Proceedings of the 2017 Winter Simulation Conference
W. K. V. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, eds.

USING SITUATIONAL AWARENESS FOR ADAPTIVE DECISION MAKING IN AGENT-BASED SIMULATION

Okan Topçu
Department of Computer Engineering
Middle East Technical University
Northern Cyprus Campus (NCC)
99738, Kalkanli, Guzelyurt, Mersin 10, TURKEY

ABSTRACT

In an agent-based simulation, a plausible decision in a specific context cannot stay be valid in the face of the changing situation. Therefore, the result of the decision making process is mostly related to the agent's situational awareness and its adaptation to the new context and situation of the environment. A sound estimation of the situation requires a clean understanding of the operational domain, not only the data taken from sensors. In this paper, it is aimed to improve decision making by increasing the situational awareness of an agent by incorporating the decision making mechanism with the prior knowledge about the problem domain, such as the existing rules. Specifically, an existing adaptive decision making architecture, which is based on the deliberative coherence theory, is adopted to be driven by situational awareness. Moreover, a case study incorporating unmanned surface vehicles that are aware of the international sea traffic rules is demonstrated.

1 INTRODUCTION

Decision making problem is studied by many disciplines for a long time. A desirable characteristic of an autonomous agent is to have the capability to choose and execute tasks in accordance with the objectives of the prescribed duties and to continuously evaluate the state of the environment by taking into account the environmental changes. Autonomous agents should have a decision making mechanism that provides adequate facilities to select the most appropriate tasks (plans) that are valid in the current context and situation. The crucial point of selecting the most appropriate task and effective execution of the objectives is to construct a convenient decision making mechanism and to feed this mechanism with the accurate and instant information that is valid in the current context, where the situation is estimated by observing and comprehending the changing operational environment constantly. A sound estimation of the situation requires a clean understanding of the operational domain, not only the perceived information taken from its run-time monitoring mechanisms, to increase the situational awareness of the agent. For this reason, the decision making mechanism and the situational awareness of an agent must be in harmony in execution. The decision making mechanism of the agent must also adapt to the changing context and situation. Furthermore, the cognitive model of the agent that forms the decision making mechanism must be separate and explicitly specified from the agent's implementation.

In this regard, this paper presents an approach how to improve decision-making using the situational awareness of an agent by incorporating the decision making mechanism with the prior knowledge (e.g. the existing rules) about the problem domain. The proposed approach brings together the situational awareness and decision-making under a unified connectionist network. In this study, the adaptive decision-making mechanism presented in (Topçu 2014) is extended with the situational awareness capability and then a case study incorporating unmanned surface vehicles (USV) that are aware of the international sea traffic rules is implemented for demonstrating the proposed approach.

978-1-5386-3428-8/17/$31.00 ©2017 IEEE

The paper is organized as follows: First, the existing adaptive decision-making mechanism is introduced, then motivation of the study, the technique, and a brief review of context and situational awareness is presented. Later, the process of the situational awareness is given. Then, a case study is provided to demonstrate the proposed process and to present the implementation of the approach. Finally, conclusions and suggested future work conclude the article.

1.1 Adaptive Decision-Making Architecture

From the view point of agent-based simulation discipline, we may narrow the decision making problem as goal deliberation for an agent. The goal deliberation can be formulated as a constraint satisfaction problem (Thagard and Verbeurgt 1998), where goals and tasks found in an execution plan of an autonomous agent have relations among them setting constraints such as facilitation and inhibition. Thagard and Millgram (1995) present a solution for goal inference by proposing coherence based approach. The coherence considerations in practical reasoning constitute the deliberative coherence, a motivational coherence, called Theory of Deliberative Coherence (TDC) (Millgram and Thagard 1996), which is based on the Explanatory Coherence Theory (Thagard 1989). In this approach, the constraint satisfaction is computed by using a parallel distributed processing algorithm (McClelland 2014), where the inference is done by selecting the most satisfied constraints. And then the elements (i.e. goals and tasks) are divided into two sets as activated or rejected, where the former set forms the most satisfied constraints and thus it is selected as the inferred plan to execute (or the decision made). The elements in the plan and their constraints constitute a model called a connectionist network (model), which can be seen as a kind of recurrent neural network. Here, the elements correspond to units and the constraints correspond to links. This connectionist network consists of the goals and the tasks that form the structure of decision making.

Topçu (2014) put TDC in an agent-based simulation context, focusing the implementation perspective of TDC with a Systems Engineering view point, to show how an adaptive decision making can be realized. As a result, the Deliberative Coherence Driven Agent (DeCoAgent) is proposed as a coherence-driven agent that has adaptive decision making ability based on the TDC computations (Topçu 2014). A practitioner may implement DeCoAgents employing DeCoAgent library (İşçi, Topçu and Yılmaz 2014), which is available as open source (İşçi and Topçu 2015). DeCoAgent library is implemented as an extension to Jadex (Braubach, Lamersdorf and Pokahr 2003), which is an agent development and execution environment. In this work, this library is ported to MS .NET development environment (DeCoAgent 2015) and then is extended to cover the proposed (situational awareness) techniques presented in this paper.

