thresholds including trust and commitment are considered in Table 1 at the resolution level. Furthermore, S2A2 is concentrating on dynamic planning aspects in complex environments while most papers in this field rely on offline planning. For this purpose, according to the first three levels of Table 1, the knowledge of the operation process is instantiated. In fact, using the computational models and algorithms based on the defined thresholds, dynamic planning is done.

Table 1. Required knowledge stored in world model

level	Required Knowledge
Perception	Recourse required for situational indexes, priorities, preferences, default and threshold values, spatial and temporal constraints, geographical constraints, risk, previous states, belief and uncertainty parameters, expectations.
Comprehension	Situational indexes (objects, entities and events related to situation and relationship between them at the certain time and place), general understanding of operation, priorities, preferences, default and threshold values, previous situations, expected predicates, belief and uncertainty parameters.
Projection	Course of events, course of actions, course of intents, required situations, priorities, preferences, default and threshold values, set of previous situations, course of expected events, course of possible actions, future expected states.
Resolution	Weighting coefficient required for environmental, team and individual factors. Environmental factors include anxiety and stress, workload, exhaustion, team size, physical place, uncertainty and deception, automation and capabilities of the system, and complexities of the interface. Team factors include communications, cooperation tools, shared mental models, team processing, team dimensions, trust and commitment. Individual factors include mental models, memory, knowledge, cognitive sources, experience, training, perceptive and problem solving capabilities, individual skills, decision making skills, mental and physical conditions, beliefs, expectations, and perception abilities.

Finally, to realize S2A2, communication, negotiation and interoperability mechanisms of software components must be specified. Regarding the multi agent systems, a methodology based on Tropos is suggested [79-82]. In this methodology, all concepts including agents, goals and planning are considered in different phases of development from analysis until implementation.

7. **Practical Example.** To show different phases in S2A2, a Generic Cognitive Crisis Management (GCCM) system for disaster management is presented. The main objective of GCCM system is gathering information about rescue teams capabilities, the impact of the crisis, collecting relevant information about the crisis, resource allocation, performance evaluation of teams and finally planning the mission [83]. There is a main crisis management headquarter that assigns one or more roles to team members for obtaining specified goal or sub-goals based on the crisis at hand and GDTA. Moreover, thresholds for required SSA and information plan at resolution level, world situation and situational plan at projection level, situation vector and situational index at comprehension level, state vector resource preparation plan at perception level and finally thresholds for environmental, team and individual factors are defined by the headquarter. According to GCM simulating results, these thresholds should be more than 50% for obtaining required SSA for achieving the goal. Based on GDTA and operational process, required knowledge (Table 1) is elicited and stored in existing databases in WMs at each level. Each team member selects one or more operational zones based on assigned roles and acts in the zone based on obtained SSA. Individuals can have access to different sensors and information

resources. Headquarter and all team members can communicate and share the provided information.

Regarding to our proposed architecture, the crisis management headquarter determines roles and cognitive thresholds for each team member based on GDTA and specified goals. Members in teams play roles such as SVJ, SMM and BG for SAP processes shown in Figure 7. They also have capabilities to perform desired action based on issued commands. Each team member uses an SSA System which provides SSA and SAs related to the specified operational zone. SAs are provided by SA systems.

To explain S2A2 details, an example of a disaster management scenario is presented [83]. In this scenario based on a natural disaster such as earthquake, tasks and operational zones are assigned. Each member decides to act based on defined roles and sub-goals in the specific operational zone. The rescue members in operational zones are expected to achieve SMM to plan for the best shared action. Hence, required SSA based on specified goals has to be determined in the SMM process. The assigned SSA along with specific thresholds are fed to SSA system by each team member. It then provides information such as possibility of surviving citizens for performing shared action plans.

Based on Figure 8, required information that each team member has to receive from the environment, is determined based on required SSA that is assigned by team members as system operators. They obtain SSA to reach SMM by instantiating obtained thresholds from cognitive thresholds and required SSA. For instance, the decision as to which team can perform best in which zone.

