Received: 28 November 2017 | Revised: 6 February 2018 | Accepted: 9 April 2018

DOI: 10.4218/etrij.2017-0294

SPECIAL ISSUE

WILEY ETRI Journal

Comprehensive architecture for intelligent adaptive interface in the field of single-human multiple-robot interaction

Mahdi Ilbeygi | **Mohammad Reza Kangavari**

Iran University of Science and Technology (IUST), Tehran, Iran.

Correspondence
Mahdi Ilbeygi, Iran University of Science and Technology (IUST), Tehran, Iran.
Email: mahdi_ilbeygi@iust.ac.ir

Nowadays, with progresses in robotic science, the design and implementation of a mechanism for human–robot interaction with a low workload is inevitable. One notable challenge in this field is the interaction between a single human and a group of robots. Therefore, we propose a new comprehensive framework for single-human multiple-robot remote interaction that can form an efficient intelligent adaptive interaction (IAI). Our interaction system can thoroughly adapt itself to changes in interaction context and user states. Some advantages of our devised IAI framework are lower workload, higher level of situation awareness, and efficient interaction. In this paper, we introduce a new IAI architecture as our comprehensive mechanism. In order to practically examine the architecture, we implemented our proposed IAI to control a group of unmanned aerial vehicles (UAVs) under different scenarios. The results show that our devised IAI framework can effectively reduce human workload and the level of situation awareness, and concurrently foster the mission completion percentage of the UAVs.

KEYWORDS
cognitive model, context aware, hierarchical timed colored petri net, human–robot interaction, intelligent adaptive interface

1 | INTRODUCTION

Along with advances in robotics and artificial intelligence, we are witnessing the omnipresence of robots in human life. Robot autonomy is gradually improving, which indicates we must interact with them as an intelligent entity. Therefore, the design and development of an efficient mechanism to facilitate the interaction between each human user and multiple robots is a serious challenge [1].

In general, human–robot interaction (HRI) is divided into two categories [2]:

1. Remote interaction: The human and the robot are not collocated and are separated spatially or even temporally.
2. Proximate interaction: The human and the robot are collocated.

In this paper, we consider remote interaction with robots that is often referred to as supervisory control.

Note that HRI is not only limited to information/command exchange between humans and robots, and it has other aspects such as situation awareness, user cognition consistency, and identifying changes in interaction context. User cognitive assistance and security must be considered in HRI. For example, it is ideal to have an interaction between human and robots that can provide a high level of situation awareness and simultaneously demonstrate a low level of cognition and perception load; this problem was focused on in many studies [3,4]. Therefore, the complexity of such an interaction is very high.

Considering the more complex problem of single human-multiple robots interaction in a dynamic environment, an intelligent adaptive interaction (IAI) mechanism is required to overcome the high complexity of such an interaction. IAI is an interface that dynamically adapts its control and display characteristics to react in real time to task, user, system, and environment states [5]. Some definitions and practices have been reported for intelligent adaptive interfaces in [6] and [7].

Some of the interesting aspects of IAI have been reported and elaborated in the literature. For example, an intelligent adaptive interface that focused on situation awareness was presented in [7]. A limited model-based context aware adaptive interface is developed in [1]. Further, cognitive assistance adaptive interface for guidance of multiple unmanned aerial vehicles (UAVs) has been introduced by [8].

Our study shows that none of the reported research studies have investigated IAI thoroughly. Therefore, we decided to propose a comprehensive architecture for interaction between a single human and a group of robots that provides all mentioned important aspects and has some notable features as follows:

1. Adaptation to all interaction context changes including the user states, interaction environment, goals, and the tasks.
2. Providing a framework to deliver the right information to human user at the right time.
3. Performing secure interaction with the lowest level of user intervention.
4. Supporting users through cognitive assistance and system behavior explanation.
5. Considering self-configuration and self-optimization for the proposed architecture to update its knowledge base over time.
6. Using ontology for modeling knowledge about concepts and relationships between them in the interaction environment.
7. Considering inter-relationship and collaboration of robots in the interaction process.
8. Proposing a method to detect new concepts and relationships in the interaction context and modeling them into the ontology.
9. Providing continuous/active authentication by user modeling and dedicated security unit.
10. Considering the user forgetting model for delivering information to his/her.

It must be noted that, to the best of our knowledge, some of these features (features 5–9) have not been presented in any previous architectures. Further, our results for some of these features were better than that provided by available architectures. For example, none of the previous works achieved secure interaction with the lowest level of user intervention, as presented in our architecture. We integrated all these features into a comprehensive architecture.

The rest of this paper is organized as follows: First, an overview to related works has been provided, in Section 2. In Section 3, the proposed architecture is elucidated and its components and the roles are further explained. Section 4 deals with the modeling of the proposed architecture based on the hierarchical timed colored petri net. The simulation of the IAI unified framework using multi-UAVs and the relevant results are presented in Section 5. Finally, our conclusions are reported in Section 6.

2 | RELATED WORKS

In this section, more details of some popular architectures and frameworks in the fields of context aware and IAI are provided.

A general framework for context-aware adaptive user interface generation is proposed in [1]. This framework covers some important issues of generating a context-aware adaptive interface such as modeling tasks, domains, users, dialogs, and presentation. However, the focus of this study is on a context modeling tool. Therefore, the remaining issues in IAI and context-aware user interface generation such as self-configuration, self-optimization, user assistance, user decision support, ability to explain system behavior, and security, have been neglected in this framework.

Some articles only focused on a single topic of intelligent adaptive interfaces. For example, the intelligent situation awareness-adaptive interface (ISAAI) [7] focuses on user situation awareness. ISAAI monitors user situation awareness only using eye-gaze tracking, and it guides the user's visual attention to relevant but unattended information. Hence, the other important issues in the intelligent adaptive interface architecture have been excluded.

In [9], a conceptual architecture of an intelligent adaptive interface was suggested. This architecture has four main units: situation assessment and support system, operator state assessment, adaptation engine, and the operator machine interface (OMI). Situation and operator state assessment have been adequately addressed by this architecture; however, other significant issues—the security, self-configuration, self-optimization, knowledgebase adaptation to the newly detected concepts in the context, and system state assessment—have been omitted in this architecture.

Another generic conceptual framework for developing IAIs was proposed in [5,6]. This framework uses a multiple-agent hierarchical structure; such agents are called adaptive intelligent agents (AIA). In these research, there are three function groups that represent a hierarchy of various AIAs: senior agent, working agent, and junior agent.

Owing to the multiple-agent hierarchical structure of this framework, it has an appropriate development capability. However, self-optimization remains ignored, and no attention has been paid to interaction security, and other topics such as collaboration between target agents in the interaction process. We considered all these shortcomings in our IAI architecture and provided an infrastructure for continuous/active authentication.

A valuable survey on adaptive model-driven user interface is presented in [10]; this survey introduces a set of properties to evaluate the performance of the state-of-the-art models: levels of abstraction, adaptive behavior, direct and indirect adaptation, and user feedback on the adapted UI are among important evaluation attributes. However, given the shortcomings of previous works, we decided to propose a new comprehensive architecture for adaptive intelligent interface in the field of single-human multiple-robot interaction.

3 | PROPOSED IAI ARCHITECTURE

Figure 1 depicts the proposed architecture at the first level.

Figure 1 shows the main components of the proposed architecture with several numbered relations between them. The duties and roles of the main components depicted in Figure 1 are described below.

- **UI**: A user interface (UI) is the final interface for delivering data and information to the end user and for receiving commands. The UI can be a simple graphical UI (GUI) with standard input and output devices or a virtual reality (VR) interface with new input and output devices including haptic, gesture recognition, and EEG interpreter devices.
- **User command controller (UCC)**: This unit receives commands from the user and classifies these commands into two categories: robot commands and IAI commands. Then, it sends robot commands to corresponding robot(s) and delivers IAI commands to the intelligent adaptive interface management (IAIM) unit. In addition, a copy of every command is sent to the context acquisition and integration unit for user activities registration and monitoring. The UCC unit can use the joint architecture for unmanned systems (JAUS) standard—which is a messaging architecture [11]—for communicating with UAVs. If the robot controlling system is equipped with some automated features such as path planning, the UCC can send the user commands to use such features to correspond to the system units and then forward their responses to the IAIM to refresh the UI.
- **Context acquisition and integration (CAI)**: CAI is responsible for gathering and integrating all context information. Context information originates from different sources and has different types, which means this unit monitors all environmental sensors and subsequently gathers, integrates, and evaluates the information of interaction context elements (the user, environment, and system).
- **Intelligent adaptive interface management (IAIM)**: This unit is the heart of the proposed architecture and is responsible for two tasks:
  - Inferring and delivering right information at the right time
  - Configuring the adaptive user interface based on the knowledge at its disposal and acquired information of the interaction context.