1.2 Motivation

The connectionist network is a local artificial neural network (ANN) model, where each goal and action in a connectionist network is represented by an individual unit. In other words, the units and links of the network represent the knowledge (the elements and their relations) locally. The default relation strengths are predefined and coded for all models. Therefore, when a decision problem is given (or identified by an agent), agent adapts its connectionist network, and then computes the activations of each unit to find the ones that maximize the coherence. The primary goal units in the network take a priority value as input according to the agent situation. Then, the agent computes the activation potentials for each unit and decides which goals/tasks are active according to those priorities. Here, the major consideration is to feed the network with the correct priority input values according to the situation of the agent. Therefore, the situational awareness of the agent is very affective on the given decision. Moreover, for many situations, we have some prior knowledge. Prior knowledge can be in different forms and encapsulated in agent's situational awareness. For instance, for some situations, we know what a correct decision is and what the desired response of an agent should be. Or we know the rules of the environment fully or partially (e.g. game rules). Or we may have some domain-specific knowledge coming from another system (e.g. an operational order from a command and control (C2) system or a scenario from a simulation manager). In the presence of a prior knowledge, if the agent can use this prior knowledge to increase its situational awareness, then we may feed the decision mechanism (i.e. connectionist network) with the predicted priorities according to the situation. Consequently, we may assert that the decision making ability of an agent is increased and the agent's situational awareness is used more affectively in the decision making process.

1.3 Technique

The decision making behavior of a connectionist network can be influenced in two ways: (i) to change the priorities of the goals, and (ii) to change the structure (topology) of the connectionist model. In this paper, the connectionist model behaviors are determined by changing the priorities of the primary goals. This work ensures that the prior knowledge is formed by the agent's situational awareness. So, we aim to increase the situational awareness of an agent by training the existing rules in the operational domain. In order to improve the situational awareness of an agent, first, contexts are determined by evaluating the information from sensors and existing rules, and secondly, the agent's situation is attained by testing the contexts in the ANN.

1.4 Context and Situational Awareness

Human beings are fairly capable of transferring ideas, thoughts, and emotions to be understood by others. When humans talk with humans, they are able to use implicit situational information or context (Dey 2001). In a contrary manner, the agents are not as successful as the humans on this issue. The agent's situational awareness is obtained from contexts. "Context is any information that can be used to characterize the situation of entity" (Dey and Abowd 2000). Here, we can consider the entity as an agent. If a factor can be used to specify the situation of the agent, then this factor is considered as context. All the contextual factors compose the current situation of entire environment. The current situation known by the agent is defined as the agent's situational awareness. As discussed earlier, the agent's situational awareness is very influential for the validity of the decision made. Endsley defines the situation awareness as "the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future" (Endsley 1995). In other words, situation awareness is to be aware of what is happening in the dynamic environment, aggregating which context factors influence the situation, and understanding of relationship between actual information. Before decision and action, Endsley divides situation awareness into three levels. These are perception of elements in current situation (Level 1), comprehension of current situation (Level 2), and projection of future status (Level 3). Level 1 is to perceive prominent information in the environment. Level 2 is to comprehend what this information emphasizes. And Level 3 is to predict the future situations from the existing information. In this paper, an agent, DeCoAgent, having the capability of first two levels and partially the third level is explained. So, a DeCoAgent is capable of being aware of situations in a dynamic environment, adjusting its goal priorities dynamically, and making changes in its decision making model.

2 PROCESS

The adaptive decision making driven by the situational awareness incorporates a supervised learning process. The learning process encompasses two phases: training the existing rules of the domain to the agent and then testing as depicted in Figure 1. Actually, training can be done prior to the agent execution. In the training phase, each network is trained using the existing rules about the environment before the agent simulation run. This phase is called as adapting the context. In the simulation run, the agent employs this trained (situational awareness) network to determine its current situation to facilitate the decision making. The situation of the environment and the agent internal state is transformed into appropriate inputs, called as situation model, to decision making mechanism. So, this phase is called as using the context. The situation model is an abstraction of the state of the situated agent at a specific time or context. The representation of the knowledge of the situation model can vary from a simple direct binary coding of sensor inputs to complex object models. For the sake of simplification, we will assume that the representation of the situation model is as simply a context vector representing the priorities for each goal. For instance, in a naval simulation, when a surface contact is identified as a friendly ship, then the priorities of each goal are set accordingly. Thus, an input vector (1 0 1) for the current situation represents the goal priorities for g_1, g_2, and g_3 respectively (see right drawing in Figure 1).