SSA system provides each member required information plan independently and based on the required SSA (Figure 8). Information plan includes determining citizens who are waiting for rescue, capabilities of teams and probability of surviving which are performed in GDTA. If the information plan could satisfy the required thresholds based on team capabilities, it will be sent to projection level of relevant SA system to generate the situational plan or information plan in the specific operational zone (Figure 9). When situational plan is obtained, comprehension level generates situational indexes (Figure 10). Situational indexes in this system are people who are waiting for help, rescue teams and survived citizens, events related to the situation and relations between citizens and rescue teams in specific time and place. Therefore, information resources have to provide for identifying situational indexes. By determining required information resources in each situational index at perception level (Figure 11), team member receives relevant environmental data.

Receiving the environmental data, processes of perception level provide the state vectors of environmental data including victims statistics, rescue teams spots and other environmental data (Figure 11) and send them to comprehension level. Processes will then transform the state vectors and their relation into specific semantic format and form the situation vector based on generated situational indexes (Figure 10). At projection level (Figure 9), related processes based on provided situational vectors, estimate a set of possible actions based on current situations and provide world situation. In other words, each victims survival probability is checked by the rescue team. Providing SAs in each SA system, all team members share provided SAs with each other through the SSA system (Figure 8) then they can obtain SMM using received data and finally, perform the best shared action (Figure 7).

8. **Conclusion.** In virtual organizations as complex adaptive systems, team members should have SSA for performing shared action plans. As a result, cognitive parameters like individual, team and environmental factors, should be considered in a cognitive SSA system. The present article introduces a new cognitive architecture for developing various

SSA systems (S2A2) which concentrate on dynamic planning aspects in complex environments. For this reason, Endsley's cognitive model of SA is mapped on the ELF model processes with four levels named perception, comprehension, projection and resolution. Interaction of the main processes of ELF named sensory processing (SP), behavior generation (BG), world modeling (WM) and value judgment (VJ) at each level brings out the on-line planning and SSA.

Quantifying the parameters affecting the coherence of S2A2 for resilience of the virtual organizations is another aspect of this work. Value judgment is the main component of the proposed architecture that specifies series of thresholds, which emanate from the culture of the organization based on goal-directed task analysis (GDTA). These threshold values are based on desired operational processes of organization and specified as necessary conditions at each level. If the required threshold is not met, the information needed for reaching awareness and finally the required SSA for virtual organizations will not be provided. In order to determine and quantify these threshold values, a normative model named GCM is used, and dynamic interactions between S2A2 processes are modeled and simulated. Simulation results indicate that normalized threshold values must be greater than 50% efficiency.

Various computational modeling can be used for internal processes of S2A2. Therefore, existing models for problem solving are introduced and can be used for SP, BG, WM and VJ processes at each level, which are carried out by computational or cognitive agents even human. These models should satisfy required threshold values based on operational processes. Furthermore, a-priori knowledge is required and defined for interpreting and supporting the outputs of SP and BG activities. Updating the models and modification of the threshold values are performed under human supervision on-line. As a result, based on the defined thresholds, the computational models and algorithms are selected and the dynamic planning is achieved for SSA in the complex environment.

REFERENCES

[1] M. R. Endsley, Designing for situation awareness in complex systems, *Proc. of the 2nd International Workshop on Symbiosis of Humans, Artifacts and Environment*, Kyoto, Japan, 2001.

[2] D. A. Lambert, Situations for situation awareness, *Proc. of the 4th International Conference on Information Fusion*, Montreal, Canada, 2001.

[3] M. E. Liggins, D. L. Hall and J. Llinas, *Handbook of Multisensor Data Fusion: Theory and Practice*, 2nd Edition, CRC Press LLC, 2009.

[4] C. A. Bolstad, H. M. Cuevas, C. Gonzalez and M. Schneider, Modeling shared situation awareness, *Proc. of Behavior Representation in Modeling and Simulation (BRIMS)*, Los Angeles, CA, USA, 2005.

[5] D. Mendona, T. Jefferson and J. Harrald, Collaborative adhocracies and mix-and-match technologies in emergency management, *Communications of the ACM*, vol.50, no.3, pp.44, 2007.

[6] L. Militello, E. Patterson, L. Bowman and R. Wears, Information flow during crisis management: Challenges to coordination in the emergency operations center, *Journal of Cognition, Technology & Work*, vol.9, no.1, pp.25-31, 2007.