Indeed, the main objective behind the development of IAIM is to minimize the demands on the user's time and attention and maximize the quality of the information supplied to him/her.

- **Knowledgebase, ontology, and database (KOD)**: This unit is responsible for storing and restoring knowledge (like reasoning and inferencing knowledge), ontology (the specific domain knowledge about domain concepts and their relationships), and information about all entities and their relationships in the interaction context. The KOD achieves the sixth feature of the proposed architecture.
- **Knowledge optimizer (KO)**: This unit edits and optimizes the ontology and knowledgebase by monitoring all the changes in the interaction context, and analyzing user behavior. In fact, the KO unit keeps the interaction system valid and updated. For example, this unit can

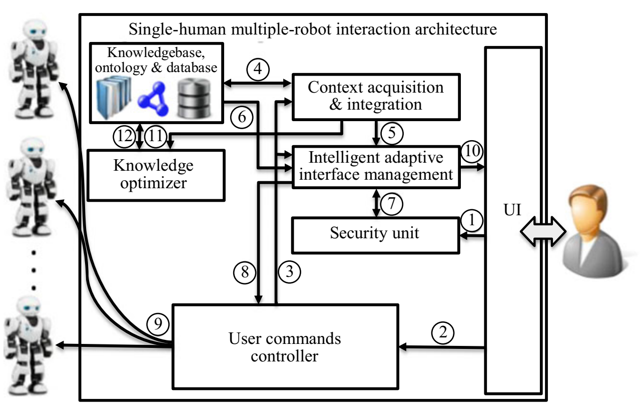

FIGURE 1 Proposed intelligent adaptive interaction (IAI) architecture for single-human multiple-robots interaction

change existing rules or create new rules to deliver right information to the user based on tracing the user's behavior and detecting specific needs of the user during the interaction. Hence, this unit realizes the fifth feature of the proposed architecture.

- **Security unit**: This unit, which has been neglected in most previous studies, is responsible for establishing a secure interaction between the human and robots. To achieve this goal, it comprises three main modules: authentication, authorization, and accounting (AAA), encryption, and steganography.

In Figure 1, the numbered relations indicate links between the above-mentioned units. These relations are explained below.

1. The user login information, commands, and other user generated data are sent to the security unit to pass the AAA processes.
2. All user commands that have been received by the user interface are delivered to the UCC unit.
3. A copy of all user commands is also forwarded to the CAI unit to monitor the user activities. In addition, a copy of the commands is sent to the IAIM unit to perform the following instructions:
   a. Update the user interface
   b. Perform security processes on these commands
   c. Send them to the UCC to transmit to the robots.
4. The CAI unit performs local inference using accessible knowledge in the KOD to gain new information about the interaction context. New recognized concepts, relationships, and newly learned rules from the interaction context will be stored on the KOD.
5. All data and information about the current context must be available to the IAIM unit to infer and detect the state of the interaction context and finally to configure the user interface.
6. The IAIM unit performs inference using stored knowledge and information available in the KOD unit.
7. The Security Unit sends security tokens such as the type of the user and user-accessibility level to the IAIM unit. The IAIM unit uses such tokens to configure the user interface. Thus, information is delivered to each user based on her/his privileges and authentication level. If the user decides to send and receive any secure data, all security requirements such as secret key agreement and parameter setting for either encryption or steganography modules are performed by the IAIM and subsequently passed to the security unit.
8. User commands, after considering security issues by the IAIM, are delivered to the UCC.
9. The UCC unit sends the received commands from the IAIM unit to the corresponding robot(s).
10. The UI generated by the IAIM, which has content and structure adapted perfectly to the context changes, is presented to the user.
11. All contextual information is delivered to the KO unit for analyzing, learning, and optimizing the knowledge-base.
12. All optimizations are applied on the knowledgebase and ontology by the KO unit. The knowledge required for the optimization is also provided by the KOD unit; therefore, this arrow is bidirectional.

3.1 | IAI architecture at level two

Each main unit of the proposed architecture has many subunits and details; thus it is considered as level two. The following sections characterize and describe these subunits.

3.1.1 | Context acquisition and integration

In our architecture, context information is divided into three states: user, situation, and system. Hence, three units were appointed to monitor and acquire information for these states. Each one of these units has some subunits to fulfill the responsibilities entrusted to them. The internal subunits of the CAI unit and their relationships are illustrated in Figure 2.