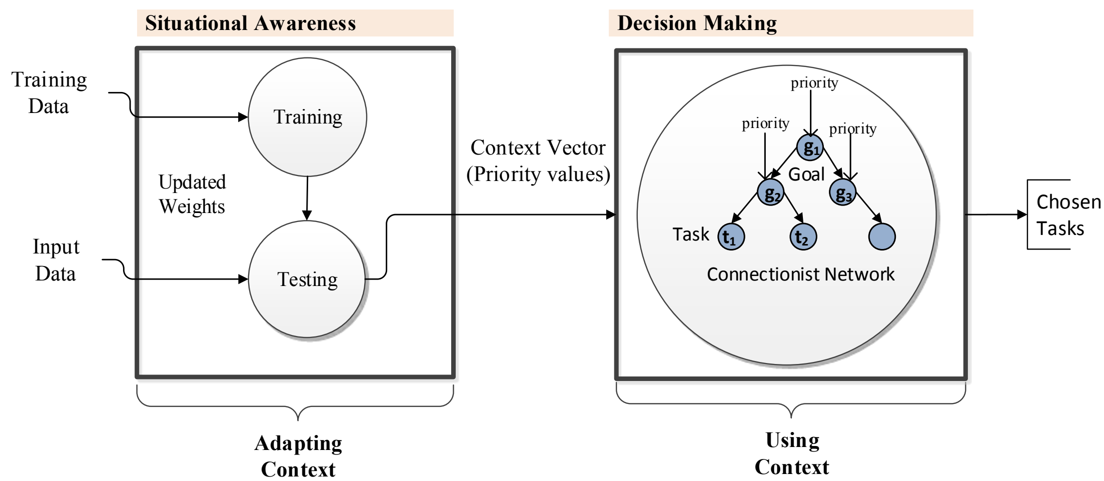

Figure 1: Adapting and using context.

The situational awareness process is modeled with cascaded ANNs connected with each other. An assemble architecture, which is composed of feedforward ANNs and a back propagation ANN, is given in the case study section. In the situational awareness network, the input data are obtained from various available sensors (for instance, geodetic position of a surface contact obtained from a navigation radar). Using this data, the ANNs, which are devised to represent different sensors and context factors, determine the contexts of the agent by computing outputs. Then, using those contexts, the agent's situation is determined. Lastly, this situation is fed to the decision making mechanism (i.e. the connectionist network) as priorities (context vector in Figure 1).

In the training phase, the results (i.e. actual response) are compared with the desired output. If the error rate between the actual and the desired response is larger than a predefined error rate, then the error is back propagated. Thus the weights are re-adjusted to lower the error. At the end of training, the weights of each neuron is adjusted and updated in a way that represents the problem domain. As the learning algorithm requires a set of desired response to a given input, it is called supervised learning.

In the decision making phase, the primary goals in the network take a priority value according to the agent situation and agent calculates the activations (i.e. activation potential) of each unit and decides which goals/tasks are active according to those priorities. As a result, the capability of decision making of an agent is improved due to the fact that the connectionist network is adjusted according to the domain knowledge and is ready for goal deliberation in response to the new situations.

The following case study demonstrates in detail how this process is modularized and implemented.

3 CASE STUDY: NAVIGATION SAFETY

As a case study, we will use a modified version of patrol mission scenario (Ünal and Topçu 2014) given to a group of autonomous unmanned surface vehicles (USV). In the scenario, the top goal of USV is to conduct intelligence, surveillance, and reconnaissance operations against sea piracy and it ensures maritime security of the merchant vessels. USV is responsible for patrolling, monitoring, preventing the sea piracy and protecting the merchant vessels. USV carries out all of these operations in the responsibility area (Topçu 2014). In this paper, the case study is extended as a proof of concept to demonstrate that an autonomous USV gains the ability of safe navigation by employing the approach, presented in this paper, where the situational awareness mechanism is integrated to its decision-making mechanism. The extended USV has the ability to navigate safely in the sea conforming to the international rules of the road as it carries out its assigned tasks (e.g. traverse task) against sea piracy and ensures the maritime security of merchant vessels. As a result, the USV autonomously decides when to maneuver to avoid collision (according to the international rules) or when to execute its assigned task using the approach proposed in this paper. The previous version of the USV decision making mechanism did not have this capability.

Seaman experience the numberless collision events at sea daily. In general, the ship collision event is induced by human error. In this regard, this paper also presents an agent-based simulation system for the decision making process avoiding ship collision in maritime traffic. If an autonomous agent having collision avoidance decision-making process and updating its decision making abilities considering the uncertain environmental conditions and situations, collision events can be reduced.

In order to obtain situational awareness from the sensor information, the data from the sensors are evaluated in a context by taking account of all the rules affecting the environment. In this regard, information is obtained from maritime regulations from International Regulations for Preventing Collisions at Sea (COLREG) (IMO 2003) and training them to USV. The USV agent can make decision to determine its situation by receiving the inputs from the ship's sensors and COLREG, which is trained to cascading neural networks. In this work, employing the feedforward backpropagation neural network as the ANN architecture has two features. First, the feedforward feature of it ensures to process and recall patterns. Second, the backpropagation feature of it can train the network.