[7] M. Liu and L. Zhao, Optimization of the emergency materials distribution network with time windows in anti-bioterrorism system, *International Journal of Innovative Computing, Information and Control*, vol.5, no.11(A), pp.3615-3624, 2009.

[8] D. L. Hall, J. Llinas, M. McNeese and T. Mullen, A framework for dynamic hard/soft fusion, *Proc. of the 11th International Conference of Information Fusion*, Cologne, Germany, pp.85-92, 2008.

[9] A. Fetanat and M. N. Fesharaki, A trust model in sensmaking process, *International Journal of Computational Cognition*, vol.8, no.2, pp.1-4, 2010.

[10] E. Teimoury, M. Fesharaki and A. Bazyar, The relationship between mediated power asymmetry, relational risk perception, and governance mechanism in new product development relationships, *Journal of Research in Interactive Marketing*, vol.4, no.4, pp.296-315, 2010.

[11] E. Teimoury, M. Fesharaki and A. Bazyar, The relationship between modes of governance and relational tie in new product development relationships, *Journal of Strategy and Management*, vol.3, no.4, pp.373-392, 2010.
[12] A. M. Meystel and J. S. Albus, *Intelligent Systems: Architecture, Design and Control*, Wiley, New York, 2001.
[13] A. M. Meystel, Multiresolutional hierarchical decision support systems, *IEEE Transactions on Systems, Man and Cybernetics – Part C: Applications and Reviews*, vol.33, no.1, pp.86-101, 2003.
[14] W. Van Wezel, R. Jorna and A. Meystel, *Planning in Intelligent Systems: Aspects, Motivations, and Methods*, John Wiley & Sons Inc., Hoboken, New Jersey, 2006.
[15] M. D. Cohen, J. G. March and J. P. Olsen, A garbage can model of organizational choice, *Administrative Science Quarterly*, vol.17, no.1, pp.1-25, 1972.
[16] M. Bedworth and J. Obrien, The omnibus model: A new model of data fusion? *IEEE Aerospace and Electronic Systems Magazine*, vol.15, no.4, pp.30-36, 2000.
[17] C. B. Frankel and M. D. Bedworth, Control, estimation and abstraction in fusion architectures: Lessons from human information processing, *Proc. of the 3th International Conference on Information Fusion*, Paris, France, 2000.
[18] D. L. Hall and J. Llinas, An introduction to multi-sensor data fusion, *Proc. of the IEEE*, vol.85, no.1, pp.6-23, 1997.
[19] A. N. Steinberg, C. L. Bowman and F. E. White, Revisions to the JDL data fusion model, *Sensor Fusion: Architectures, Algorithms, and Applications, Proc. of the SPIE*, vol.3719, Orlando, FL, USA, 1999.
[20] J. Llinas, C. Bowman, G. Rogova, A. Steinberg, E. Waltz and F. White, Revisiting the JDL data fusion model II, *Proc. of the 7th International Conference on Information Fusion*, Stockholm, Sweden, pp.1218-1230, 2004.
[21] A. N. Steinberg and C. L. Bowman, Rethinking the JDL data fusion levels, *Proc. of in National Symposium on Sensor and Data Fusion*, Maryland, 2004.
[22] D. L. Hall and J. Llinas, *Handbook of Multi-sensor Data Fusion*, CRC Press LLC, 2001.
[23] E. F. Nakamura, A. A. F. Loureiro and A. C. Frery, Information fusion for wireless sensor networks: Methods, models, and classications, *ACM Computing Surveys (CSUR)*, vol.39, no.3, 2007.
[24] E. Shahbazian, D. E. Blodgett and P. Labb, The extended OODA model for data fusion systems, *Proc. of the 4th International Conference on Information Fusion*, Montreal, Canada, 2001.
[25] D. A. Lambert, Grand challenges of information fusion, *Proc. of the 6th International Conference on Information Fusion*, Cairns, Australia, pp.213-220, 2003.
[26] E. Blasch and S. Plano, DFIG level5 (user refinement) issues supporting situation assessment reasoning, *Proc. of the 8th International Conference on Information Fusion*, Philadelphia, PA, USA, 2005.
[27] E. Blasch, I. Kadar, K. Hintz, J. Beirmann, C. Ching, J. Salerno and S. Das, Resource management and its interaction with level 2/3 fusion from the fusion06 panel discussion, *Proc. of the 10th International Conference on Information Fusion*, Quebec Canada, pp.1-10, 2007.
[28] D. McMhchael and G. Jarrad, Grammatical methods for situation and threat analysis, *Proc. of the 5th International Conference on Information Fusion*, Philadelphia, PA, USA, pp.8, 2005.
[29] D. A. Lambert, A unification of sensor and higher-level fusion, *Proc. of the 9th International Conference on Information Fusion*, Felorence, Italy, pp.1-8, 2006.
[30] D. A. Lambert, STDF model based maritime situation assessment, *Proc. of the 10th International Conference on Information Fusion*, Quebec, Canada, pp.1-8, 2007.
[31] D. A. Lambert, A blueprint for higher-level fusion systems, *Journal of Information Fusion*, vol.10, no.1, pp.6-24, 2009.
[32] J. J. Salerno, Resource management: A necessary and integral component to any level 2/3 fusion capability, *Proc. of the 9th International Conference on Information Fusion*, Florence, Italy, 2006.
[33] J. Salerno, Where's level 2/3 fusion a look back over the past 10 years, *Panel Discussion of the 10th International Conference on Information Fusion*, Quebec, Canada, 2007.
[34] J. Salerno, Measuring situation assessment performance through the activities of interest score, *The 11th International Conference on Information Fusion*, Cologne, Germany, pp.326-333, 2008.
[35] J. Salerno, M. Hinman, D. Boulware and P. Bello, Information fusion for situational awareness, *Proc. of the 6th International Conference on Information Fusion*, Cairns, Australia, pp.507-513, 2003.
[36] J. Salerno, M. Hinman and D. Boulware, Building a framework for situation awareness, *Proc. of the 7th International Conference on Information Fusion*, Stockholm, Sweden, pp.219-226, 2004.