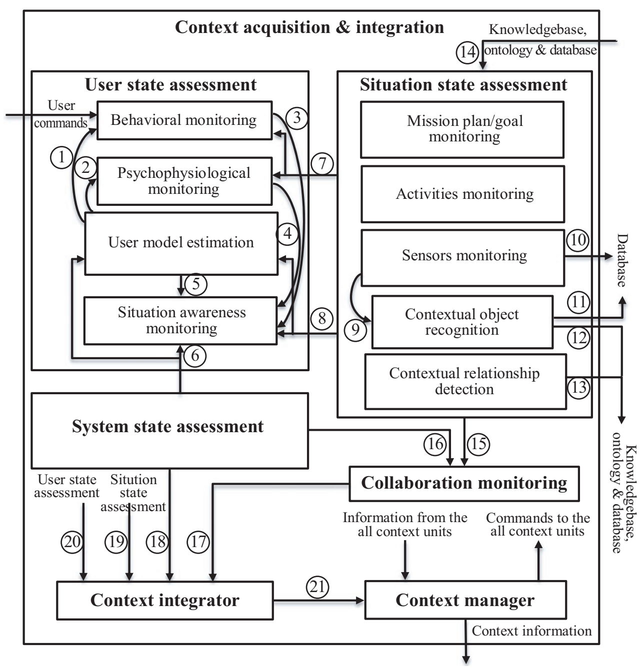

FIGURE 2 Internal components of the context acquisition and integration unit

- **Situation state assessment (SSA)**: This unit monitors and detects all changes and events that occur in the interaction environment. Thus, monitoring the status and all sensors of the under-command robots (as the main parts of the interaction environment) is one of the most important responsibilities of this unit. To handle this responsibility, the SSA unit engages four subunits: mission goal monitoring, activities monitoring, sensors monitoring, and contextual concept recognition. The first three subunits have been presented in a previous work [9] in a different manner. The tasks assigned to these subunits are expressed below.
  - **Mission goal monitoring**: This subunit monitors if the plans and goals and the sub-plans and sub-goals of the robots' mission are proceeding in the correct order.
  - **Activities monitoring**: This subunit monitors whether tasks and activities corresponding to the mission goals and plans are proceeding in the right order.
  - **Sensor monitoring**: This subunit acquires all environmental sensors information and detects events in the environment.
  - **Contextual concept recognition (CCR):** This subunit detects and recognizes new concepts in the interaction environment and saves these concepts into the ontology. Therefore, CCR is a means of turning raw sensor data and other sources of information into new contextual concepts. Thus, the ontology of the KOD unit can be updated at all times.
  - **Contextual relationship detection**: One of the most important issues in context awareness, is awareness about relationships between existing entities and concepts in the interaction context. For example, relationships such as friendship, hostility, and authority. These relationships are retained within the ontology. However, in the dynamic context, new entities and new relationships can be recognized. Thus, this unit is responsible for detecting new relationships between ontology concepts and modifying the ontology accordingly. For example, by inference on existing relationships between concepts, new relationships can be inferred (for example, the friend of our friend is our friend). This subunit and the contextual concept recognition subunit realize the eighth feature of the proposed architecture and help realize the fifth feature.
- **User state assessment**: In this unit, all aspects of the user personality and activities are monitored. It should be noted that user personality and activities play the most important role in the context of interaction. This unit is composed of four embedded subunits that have been introduced as follows:
  - **Behavioral monitoring**: All user commands and activities are captured and analyzed by this unit to verify user actions and to predict the future activities of the user. Thus, this unit responds to two main questions: "What is the user doing?" and "What will the user do in the near future?" Indeed, this subunit helps realize the second feature of the proposed architecture.
  - **Psychophysiological monitoring**: The main responsibilities of this unit are monitoring and detecting user emotions and attentions. Therefore, this unit gathers a variety of information in order to fulfill these objectives. For example, for emotion recognition, this unit needs to capture the user's facial images, gestures, and voice or speech. Thus, any real-time and accurate methods for emotion recognition such as [12] can be selected to implement the emotion monitoring part of this important unit. Further, for user attention detection, this unit captures and traces the user's eye gaze.
  - **User model estimation**: The user plays the main role in HRI. Therefore, we must monitor all behavioral and non-behavioral aspects of the user to realize an efficient interaction. One of the most popular methods for reaching this aim is user modeling. Therefore, this unit is responsible for estimating various types of user models, such as, cognitive model, control ability model, communication model, and interaction model. Further, one of the most important aspects in the user cognitive model is forgetting modeling, which can be used for delivering the right information at the right time to the right user. User cognitive model is used to answer very important questions about the user state. For example, the user cognitive model can be used to respond to this significant question: "What does the user know right now?" Thus, this subunit realizes the tenth feature of the proposed architecture.
  - **User situation awareness monitoring**: One of the most notable parameters for establishing efficient interaction is user situation awareness. It means that the higher the level of situation awareness, the more efficient is the interaction. Thus, we must trace and monitor the user situation awareness continuously and attempt to maintain user situation awareness at a high level during the interaction.
- **System state assessment**: Another important issue in the interaction process, which has been neglected in almost all previous works, is the interaction system state. Some of the important items in the interaction system are available hardware and software and their status, networks bandwidth, and latency time. The conditions of each one of these items can directly affect interaction quality. Therefore, we must monitor and assess the

interaction system state and then adapt the user interface based on this assessment.

- **Collaboration monitoring**: In the proposed architecture, the collaboration monitoring unit is responsible for monitoring collaboration between the robots during a given mission. Further, detecting the workload of each robot and reporting this information to the user (at the right time) is another responsibility of this unit, which has also been neglected in many other architectures. This subunit realizes the seventh feature of the proposed architecture.
- **Context integrator**: All environmental sensor data and acquired information about the interaction context are sent to this unit for integration. It means that this unit receives all data and information from other context acquisition units and produces an integrated and united view about the whole context information. Finally, context integrator delivers integrated information about context, in a well-defined structured format, to the context manager unit.
- **Context manager**: This unit is responsible for managing all requests that have been sent to the Context Acquisition and Integration unit. Thus, the Context Manager receives all requests about the context information and sends corresponding responses to applicators according to the CAI's subunits' answers.

The numbered relations between the subunits in Figure 2 are explained below.

1. The behavioral monitoring unit requires information about the user model, in order to fulfill its responsibilities. For example, by inferencing the user actions and somewhat the user knows it at a given time, this unit can predict that the user has decided to obtain details about a given object. Then, the behavioral monitoring unit sends this prediction to the IAIM to represent obtained details to the users, preemptively.

2. Psychophysiological monitoring unit needs the user models' information (for example, something that the user has forgotten, such as weather conditions) to identify the user attention more accurately.

3, 4, 5. Information that have been produced by the behavioral monitoring unit, psychophysiological unit, and user model estimation, are sent to the user situation awareness monitoring unit. This information is used by the inferencing process to detect the user situation awareness level.

6. The situation awareness monitoring unit and the user model estimation unit need system state information for assessing the user situation awareness and estimating the user cognitive model, respectively.

7, 8. All information about the situation state are needed for all the subunits of user state assessment. Therefore, this information is sent to those subunits via these links.

9. All gathered environmental sensor data are sent to the contextual concept recognition unit to recognize new concepts in the environment.

10. All captured environmental sensor data that should be saved (such as concepts properties) are stored into the database.

11. All information about new recognized concepts are saved into the database by the contextual concept recognition unit.

12, 13. All new recognized concepts and new detected relationships are stored within the ontology.

14. All five units in the situation state assessment need to be accessed to data, concepts, concepts' relationships, and knowledge that exist in the KOD.

15, 16. The collaboration monitoring unit needs to be informed about the situation and system state information to monitor collaboration conditions between robots.

17, 18, 19, 20. All acquired contextual information by the user state assessment unit, the situation state assessment unit, the system state assessment unit, and the collaboration monitoring unit are delivered to the context integrator unit.

21. The integrated contextual information is delivered to the context manager.

3.1.2 | Intelligent adaptive interface management

The heart of the proposed architecture is the IAIM unit that is mainly responsible for adopting the user interface based on the context changes and delivering the right information at the right time to the user. Thus, the main unit that realizes the first feature of the proposed architecture is IAIM. This unit has some subunits that are shown in Figure 3.

A brief description of the IAIM subunits, shown in Figure 3, is provided below.

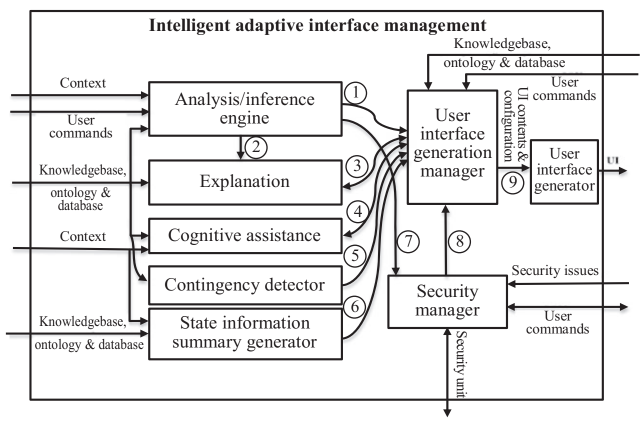

FIGURE 3 Subunits and inner details of the intelligent adaptive interface management unit

- **Analysis/Inference engine**: The heart of our proposed architecture is the IAIM and the core of the IAIM unit is the analysis/inference engine. This subunit performs inference on all contextual information to recognize changes in the interaction context. Then, based on this inference and the user command, this unit provides a pack of information to be sent to the user at the right time. The analysis/inference engine uses a user cognitive model (forgetting model) to prune the information pack. That is, those parts of information that the user remembers are removed from the information pack. As the result, the user cognitive load to process the information pack is reduced. Finally, the pruned information pack is sent to the user interface generation manager by the analysis/inference engine.
- **Explanation**: A significant feature of an intelligent system is its explanation ability. The explanation unit is responsible for providing necessary explanations about IAI behavior according to the user request. To do so, it uses the stored knowledge in KOD. For example, this explanation can be generated using rules that were used for inferencing by the analysis/inference engine. This unit realizes the fourth feature of the proposed architecture.
- **Cognitive assistance**: This subunit assists the user by providing information about happenings in the context and also makes some suggestions for possible solutions. These assistances and suggestions can lead to reduce the user's cognitive load. Thus, this unit also helps to realize the fourth feature of the proposed architecture.
- **Contingency detector**: This subunit is responsible for detecting contingency events in the near future. For example, collision with an obstacle, contingency behaviors, near future changes in the environment and goals are some of the contingency events. Therefore, all context information, data, and knowledge in the KOD are accessible for the contingency detector to accomplish its responsibilities. Thus, the contingency detector enables the user to act proactively in dealing with contingency events in the near future. A specific paper that has been focused on holistic contingency management was proposed in [13].
- **State information summary generator**: When the user controls a group of robots, he/she needs to detect unforeseen problems as soon as possible. Thus, a concise information about the overall state of each robots should be delivered to the user. Errors, events, and unusual behavior can be easily detected by the user using this information.
- **User interface generation manager (UIGM)**: This subunit handles adaptation issues related to the content of the user interface. Indeed, UIGM designs and configures the final user interface using the delivered information pack and guidelines from the analysis/inference engine, explanation, cognitive assistance, contingency detector, and state information summary generator. This subunit can design and configure any type of user interface such as, regular WIMP (windows, icons, menus, and pointer) or sophisticated natural user interface (NUI). UIGM describes the configuration of the user interface using a UI description language (UIDL) such as UsiXML [14,15]. It must be noted that by describing a user interface using the UIDLs makes UIGM technology independent. Another duty of this unit is managing user requests for explanation and assistance.
- **User interface generator**: The final user interface is generated by this subunit based on the UI contents and configuration that were proposed in form of a UIDLs by the user interface generation manager.
- **Security manager**: As mentioned before, security is very important issue in HRI. This subunit is designed to guarantee secure interaction. The security manager receives all security issues such as the user access rights from the security unit. It also receives user commands and the information pack from the analysis/inference engine. Then, the security manager sets all security requirements for encryption, decryption, and steganography (such as secret keys and other security algorithm parameters) based on security requirements. The result of this process is sent to the security unit to perform the required security processes on the information pack.