In the scope of the case study, it is demonstrated how to train the USV with the international safety navigation rules first, and then how the USV employs this to predict the current situation and integrates it with its decision mechanism. Therefore, the situational awareness capability developed in this case study has a potential to realize the e-navigation concept. In order not to disrupt the scope, the collision avoidance maneuvers are implemented in its simplest forms (e.g. changing the course using a predefined degrees).

3.1 Agent Architecture

Each USV in the case study is modeled and implemented as a DeCoAgent. The architecture of USV agent, depicted in Figure 2 (next page), conforms to the DeCoAgent generic architecture presented in (Topçu 2014).

Each USV has an adaptive decision making module for goal reasoning driven by a situational awareness manager (i.e. Navigation Advisor). Navigation advisor is used to feed the goal priorities according to the current situation. Then the goal deliberator (coherence computation component) decides which goals are active/valid in the current situation, where it maintains a connectionist network of tasks, goals, and the relations among them and calculates the activations and the total coherence of the connectionist network. The coherence computation continues until the activation of nodes in the connectionist network becomes stable. Task selection is simply mapping the active/valid goals to the related tasks (execution plan) that control the actuators (e.g. rudder). The adaptive decision making and the architecture of the situational awareness component is presented in the following sections.

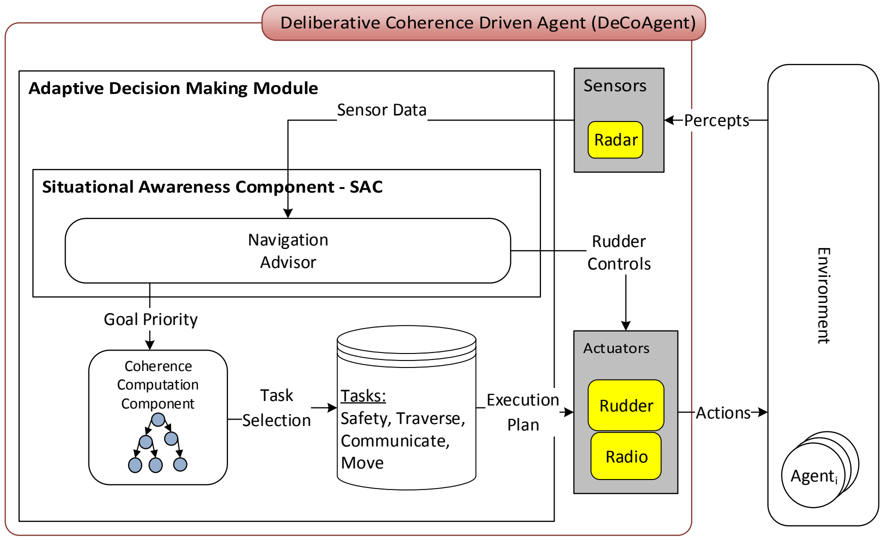

Figure 2: The architecture of USV DeCoAgent.

3.2 Adaptive Decision Making

The principle duty of the agent's adaptive decision making mechanism is to select which goals to be persuaded. In order to do that agent monitors the environment, estimates the situation, evaluates the goals and tasks, determines the active and valid goals, and executes the appropriate plan of the selected goals and tasks. The agent uses the connectionist network for goal reasoning and making decision. The connectionist network of the USV is simplified to depict the new dynamic situational awareness. A modified version of the connectionist network of the USV decision making mechanism is presented in Figure 3.

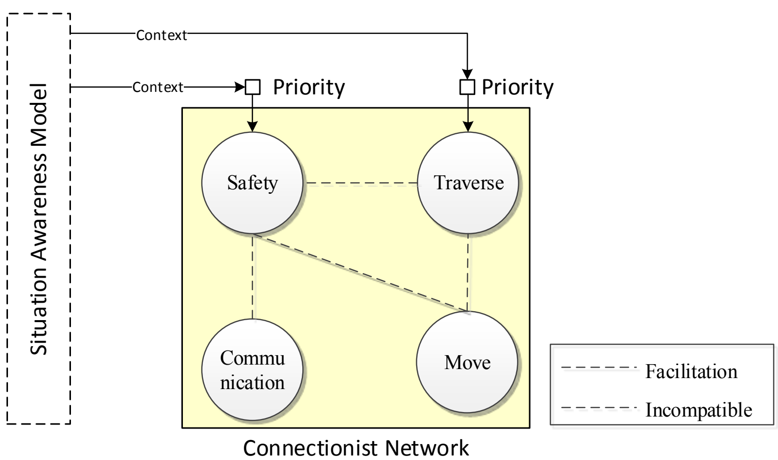

Figure 3: A sample USV connectionist network used for decision making.