[37] D. J. Bryant, Modernizing our cognitive model, *Proc. of Command and Control Research and Technology Symposium*, San Diego, CA, USA, 2004.

[38] H. S. Carvalho, W. B. Heinzelman, A. L. Murphy and C. J. N. Coelho, A general data fusion architecture, *Proc. of the 6th International Conference of Information Fusion*, Cairns, Australia, pp.1465-1472, 2003.

[39] S. Jameson and C. Stoneking, Army aviation situational awareness through intelligent agent-based discovery, propagation, and fusion of information, *Proc. of the 58th American Helicopter Society International Annual Forum*, Montreal, pp.292-302, 2002.

[40] W. J. Farrell, S. Jameson and C. Stoneking, Shared situation awareness for army applications, *Proc. of 2003 National Symposium on Sensor and Data Fusion*, San Diego, CA, USA, 2003.

[41] S. Jameson, C. Stoneking, D. G. Cooper, P. Gerken, C. Garrett and A. Hughes, Data fusion for the Apache longbow: Implementation and experiences, *Proc. of the 61th American Helicopter Society International Annual Forum*, Grapevine, TX, USA, 2005.

[42] V. Teresius, Reference architecture of intelligent system, *Electronics and Electrical Engineering*, vol.63, no.7, pp.53-56, 2005.

[43] C. Schlenoff, R. Madhavan, J. Albus, E. Messina, T. Barbera and S. Balakirsky, Fusing disparate information within the 4D/RCS architecture, *Proc. of the 8th International Conference on Information Fusion*, Philadelphia, PA, USA, 2005.

[44] R. Sanz and A. Meystel, Modelling, self and consciousness: Further perspectives of AI research, *Workshop on Performance Metrics for Intelligent Systems*, Gaithersburg, USA, 2002.

[45] H. Parvar, M. N. Fesharaki and B. Moshiri, Shared situation awareness system architecture for network centric environment decision making, *Proc. of the 2nd International Conference on Computer and Network Technology*, Bangkok, pp.372-376, 2010.

[46] F. Capra, *The Hidden Connections: Integrating The Biological, Cognitive, And Social Dimensions Of Life Into A Science Of Sustainability*, Doubleday, New York, 2002.