As shown in Figure 3, some numbered relations have been depicted as connections between mentioned IAIM's sub-units. In the following, some brief descriptions about these connections have been presented.

1. The information packs and some guidelines generated by the analysis/inference engine about how to display information packs are delivered to the user interface generation manager.
2. Information about employed rules and methods used in inferencing process is exchanged through this link. This information is used by the explanation subunit to generate on-demand explanations about the system behaviors.
3. User requests for explanation are sent to the explanation subunit via the user interface generation manager. Further, explanation subunit sends some explanations about system behavior to that subunit. Thus, this arrow is bidirectional.
4. Some assistants about current situation are prepared by the cognitive assistance subunit and sent to the user

interface generation manager, based on the user request. Thus, this arrow is also bidirectional.
5. The contingency detector subunit sends detected contingencies to the user interface generation manager to deliver to the user at the right time.
6. Summaries, which describe current state of interaction, are generated and delivered to the user interface generation manager, by the state information summary generator.
7. All information-packs prepared by the analysis/inference engine are sent to the security manager subunit to check for security issues and about presenting them to the user. Thus, continuous/active authentication also can be performed.
8. All information-packs that passed security checks and also (if necessary) passed decryption stages, are delivered to the user interface generation manager by the security manager subunit.
9. User interface content and its configuration that prepared by the user interface generation manager, are sent to the user interface generator subunit using a given UI description languages (UIDLs).

3.1.3 | Security unit

The security unit is responsible for all security duties. Thus, all security tasks such as AAA, encryption, decryption, and steganography are assigned to this unit. The subunits of the security unit have been illustrated in Figure 4.

The security unit cooperates with user state assessment unit (that provides the user model) to perform continuous/active authentication. Indeed, our proposed architecture provides suitable infrastructure for implementation of existing continuous/active authentication methods such as cognitive fingerprint that has been proposed by DARPA [16,17].

A general description about two main subunits of the security unit are provided below.

- **AAA unit**: This unit performs AAA on the user's login information, user commands, and user generated data.
- **Cryptography unit**: All encryption, decryption, and steganography tasks are assigned to this unit. This unit performs requested security processes by the IAIM unit on the delivered information, using security parameters that were set by the security manager.

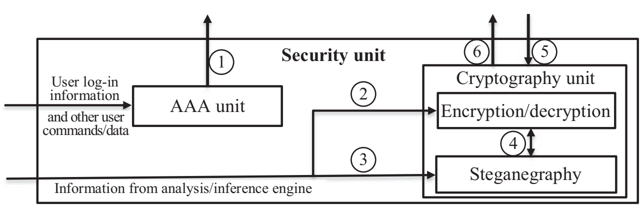

FIGURE 4 Subunits and internal details of the security unit

Some numbered relations that were shown in Figure 4, are introduced below.

1. The AAA Unit sends all security information about the user authentication and authorization to the IAIM unit in order to use this information to deliver right information to the right user.

2, 3. All information that should be encrypted are sent to this unit by the IAIM and will be delivered to related subunits.

4. Toward more secure information transmission, the Steganography subunit embeds encrypted information into images or videos. Further, we often need to encrypt or decrypt extracted data from images or videos, and therefore, the extracted data is forwarded to the encryption/decryption subunit. That is why this arrow is bidirectional.

5. All security requirements for encryption, decryption, and steganography that have been set by the security manager subunit, are sent to the cryptography unit.

6. All encrypted/decrypted information or extracted data from images or videos, are sent to the security manager subunit in order to send to the robots or show to the user.

3.1.4 | Knowledge, ontology, and database

Knowledge, ontology, and database consist of three components: the knowledgebase, ontology, and database.

The required knowledge for IAIM and CAI units are stored in the knowledgebase unit. These include knowledge for

1. Inferencing right information at the right time.
2. Contingency detection.
3. Cognitive assistance generation.
4. User/system/situation state assessment.
5. Collaboration monitoring.

For example, if fuzzy inference to be used as the inference method, fuzzy rules and fuzzification methods can be saved as fuzzy control language (FCL) file and stored into the KOD unit. Hence, the knowledge for different domains can be stored as separated FCL files.

On the other hand, the Ontology stores all concepts and their relationships in the interaction context. The Ontology can be defined using any ontology editor (like Protégé®) and stored in web ontology language (OWL) or resource description framework (RDF) file. Then, the existence of a concept or relationship can be checked by processing this file and new detected concepts and relationships can be inserted into the file. Finally, all sensory data that have been collected by environmental sensors will be stored into the general-purpose distributed database in the KOD. All KOD's subunits and their relationships are depicted in Figure 5.

The sensor/data fusion subunit is responsible for the integration of multiple raw data characterizing the same real-world object and properties into a consistent, accurate, and useful representation.

In Figure 5, some numbered relations have been shown, which are described below.

Data, knowledge, and ontology are available for all subunits of the IAIM and CAI. Further, these units have privileges to insert new data into the database.

2, 3. The KO unit changes and optimizes ontology and knowledgebase, so this unit has been authorized to change the ontology and the knowledgebase.

4. New concepts and relationships have been detected by the contextual concept detection and the contextual relationship detection, must be stored within the ontology. Therefore, this arrow links CAI to the Ontology subunit.

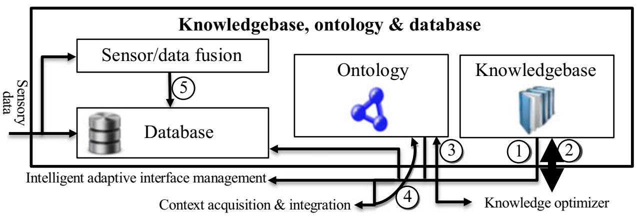

FIGURE 5 Subunits and internal details of knowledge, ontology, and database (KOD) unit

5. The sensor/data fusion subunit stores all fused data into the database.

So far, the proposed comprehensive architecture for interaction between human user and a group of robots was thoroughly explained.

4 | MODELING THE PROPOSED IAI ARCHITECTURE

In this section, a model of the proposed architecture is provided to responsd to this significant question: "Can the mentioned components of this architecture work together to meet the final goal?" Consequently, the colored petri net [18] was selected as a most popular and powerful mathematical modeling method to model our proposed interaction system. Afterward, we found that to precisely model the user behavior and events in the interaction context, we should utilize hierarchical timed colored petri net. Therefore, we modeled our proposed IAI architecture by hierarchical timed colored petri net using CPN tools [19,20]. The top level of this hierarchical model is illustrated in Figure 6.