According to deliberative coherence theory, the each unit is considered as a goal or an action (e.g. Safety and Traverse in Figure 3) in the connectionist network and a unit is considered as a neuron in the ANN. If the unit facilitates the other, there is a positive constraint between them (e.g. Communication task facilitates Safety goal). If the unit is incompatible with other, there is a negative constraint between them (e.g. Safety and Traverse goals are incompatible). The positive constraint is depicted with a straight line, the negative constraint is depicted with a dashed line in the connectionist network. In the partial connectionist model of USV decision making mechanism, some goals are desirable. There is a priority value of each primary goal (e.g. a priority for Traverse and Safety goals). The goal priority determines how much a goal is desirable. These priorities of primary goals are dynamically fed by the agent's situational awareness model including the agent and the environment states. Finally, the USV decision making mechanism computes unit activation potentials (i.e. makes a decision) by taking account of these goal priorities in order to determine which goals to pursue. The connectionist network in the context of agent-based simulation is discussed in detail in (Topçu 2014).

4 SITUATIONAL AWARENESS MODEL

The proposed model to evaluate the ship collision risk is attributed as a knowledge based system. This knowledge based system includes ship to ship collision situations and collision avoidance maneuvers by considering the COLREG. COLREG is composed of international rules that are treated as international laws, and it is well known for sailors. If a risk of collision between two power-driven vessels is deemed to exist, the mariners on the bridge decide on the need for any collision avoidance maneuver through their assessment of the collision regulation. The mariners take into account the International Collision Regulations to diminish or thoroughly remove the occurrence of risk of collision with another ship. In the case study, the encounter situations between two ships are assessed by the autonomous agent that having decision-making ability considering not the whole COLREG rules, but the main types of situations. The COLREG rules (IMO 2003) considered in this study are Rule 7, 8, 11, and 13 - 17. Additionally, according to COLREG, if the give-way ship does not take appropriate maneuver to avoid collision on the encounter situation, the stand on ship should take action. This situation is seen as a behavior anomaly.

4.1 Architecture – Cascading Neural Networks

There are many context factors that affect the situation in safe navigation. For instance, in night, the navigation lights seen by the officer of the watch, and the relative sector, where those lights are seen, specify different context factors. As the combined contexts specify the situation, we use an ANN for each context factor and then cascade them to define the situation. Thus, the cascaded neural network feeds the situation networks (see Figure 4).

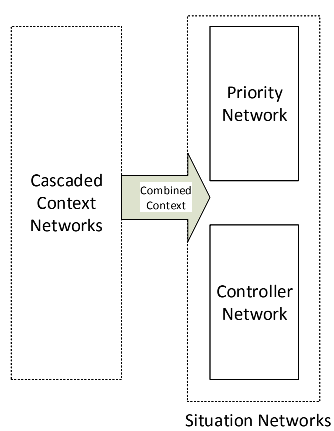

Figure 4: Context networks and situation networks.

The priority network is used to determine the priorities of primary goals in the connectionist model as the controller network is used to determine the action that will be taken by the actuators (i.e. the rudder). Both networks are perceptron networks. Figure 5 depicts the detailed architecture of situational awareness network for COLREG capability. In this model, the input data consists of the information of own ship (the position, speed, and course) and of the surface contacts (i.e. target surface ships) such as the speed, course, and true bearing, which are obtained from the sensors. This information is preprocessed and then, the results that attained are evaluated by feedforward neural networks found in input layer.

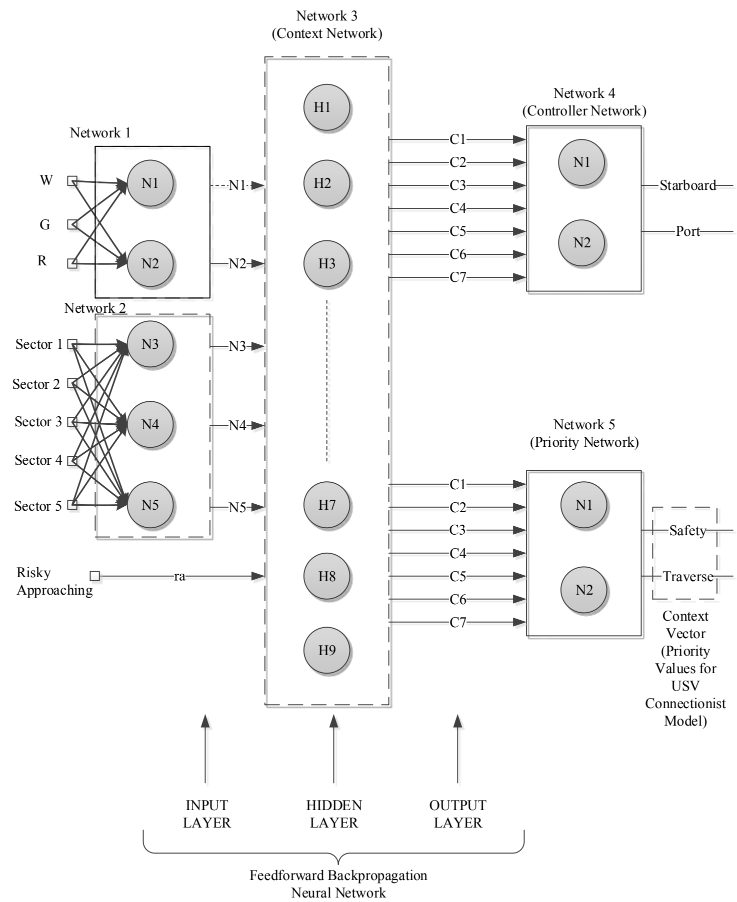

Figure 5: The structure of feedforward backpropagation neural network for ship collision avoidance (Sayar 2015).