[47] G. Schreiber, H. Akkermans, A. Anjewierden, R. de Hoog, N. Shadbolt, W. Van de Velde and B. Wielinga, *Knowledge Engineerins and Management: The CommonKADS Methodology*, The MIT Press, London, England, 2000.

[48] X. Li, F. Zhou and X. Yang, Developing dynamic P2P trust model using theory of entropy-based multi-source information fusion, *International Journal of Innovative Computing, Information and Control*, vol.7, no.2, pp.777-790, 2011.

[49] D. McMichael, G. Jarrad, S. Williams and M. Kennett, Modelling, simulating and estimation of situation histories, *Proc. of the 7th International Conference on Information Fusion*, Stockholm, Sweden, pp.928-935, 2004.

[50] T. Sawaragi and K. Murasawa, Simulating behaviors of human situation awareness under high workloads, *Artificial Intelligence in Engineering*, vol.15, no.4, pp.365-381, 2001.

[51] G. Fioretti and A. Lomi, An agent-based representation of the garbage can model of organizational choice, *Journal of Artificial Societies and Social Simulation*, vol.11, no.1, pp.1, 2008.

[52] D. L. Hall and A. K. Garga, Pitfalls in data fusion (and how to avoid them), *Proc. of the 2th International Conference on Information Fusion*, Sunnyvale, CA, USA, 1999.

[53] D. S. Albert and R. E. Hayes, *Power to the Edge: Command and Control in the Information Age*, CCRP, USA, 2003.

[54] D. Perugini, D. Lambert, L. Sterling and A. Pearce, Distributed information fusion agents, *Proc. of the 7th International Conference on Information Fusion*, Cairns, Australia, pp.86-93, 2003.

[55] M. G. Ceruti, Mobile agents in network-centric warfare, *Proc. of the 5th International Symposium on Autonomous Decentralized Systems*, Dallas, TX, USA, pp.243-246, 2001.

[56] T. Hanaratty, J. Dumer, J. Yen and X. Fan, Using agents with shared mental model to support network-centric warfare, *Proc. of the 7th Multiconference on Systemic, Cybernetics and Informatics*, OR, USA, 2003.

[57] J. Llinas, G. Pavlin, D. Snyder, A. Steinberg and K. Sycara, Agent based information fusion, *Panel Discussion of the 10th International Conference on Information Fusion*, Quebec, Canada, 2007.

[58] V. Gorodetsky, O. Karsaev, I. Kotenko and V. Samoilov, Multi-agent information fusion: Methodology, architecture and software tool for learning of object and situation assessment, *Proc. of the 7th International Conference on Information Fusion*, Stockholm, Sweden, pp.346-353, 2004.

[59] J. M. Corchado, D. I. Tapia and J. Bajo, A multi-agent architecture for distributed services and applications, *A Multi-Agent Architecture for Distributed Services and Applications*, vol.8, no.4, pp.2453-2476, 2012.

[60] G. Jakobson, L. Lewis and J. Buford, An approach to integrated cognitive fusion, *Proc. of the 7th International Conference on Information Fusion*, Stockholm, Sweden, pp.1210-1217, 2004.

[61] D. A. Lambert, Tradeoffs in the design of higher-level fusion systems, *Panel Discussion of the 10th International Conference on Information Fusion*, Quebec, Canada, pp.1-4, 2007.

[62] D. Smith and S. Singh, Approaches to multi-sensor data fusion in target tracking: A survey, *IEEE Transactions on Knowledge and Data Engineering*, vol.16, no.12, pp.1696-1710, 2006.

[63] D. Y. Kim, J. H. Yoon, M. Jeon and V. Shin, Distributed information fusion with intermittent observations for large-scale sensor networks, *International Journal of Innovative Computing, Information and Control*, vol.7, no.11, pp.6437-6451, 2011.

[64] S. Wan, Applying interval-value vague set for multi-sensor target recognition, *International Journal of Innovative Computing, Information and Control*, vol.7, no.2, pp.955-963, 2011.

[65] S. Saberi and M. N. Fesharaki, An intelligent architecture for distributed data and knowledge management in a network-centric organization, *Proc. of the 5th International Conference on Semantics Knowledge and Grid*, Zhuhal, China, pp.250-253, 2009.