For modeling user commands, seven types of commands were considered: robot commands, explanation, assistance, encryption, steganography, plain communication, and other commands. All of these commands are generated randomly and fed to the system, to test all types of user commands in modelling process. Afterwards these commands are sent to the user command controller. Then, this unit will send the commands to the IAIM unit or to

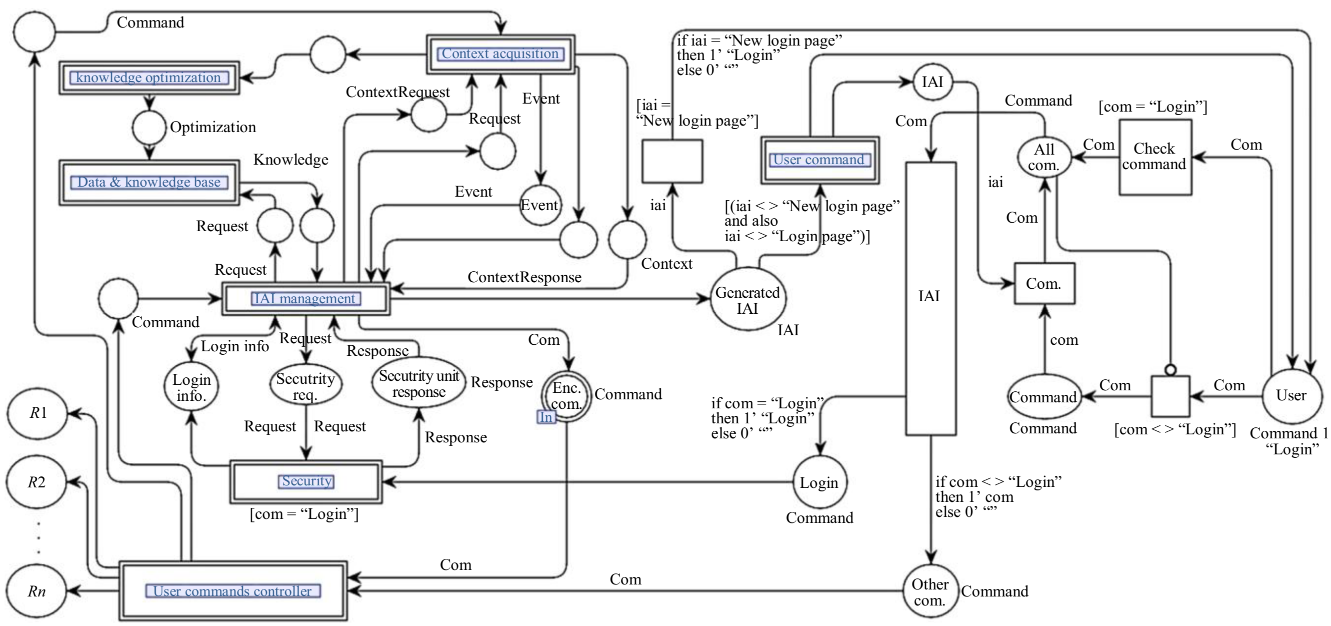

FIGURE 6 Top level of hierarchical timed colored petri net model of our proposed intelligent adaptive interaction (IAI) architecture

the robot(s) ($R1$, $R2$, … , Rn refer to robots' name in Figure 6).

Note that, if the user has already selected secure interaction, then the robot commands are encrypted or embedded into an image (or video) before sending to the robots. In fact, IAIM unit receives the robot commands from the user command controller and then sends these commands to the security unit via the security management subunit. Finally, encrypted commands will be delivered to the IAIM unit and sent to the user command controller unit.

If the user commands are not robot commands, then the IAIM manages execution of these commands. Further, IAIM adapts the user interface according to these commands and the context changes. Besides, the IAIM unit needs to be aware about all changes and events in the interaction context. Therefore, there are some connections between the IAIM and the context acquisition unit in Figure 6. Finally, all information about the interaction context are continuously delivered to the KO unit to optimize the knowledgebase and the ontology whenever necessary.

Each one of the main units, which have been shown in Figure 6, has one or more levels and many details in modeling. However, discussion about all of them is beyond the scope of this article. Nevertheless, for example, the details of the IAIM unit model in the third level have been demonstrated in Figure 7.

Finally, we ran the model by different parameters and captured simulation results to monitor model behavior. Our goals of this simulation are listed below

1. Checking whether the proposed model's components work together properly.
2. Defining inputs and outputs of each part of the proposed architecture more accurately.
3. Examining the model functionality considering simultaneous occurrence of multiple contextual events.
4. Inspecting occurrence of deadlock in simulated conditions.
5. Testing number of refreshes of UI based on different events occurrence frequencies.

To achieve the above goals, we considered two different conditions: low probable occurrences of events (all contextual event occurrences with probability per second = 0.01) and high probable occurrences of events (all contextual events occurrence probability per second = 0.9). Contextual events are all events that have been triggered by CAI's units. For example, events that have been generated by user state assessment, situation state assessment, and system state assessment units are contextual events. A number of monitors were defined for some selected places of the petri net model, owing to the report count of tokens in these places. Finally, the model was executed several times for each of the two conditions by firing the model transitions one million times in random order.

The results of modeling showed that in the case of probability per second = 0.01, the total number of the user commands and the IAI's refresh count were equal to 41,040 and 41,707, respectively. It is seen that the IAI's refresh count is greater than the total number of the user

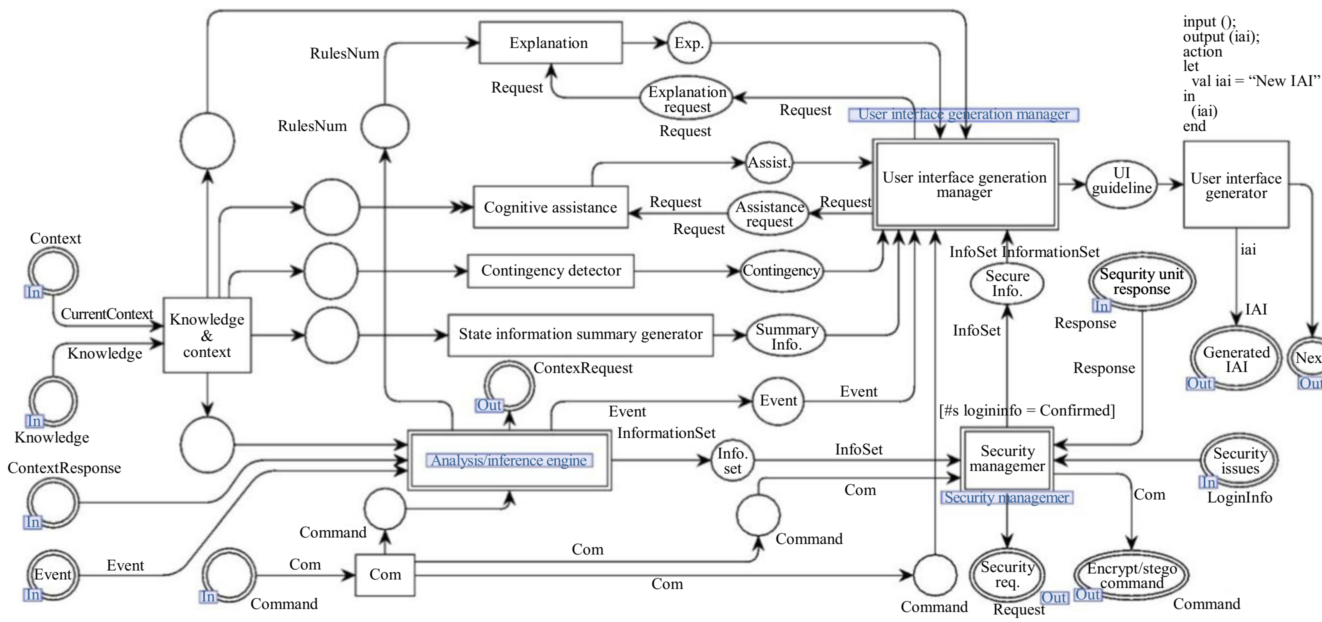

FIGURE 7 Intelligent adaptive interface management (IAIM) unit model in the third level of the intelligent adaptive interaction (IAI) architecture model

commands. This is because, by each user command, the IAI is refreshed. On the other hand, by each occurrence of event, the IAI may be refreshed (if needed). As a result, the IAI's refresh count (41,707) is less than the number of user commands plus the total number of occurred events (41,904). This is because there are some co-occurrence events causing one IAI refreshing. Therefore, this is the normal behavior of the system.