The feedforward neural network is employed as the main model for ANNs found in Input Layer. The information that is processed moves in one direction and forward from input layer to output layer. The feedforward feature of this neural network ensures to process and recall patterns. Thus, the contexts related to the environment acquired as the output after the process of the first neural network layer. The data obtained from sensors (e.g. track position) is preprocessed to obtain input for context networks. Here, data obtained from Radar and compass are transformed into data that will be used as input in context networks.

The first neural network (See Network 1 in Figure 5) is used to determine the context of navigation lights of the target ships as defined in COLREGs in night navigation. From this network, the (navigation lights) context is acquired by processing the input data (W, G, and R) about navigation lights of target ship seen by the own ship. W, G, and R stands for the White, Green, and Red navigation lights respectively. This network is basic feedforward perceptron network with 3 inputs and 2 outputs.

All the rules of the COLREG apply in all respects, but the mariners particularly consider COLREG Rule 13, 14, 15, and 16 of Part B, Section II (IMO 2003), when making decisions on finding effective collision avoidance maneuver. When these rules are evaluated and interpreted, three main types of situations and five different sectors are obtained (Tsou and Hsueh 2010). The sector context network (Network 2 in Figure 5) is a basic feedforward perceptron network with 5 inputs and 3 outputs. The input data (i.e. sectors S1 through S5) about the location of target ship relative to the own ship is used to determine the relative sector context.

The risky approaching input fed to the Network-3 is used to determine the context of approaching and safety condition of the target ships. It determines whether our ship is approaching the track and the Distance to Closest Point of Approach (DCPA) is within the safety range. The safety context is acquired by computing two parameters processing the input data (Time to Closest Point of Approach (TCPA) and DCPA). After that, this context information is used as input data for the second layer (Network 3), as depicted in Figure 5, which is modelled as a feedforward back propagation neural network. This neural network architecture includes the input layer and output layer and additionally the hidden layer. The ship agent can make decision that it is in which situation by receiving the necessary inputs on the ship's sensors and COLREG. Through this layer, the agent's situational awareness is increased substantially by using the feature of back propagation can train the network. Then, the ship agent makes decision which the ship collision avoidance maneuver will apply for preventing the ship collision. At the end of the Network-3, the output ensures which collision avoidance maneuver as starboard (S) or/and port (P) is appropriate.

4.1.1 Feeding the Decision Mechanism

Using the cascaded neural networks, the priorities of decision making mechanism can be driven in some combinations. A situation neural network can feed all the priorities, or each priority is driven with a different situation network), or a hybrid of both configurations can be employed. In our case study, feeding mechanism conforms to the first configuration. Using these priorities, the USV agent calculates the activation potentials of each unit and decides which goals/tasks are active according to those priorities. Thus, the USV agent can navigate safely and avoid from a probable ship collision situation and perform more effective decision making process using the prior knowledge dictated by COLREG, which is formed by the agent's situational awareness, while it conducts these operations against sea piracy and ensures the maritime security of merchant vessels in its responsibility area. The networks and the related decision making parameters for each agent can be monitored separately.

4.2 Situational Awareness Component

The Situational Awareness Component (SAC) is responsible to evaluate the sensor inputs and then to feed the deliberation subsystem by deciding the goal/task priorities. One of the decisions made by a watch officer is to decide when to begin to carry out a COLREG rule and when to end it. In our case study, a COLREG range is pre-defined and is changeable by the user. It indicates the maximum range that a COLREG rule will be applied. It can be evaluated as the eye visibility range (COLREG Rule 11). When there are multiple tracks in COLREG range, then the SAC determines the critical tracks. A critical track is the one that has a TCPA, which is greater than zero, and the DCPA, which is lower than safety range, for each surface contact in the maximum COLREG range (i.e. in sight of the USV). Safety range is the captain's intent that no ship will enter this range. A TCPA greater than zero indicates that two ships are approaching each other. A TCPA smaller than zero indicates that the CPA has occurred in the past, and the ships are drafting apart. After determining all the critical tracks, the SAC sorts the critical tracks according to their TCPAs and selects the track with the smallest TCPA as the most critical track. Then, the USV determines its situation (i.e. which COLREG rule is applicable) according to this track. The situation neural networks calculate how to maneuver and if an action (e.g. turn to starboard) is required then the USV carries it out. The action is carried out until DCPA is greater than the safety distance. So, this is the stop condition for ending carrying the COLREG rule. When the COLREG rule indicates to keep the course and speed of the ship (e.g. crossing green lights), then the USV checks the second critical track to determine its situation and applies the same algorithm defined here.