[66] S. Saberi, P. Trunfio, DomenicoTalia, M. N. Fesharaki and K. Badie, Using social network and semantic overlay network approaches to share knowledge in distributed data mining scenarios, *Proc. of International Conference on High Performance Computing and Simulation*, Caen, France, pp.536-544, 2010.

[67] M. G. Ceruti, Data management challenges and development for military information systems, *IEEE Transactions on Knowledge and Data Engineering*, vol.15, no.5, pp.1059-1068, 2003.

[68] M. R. Badeloo, B. Moshiri and B. N. Araabi, Application of OWA on a robot detector using data fusion, *Journal of Control, A Publication of Iranian Society of Instrumentation and Control Engineers*, vol.3, no.1, 2009.

[69] A. H. Keyhanipour, B. Moshiri, M. Kazemian, M. Piroozmand and C. Lucas, Aggregation of web search engines based on users' preferences in WebFusion, *Journal of Knowledge-Based Systems*, vol.20, no.4, pp.321-328, 2007.

[70] I. V. Maslov and I. Gertner, Multi-sensor fusion: An evolutionary algorithm approach, information fusion, *Journal of Information Fusion*, vol.7, no.3, pp.304-330, 2006.

[71] X. Li, X. Huang, J. Dezert, L. Duan and M. Wang, A successful application of DSMT in sonar grid map building and comparison with DST-based approach, *International Journal of Innovative Computing, Information and Control*, vol.3, no.3, pp.539-549, 2007.

[72] W. Li, K. Guo and Y. Huo, Study on decision making method under uncertain information based on D-S evidence theory, *International Journal of Innovative Computing, Information and Control*, vol.6, no.8, pp.3737-3749, 2010.

[73] M. Tabassian, R. Ghaderi and R. Ebrahimpour, Handling classification problems with imperfect labels using an evidence-based neural network ensemble, *International Journal of Innovative Computing, Information and Control*, vol.7, no.12, pp.7051-7066, 2011.

[74] E. Dorion and S. Fortin, Multi-source semantic integration-revisiting the theory of signs and ontology alignment principles, *Proc. of the 10th International Conference on Information Fusion*, Quebec, Canada, 2007.

[75] N. Baumgartner and W. Retschitzegger, A survey of upper ontologies for situation awareness, *Proc. of IASTED International Conference on Knowledge Sharing and Collaborative Engineering*, St. Thomas, U.S. Virgin Islands, 2006.

[76] M. M. Kokar, C. J. Matheus and K. Baclawski, Ontology-based situation awareness, *Journal of Information Fusion*, vol.10, no.1, pp.83-98, 2009.

[77] A. Claire and B. Brisset, Ontology based approach for information fusion, *Proc. of the 6th International Conference on Information Fusion*, Cairns, Australia, pp.522-529, 2003.

[78] Y. Liu, Z. Sui, Q. Zhao, Y. Hu and R. Wang, On automatic construction of medical ontology concept's description architecture, *International Journal of Innovative Computing, Information and Control*, vol.8, no.5(B), pp.3601-3616, 2012.

[79] P. Bresciani, A. Perini, P. Giorgini, F. Giunchiglia and J. Mylopoulos, Tropos: An agent-oriented software development methodology, *Autonomous Agents and Multi-Agent Systems*, vol.8, no.3, pp.203-236, 2004.

[80] J. Castro, M. Kolp and J. Mylopoulos, Towards requirements-driven information systems engineering: The tropos project, *Journal of Information System*, vol.27, pp.365-389, 2002.

[81] P. Giorgini, S. Rizzi and M. Garzetti, GRAnD: A goal-oriented approach to requirement analysis in data warehouses, *Journal of Decision Support Systems*, vol.45, no.1, pp.4-21, 2008.

[82] V. Bryl, P. Giorgini and J. Mylopoulos, Designing socio-technical systems: From stakeholder goals to social networks, *Requirements Engineering*, vol.14, no.1, pp.47-70, 2009.

[83] F. S. N. Fard, H. Parvar, M. E. Shiri and E. Soleimani, Using self-configurable particle swarm optimization for allocation position of rescue robots, *Proc. of the 2nd International Conference on Computer and Network Technology*, Bangkok, pp.362-366, 2010.