After running the model in case of contextual events probability per second = 0.9, it was observed that the maximum number of tokens in the place of the Displayed IAI, was equal to 8, while that in the case of low frequency of event occurrence this number was equal to 3. It shows that when the contextual events probability is increased, the proposed system responds to more events and refreshes the user interface accordingly. Thus, the user interface may be refreshed several times between two successive user commands. Using this modeling, we achieved all mentioned goals and tested the overall functionality of the proposed architecture.

At the end of this section, it must be noted that, one of the most important goals for modeling our IAI architecture using timed petri net is to prove the fact that our proposed architecture can react in real time to task between the user and a group of robots. Indeed, by this modeling, it is proven that if each main part of the architecture (CAI, IAIM, UCC, and security unit) can do their job in real time, then the system can react in real time. In other words, because we considered the components of these main units in this model as black boxes and because there are many real-time methods and algorithms to implement each one of the components, our modeling proves that the implemented IAI based on our architecture can react in real time to task between the user and the robots.

5 | SIMULATION AND RESULT

Now, the main question is whether the proposed IAI architecture can improve efficiency measurements in a real operation. To answer this question, we implemented an IAI based on our proposed architecture to control a group of UAVs. In this paper, to implement IAIM, we used a fuzzy inference system to infer the right information at the right time. For intelligent configuration of the IAI, we also employed fuzzy reasoning. Note that, fuzzy rules for each one of these works, have been defined and stored in separate FCL files in the KOD unit. For example, 81 fuzzy rules were defined to intelligent warning for inability to continue operations.

Further, we defined 25 fuzzy rules to infer appropriate luminance for each one of the UAVs' icons, based on their current backgrounds. In our IAI, the luminance of the UAV's icon is inferred every 0.5 s during the operation and will be immediately applied on the UAV's icon. Thus, in this implementation, to realize IAIM goals, we utilize fuzzy inference system.

In addition, to implement cognitive model of user model estimation subunit, we used our enhanced version of ACT-R, which has equipped with scale-invariant memory, perception, and learning (SIMPLE) [21] forgetting model. In this expansion of ACT-R, we equipped standard ACT-R with the SIMPLE forgetting model to model forgetting in short-term memory based-on the time and capacity of the short-term memory. Finally, we used this cognitive model to display the right information at the right time to the user based on the cognitive and forgetting model. Nevertheless, our implementation is very detailed and elaborate, and therefore, it is not covered in this article.

To evaluate the IAI, different simulated scenarios were performed by eleven contributors. Each contributor completed operations in two modes: IAI on; IAI off.

In the IAI-on mode, features (such as: intelligent warning to avoid collision, displaying important information based on the user cognitive and forgetting model, and intelligent warning for inability to continue operations) were enabled.

In IAI-off mode, IAI features were disabled (that is named classic interface).

The main screen of IAI has been shown in Figure 8. It should be noted that we used AnyLogic® simulation software, which is one of the most powerful ones, to implement our IAI and test it in simulated environment.

To test the proposed IAI in different levels of workload, we designed three scenarios:

1. Reconnaissance operation with three UAVs that is called "3 UAVs" operation.
2. Reconnaissance operation with five UAVs that is called "5 UAVs" operation.
3. Reconnaissance operation with five UAVs and with compulsion to cryptography in specific situation meanwhile the operation, which is called "5 UAVs with Encryption" operation.

To evaluate each one of these operations, objective and subjective measures have been considered, as stated below.

1. Objective measures:
    a. Situation awareness global assessment technique (SAGAT) [22].
    b. Mission completion percentage.
2. Subjective measures:
    a. NASA task load index (NASA-TLX) [23].
    b. Overall perceived situation awareness (SA) assessment using 7-Point Likert-type scale (from very-low or 1 to very-high or 7) questionnaire.

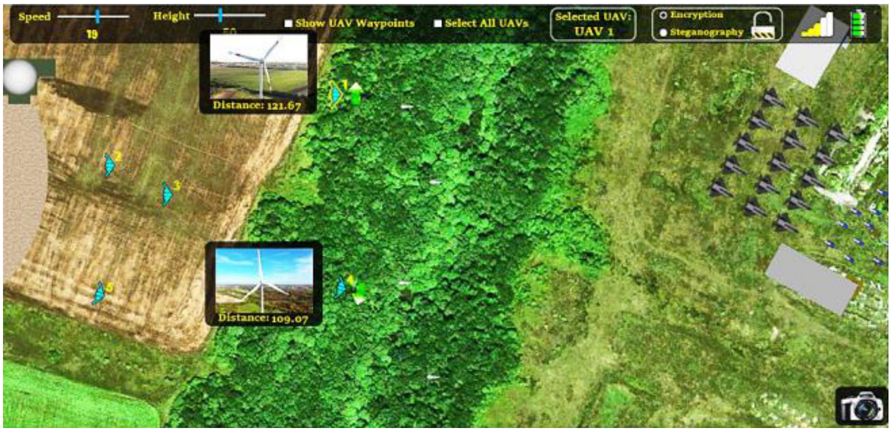

FIGURE 8 Overall view of the proposed intelligent adaptive interaction (IAI) for five unmanned aerial vehicles (UAVs). Selected UAV has been highlighted in yellow. Two simultaneously occurred collisions and their accordant temporal in-place UAV's camera displays at the back of the UAV's icons have been depicted

Note that, the four mentioned measures are standard and widely used in the literature.

As mentioned before, we designed three operation scenarios and each contributor completed the operations in two modes (IAI on and IAI off). Finally, all information about objective and subjective measures have been gathered for every six operations, and for each contributor. Figure 9 provides a statistical comparison between the obtained results for the SAGAT score that has been made by the box and whiskers diagrams.

As shown in Figure 9, it is obvious that the average of the SAGAT score (represented by × in the box) in operations have been performed using IAI is clearly greater than operations that have been completed by the classic interface. It shows that user's situation awareness level can be improved using proposed IAI based-on our architecture. Second objective measures to compare IAI and classic interface is mission completion percentage. Figure 10 shows a comparison between statistical results for mission completion percentages of all operations.

As depicted in Figure 10, the average of mission completion for operations that have been performed using IAI is significantly above that for operations that have been completed by the classic interface. Note that, the operations' scenarios were designed such that the contributors had to cancel reconnaissance operation (in the middle of the operation), for safely bringing UAVs to the base. Therefore, the max value for operation completion percentage is about 70%, and therefore, 50% mission completion is a good operation completion percentage.

NASA task load index and overall perceived SA were selected as subjective measures for gauging cognitive workload and perceived situation awareness, respectively. Obtained results for these subjective measures have been illustrated in Figures 11 and 12.

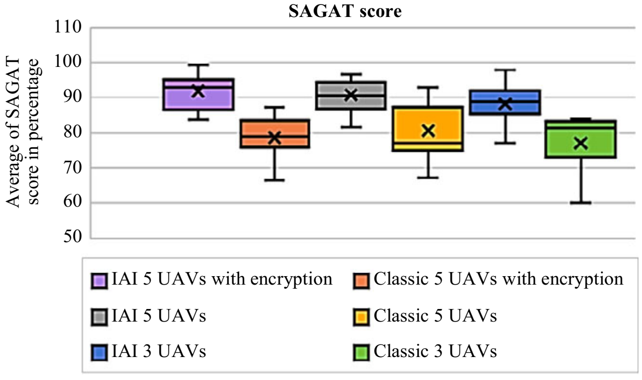

FIGURE 9 Box and whiskers diagrams for situation awareness global assessment technique (SAGAT) score. Operations that completed using interface that has intelligent features, were named as intelligent adaptive interaction (IAI) and otherwise were called as classic