Perception is the phase of collecting data from sensors by observing the environment. In the case study, radar (i.e. a sensor) provides all the tracks found in the radar cover range. Radar detection algorithm is simply filtering all the agents found in the radar cover range assuming that track position, heading, and speed is obtained from this radar scan. Using these data, radar computes the track distance, true bearing, TCPA**,** and DCPA according to the USV. The sensor data are used as input to the situational awareness model. Comprehension is the phase, where the agent assesses its situation by determining the contextual factors. Comprehension is carried out on two distinct level. At the top level, the perception is mapped to goal priorities to feed the deliberation subsystem (i.e. Coherence Computation Component). In the case study, there is no need to execute Safety task, when there is no track in maximum COLREG range. At the lower level, when the USV executes the Safety task (this indicates that it is in COLREG situation), it assesses its situation by determining COLREG contexts (e.g. track is at which sector, which navigational lights are seen etc.) and determines the rudder (i.e. an actuator) actions (port or starboard) by the help of the COLREG navigation advisor, which is a cascaded neural network that is trained with COLREG rules (see Figure 5). Projection is the prediction of the possible maneuver that a critical track must do according to COLREG. This is used to detect the behavior anomaly of other surface vessels that do not comply with COLREG. The situational awareness component predicts the course of actions (in terms of COLREG maneuver) of each critical track. Basically, the cascaded neural networks are rerun for each critical track. Thus, the projected behavior is estimated.

4.3 Related Work

In literature, there are many different researches and methods related to the navigation of autonomous systems in the scope of e-navigation (see (Sayar 2015) for a detailed overview). One of these techniques includes ANN Fuzzy Inference System (ANFIS) by Selma and Chouraqui (2013), Zhuo and Hearn (2008). In this model, a hybrid approach that combined neural and fuzzy techniques for problem of control is presented. This model is trained with the back propagation gradient descent method using appropriate input-output data. Another ANFIS model is proposed by Shing and Jang (1993). This model employs a hybrid learning model and is constructed an input-output mapping based on human knowledge and stipulated input-output data pairs. Tsuo and Hsueh applied a concept of e-navigation as a framework and Ant Colony Algorithm in the field of artificial intelligence to set up a collision avoidance model (Tsou and Hsueh 2010). This model consists of navigational practices, a maritime regulations knowledge and timely navigation information from e-navigation system. Liu and Shi use the fuzzy neural inference network model that constructed by combining the fuzzy technology and ANN (Liu and Shi 2005). This model comprised three subsets. The first two take the data from the user and make inference which ship encounter situation is. The subset 3 takes the outputs of other subsets as input and makes a decision. ANN models in the related studies are mostly employed as a controller for controlling the movement of a ship. In this work, the neural network is employed to determine and increase the situational awareness by training the network with the existing rules about the environment in order to improve the decision making ability of an agent.

5 CONCLUSIONS AND FUTURE WORK

The paper proposes a new approach to improve decision making by increasing the situational awareness of an agent by incorporating the prior knowledge about the environment, such as the existing rules, to the decision making mechanism. The situational awareness of an agent can be increased by training the existing rules about the environment, and then using this dynamic assessment, the state inputs in form of goal priorities can be fed into the agent's decision making mechanism to improve its decision making ability as a result. The prior knowledge is acquired at the end of training process of the agent's situational awareness component, where it is modeled as cascaded neural networks. Specifically, this work extends the adaptive decision making architecture, which is based on the deliberative coherence theory, by feeding it with cascaded neural networks to acquire the prior knowledge with supervised learning techniques that involve error correction (minimizing) learning rule. A case study is implemented as a proof of concept to demonstrate that an autonomous USV gains the ability of safe navigation using the situational awareness integration approach presented in the paper. The demonstrated case study includes the maritime regulations from International Regulations for Preventing Collisions at Sea as a knowledge base and navigational information about the surface vessels. The selected case study is an agent-based simulation that targets the e-navigation concept (autonomous navigation) and it is believed that it can be extended for realistic navigational traffic management by incorporating intelligent collision avoidance algorithms. The extended USV safely navigates in the sea conforming to the international rules of the road (i.e. COLREG) as it carries out its assigned tasks. As a result, the USV autonomously decides when to maneuver to avoid collision (according to the international rules) or when to execute its assigned task (e.g. traverse) using the approach proposed in this paper. This provided ability adds value to the previous autonomous USV models in agent-based simulations.

In order to develop the simulation, a DeCoAgent library is implemented. All the libraries and the case studies are available at DeCoAgent Web Site (DeCoAgent 2015). In this work, it is assumed that all the agents run in the same host but the work is ongoing with distributing the agents using High Level Architecture (HLA) (IEEE Std 1516-2010 2010). Moreover, the autonomous surface vehicles (agents) need to communicate with each other in order to avoid and prevent a possible collision. Sometimes, they just need to understand each other’s intention as the most man onboard ships do.