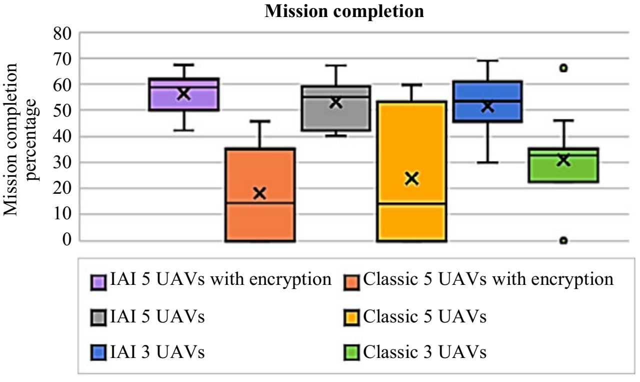

FIGURE 10 Box and whiskers diagrams for the mission completion percentage

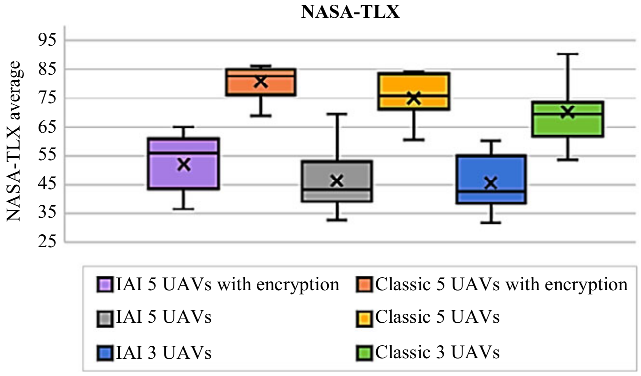

FIGURE 11 Box and whiskers diagrams for the NASA Task Load Index (NASA-TLX)

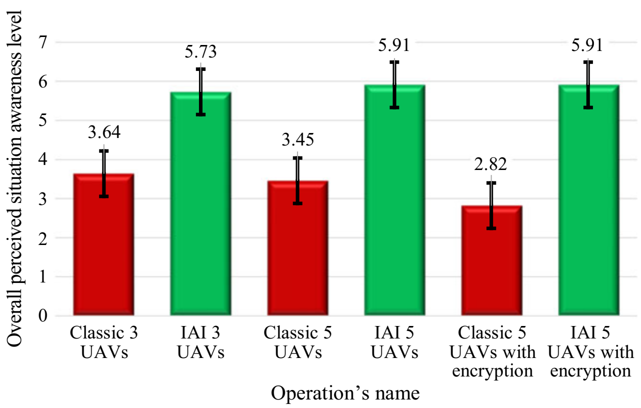

FIGURE 12 Comparison between averages of overall perceived situation awareness level calculated for all six operations

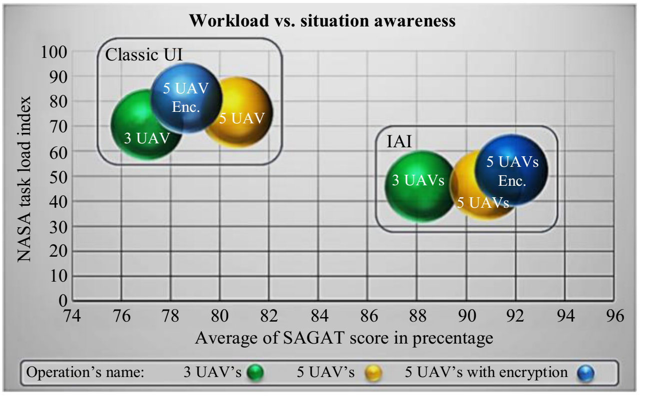

FIGURE 13 Reducing users' workload and simultaneously increasing users' situation awareness for all operations that have been done using intelligent adaptive interaction (IAI) against classic interface

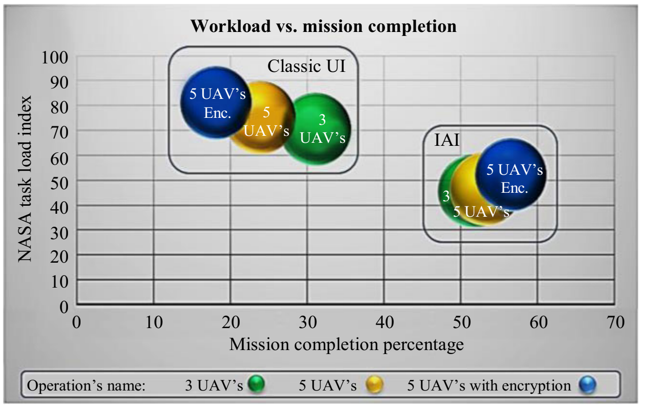

FIGURE 14 Decreasing users' workload that leads to increasing mission completion percentage for all operations that have been done using intelligent adaptive interaction (IAI) against classic interface

According to Figure 11, the averages value of NASA-TLX for classic interfaces are dramatically greater than IAI ones. It shows that more workload was imposed on contributors when they were doing the operations using classic interface. It should be noted that the highest value of the NASA-TLX is equal to 100 that shows the highest workload on user. Further, as shown in Figure 12, all users have stated that overall perceived situation awareness is significantly greater in operations that have been done using IAI in comparison with classic interface.

Two of the most important advantages of IAIs are reducing user workload and simultaneously increasing user situation awareness. For simultaneous comparison, we compared users' situation awareness that have been calculated by the SAGAT score and the users' workload that have been obtained by NASA-TLX for all operations. Figure 13 demonstrates this compression.

As previously emphasized, one of the most prominent goals of IAIs is reducing user workload to reach desirable and efficient interaction, which finally leads to increasing operation completion percentage. Figure 14 proves that we could decrease users' workload and simultaneously increases operation completion percentage in all operations.

At the end of this section, it must be noted that we cannot compare our results with any previous work because of two main reasons:

1. Each of the previous works used its own custom scenarios without mentioning the details. Therefore, we could not simulate the same scenarios with the same workload as the previous works' scenarios.
2. Not only are there no standard scenarios for testing intelligent adaptive interfaces in the field of multi-UAV controlling but also each one of the previous works tested its proposed IAI with different criterions and measures.

Nevertheless, for better understanding, we decided to compare our results with somewhat similar experiments. To do this, given the reasons for not being able to

TABLE 1 Brief introduction to some similar experiments

Experiment reference number	Number of under-command unmanned vehicles	Concise description
[24]	5	Situation awareness global assessment technique (SAGAT) and NASA task load index (NASA-TLX) have been calculated in different scenarios and in two modes: intelligent adaptive interaction (IAI) On and IAI Off
[25]	4	Cognitive workload (CW) and situation awareness (SA) have been evaluated by NASA-TLX and SAGAT in a transparent autonomy interface
[26]	3	A comparison has been made between CI (conventional interface) and PCI (predictive conventional interface) and PVRI (predictive virtual reality interface) using NASA-TLX and SAGAT
[27]	4	Adaptive console for supervisory control of multiple unmanned aerial vehicles (UAVs) has been evaluated by NASA-TLX and SAGAT
[28]	4	NASA-TLX and SAGAT have been used to evaluate a cognitive and cooperative assistant system for aerial manned-unmanned teaming missions

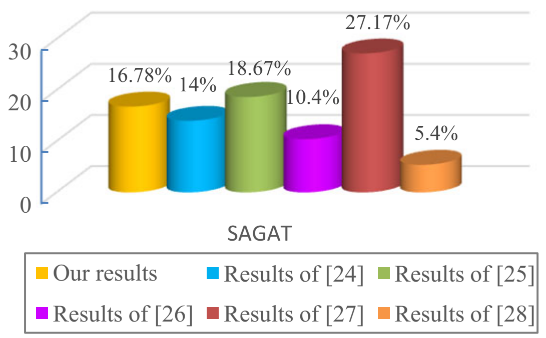

FIGURE 15 Comparison between situation awareness global assessment technique (SAGAT) percentage increase in our results and reported results of somewhat similar experiments

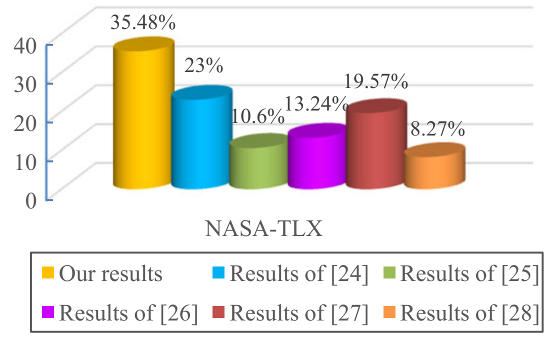

FIGURE 16 Comparison between NASA task load index (NASA-TLX) percentage decrease in our results and reported results of somewhat similar experiments

compare precisely, we considered percentage increase for situation awareness level and percentage decrease for user workload. It means that, we compare two works using their reported percentage increase for SAGAT and reported percentage decrease for NASA-TLX. Note that, all these experiments presented adaptive or intelligent interface (console) to control a group of unmanned vehicles not an IAI framework or architecture. Some concise descriptions for these experiments have been presented in Table 1.