ACKNOWLEDGMENTS AND DISCLAIMER

This work has been supported and funded by the Scientific and Technological Research Council of Turkey under grant 114E008. The views and conclusions contained herein are those of the author and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of any affiliated organization or government.

REFERENCES

Braubach, L., W. Lamersdorf, and A. Pokahr. 2003. "Jadex: Implementing a BDI-Infrastructure for JADE Agents." *EXP: In Search of Innovation (special issue on JADE)* 3(3):76-85.

DeCoAgent. 2015. *DeCoAgent Framework Web Site.* Accessed July 1, 2017. https://sites.google.com/site/okantopcu/decoagent.

Dey, A. K., and G. D. Abowd. 2000. "Towards a Better Understanding of Context and Context-Awareness". *CHI 2000 Workshop on the What, Who, Where, When, and How of Context Awareness.*

Dey, A. K. 2001. "Understanding and Using Context." *Personal and Ubiqitous Computing* 5(1):4-7.

Endsley, M. R. 1995. "Toward a Theory of Situation Awareness in Dynamic Systems". *Human Factors* 37(1):32-64.

IEEE Std 1516-2010. 2010. *Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) - Framework and Rules.* New York: IEEE.

IMO. 2003. *COLREG: Convention on the International Regulations for Preventing Collisions at Sea, 1972. Consolidated Edition 2003.* 4th. IMO (International Maritime Organisation).

İşçi, S., and O. Topçu. 2015. *DeCoAgent Library.* Accessed July 1, 2017. http://sourceforge.net/projects/decoagentlibrary/.

İşçi, S., O. Topçu, and L. Yılmaz. 2014. "Extending the Jadex Framework with Coherence-Driven Adaptive Agent Decision-Making Model". In *Proceedings of the IEEE/WIC/ACM Conference on Intelligent Agent Technology (IAT 2014).* Warsaw, Poland. 48-55. doi:10.1109/WI-IAT.2014.149.

Liu, Y., and C. Shi. 2005. "A Fuzzy-Neural Inference Network for Ship Collision Avoidance". In *Proceedings of the Fourth International Conference on Machine Learning and Cybernetics.* Guangzhou.

McClelland, J. L. 2014. *Explorations in Parallel Distributed Processing: A Handbook of Models, Programs, and Exercises.* Stanford. Accessed July 1, 2017. https://web.stanford.edu/group/pdplab/pdphandbook/.

Millgram, E., and P. Thagard. 1996. "Deliberative Coherence". *Synthese* 108 (1):63-88.

Sayar, O. 2015. *Improving Decision Making with Increased Situational Awareness.* MSc Thesis, C4I, Naval Science and Engineering Institute, Istanbul: Turkish Naval Academy Naval Science and Engineering Institute, 85.

Selma, B., and S. Chouraqui. 2013. "Neuro-Fuzzy Controller to Navigate an Unmanned Vehicle." *SpringerPlus* 2 (188):1-8.

Shing, J., and R. Jang. 1993. "ANFIS: Adaptive-Network-Based Fuzzy Inference." *IEEE Transactions on Systems, Man, and Cybernetics* 23(3):665-685.

Thagard, P. 1989. "Explanatory Coherence". *Behavioral and Brain Sciences* 12 (3):435-502.

Thagard, P., and E. Millgram. 1995. "Inference to the Best Plan: A Coherence Theory of Decision". In *Goal-driven Learning*, edited by Ashwin Ram and David B. Leake, 439-454. Cambridge, MA: MIT Press.

Thagard, P., and K. Verbeurgt. 1998. "Coherence as Constraint Satisfaction". *Cognitive Science* 22(1):1-24.

Topçu, O. 2014. "Adaptive Decision Making in Agent-based Simulation". *Journal of Simulation: Transactions of the Society for Modeling and Simulation International* (SAGE) 90 (7): 815-832. doi:10.1177/0037549714536930.

Tsou, M., and C. Hsueh. 2010. "The Study of Ship Collision Avoidance Route Planning by Ant Colony Algorithm". *Journal of Marine Science and Technology* 18(5):746-756.

Ünal, O., and O. Topçu. 2014. "Modeling Unmanned Surface Vehicle Patrol Mission with C-BML". *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology* 11(3):277-308. doi:10.1177/1548512912475107.

Zhuo, Y., and E. Hearn. 2008. "A Ship Based Intelligent Anti-Collision Decision-Making Support System Utilizing Trial Manoeuvers". *In Chinese Control and Decision Conference.* Yantai, China.

AUTHOR BIOGRAPHIES

OKAN TOPÇU is an Associate Professor in the department of Computer Engineering at the Middle East Technical University, Northern Cyprus Campus (METU NCC). Professor Topçu received his Ph.D. and M.S. degrees from the METU, Ankara, Turkey in 2007 and 1999 respectively. His current research interests include modeling and simulation, distributed simulation, agent-based simulation, intelligent agents, autonomous systems, model driven engineering, and creative evolutionary systems. His e-mail address is okantopcu@gmail.com.