As mentioned in Table 1, all these experiments have used SAGAT score to evaluate situation awareness level and NASA-TLX to assess user workload. A comparison between our percentage increase for SAGAT and the reported results of mentioned works for SAGAT percentage increase has been made in Figure 15.

Further, we compared our percentage decrease for NASA-TLX with the reported percentage decrease of the mentioned experiments. This comparison has been illustrated in Figure 16. Note that, we used our results for controlling five UAVs in all comparisons.

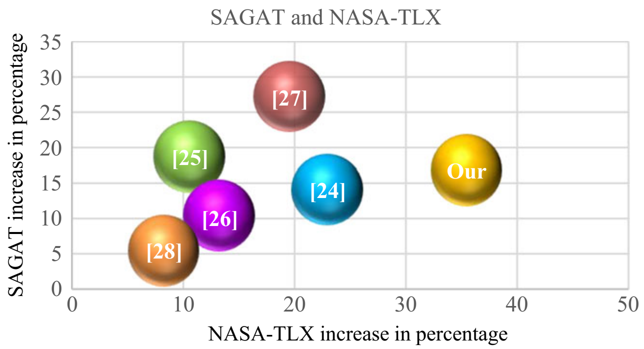

FIGURE 17 Simultaneous comparison between reported results for situation awareness global assessment technique (SAGAT) and NASA task load index (NASA-TLX)

Eventually, Figure 17 illustrates the comparison between the reported results for SAGAT and NASA-TLX and also determines the position of our results among other reported results.

As depicted in Figure 17, our result has a good balance in terms of increasing SAGAT and decreasing NASA-

TLX. Furthermore, in [24], it has been reported that the perceived SA for controlling five UAVs is equal to 5.3. Our result for this score for controlling five UAVs is equal to 5.91. Thereupon, apart from the fact that our IAI yields good results, also these comparisons indicate that our results are in a reasonable range reported in the literature.

6 | CONCLUSION

In this paper, we proposed a comprehensive architecture for IAI. This architecture provides a uniform framework for interaction between the user and robots. The focus of our design has been single-human multiple-robot interaction. The major assumption is that the human is stationary but robots are remote and mobile. The major features of our design are

- Delivering the right information at the right time
- Adaptation to all interaction's context changes
- Supporting user by cognitive assistance and system behavior explanation
- Self-configuration and optimization
- Using the ontology for modeling domain knowledge
- Providing secure interaction
- Preparing infrastructure for continuous/active authentication
- Involving user model with forgetting model in the interaction process.

We strongly believe that these features or requirements are important considerations for the development of real IAI, and that our proposed architecture is the correct framework response to these requirements.

To verify the functionality of our proposed architecture, we modeled our architecture using the hierarchical timed colored petri net. Obtained results of this modeling has shown that the components of our proposed design can work together perfectly and cover all our objectives.

Finally, to test our proposed architecture under real operational scenarios, we implemented an IAI based on our proposed architecture to control a group of UAVs. We defined three operation scenarios. These scenarios were executed in two modes: using IAI and using classic interface. Then, each one of these operations was completed by eleven contributors. The results showed that the proposed IAI based on our architecture could significantly increase users' situation awareness and also decrease users' workload and finally leads to promote mission completion percentage.

ORCID

Mahdi Ilbeygi http://orcid.org/0000-0003-1228-3194

REFERENCES

1. R. G. Hanumansetty, Model based approach for context aware and adaptive user interface generation, Ph.D. Dissertation, Computer Science, Virginia Tech, Virginia, 2004.
2. M. A. Goodrich and A. C. Schultz, *Human-robot interaction: A survey*, Found. Trends® Human-Computer Interact. **1** (2007), no. 3, 203–275.
3. B. Larochelle et al., Establishing human situation awareness using a multi-modal operator control unit in an urban search and rescue human-robot team, *Proc. IEEE Int. Work. Robot Hum. Interact. Commun.*, Atlanta, GA, USA, July 31-Aug. 3, 2011, pp. 229–234.
4. J. M. Riley et al., *Situation awareness in human-robot interaction: Challenges and user interface requirements*, Human-Robot Interact. Futur. Mil. Oper. CRC Press, Burlington, VT, USA, 2010, pp. 171–192.
5. M. Hou et al., Advances and challenges in intelligent adaptive interface design, *Human–Machine Syst. Des.*, Wiley, Hoboken, NJ, USA, 2015, pp. 369–424.
6. M. Hou et al., *Optimizing operator–agent interaction in intelligent adaptive interface design: A conceptual framework*, IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. **41** (2011), no. 2, 161–178.
7. F. Fortmann and A. Lüdtke, An intelligent SA-adaptive interface to aid supervisory control of a UAV swarm, *IEEE Int. Conf. Ind. Informat.*, Bochum, Germany, July 29-31, 2013, pp. 768–773.
8. D. Donath, A. Rauschert, and A. Schulte, Cognitive assistant system concept for multi-UAV guidance using human operator behaviour models, *Humous'10*, Toulouse, France, Apr. 26-27, 2010.
9. G. R. Arrabito et al., *Human factors issues for controlling uninhabited aerial vehicles: Preliminary findings in support of the Canadian forces joint unmanned aerial vehicle surveillance target acquisition system project*, Technical Report 2009-043, Defence R&D Canada, Toronto, CA, 2010, available at http://pubs.drdc.gc.ca.
10. P. A. Akiki, A. K. Bandara, and Y. Yu, *Adaptive model-driven user interface development systems*, ACM Comput. Surv. **47** (2014), no. 1, 9:1-9:33.
11. S. Rowe and C. R. Wagner, *An introduction to the joint architecture for unmanned systems (JAUS)*, Ann Arbor **1001** (2008), 48108.
12. M. Ilbeygi and H. Shah-Hosseini, *A novel fuzzy facial expression recognition system based on facial feature extraction from color face images*, Eng. Appl. Artif. Intell. **25** (2012), no. 1, 130–146.
13. J. L. Franke et al., Holistic contingency management for autonomous unmanned systems, *Proc. AUVSI's Unmanned Syst. North Am.*, 2006.
14. Q. Limbourg, USIXML: A user interface description language supporting multiple levels of independence, Eng.Adv.Web Applicat.: Proc. Workshops connection Int. Conf.Web Eng. (ICWE 2004), Munich, Germany, July 28-30, 2004, pp. 325–338.
15. J. Guerrero-García et al., A theoretical survey of user interface description languages: Preliminary results, *Latin American Web Cong.- Joint LA-WEB/CLIHC Conf.*, Merida, Mexico, Nov. 9-11, 2009, pp. 36–43.
16. R. P. Guidorizzi, SpringerSecurity: Active authentication, *Soc. Robot.*, pp. 452–459.
17. M. Abramson, Cognitive fingerprints, *AAAI Spring Symp. Series*, Palo Alto, CA, USA, Mar. 23-25, 2015.

18. K. Jensen and G. Rozenberg, *High-level Petri nets: theory and application*, Springer Science & Business Media, Berlin Heidelberg, 2012.
19. K. Jensen, L. M. Kristensen, and L. Wells, *Coloured Petri nets and CPN tools for modelling and validation of concurrent systems*, Int. J. Softw. Tools Technol. Transf. **9** (2007), no. 3–4, 213–254.
20. K. Jensen and L. M. Kristensen, *Colored Petri nets: A graphical language for formal modeling and validation of concurrent systems*, Commun. ACM **58** (2015), no. 6, 61–70.
21. G. D. A. Brown, I. Neath, and N. Chater, *A temporal ratio model of memory*, Psychol. Rev. **114** (2007), no. 3, 539–576.
22. M. R. Endsley, *Measurement of situation awareness in dynamic systems*, Hum. Factors **37** (1995), no. 1, 65–84.
23. S. G. Hart, *NASA-task load index (NASA-TLX); 20 years later,* Hum. Factors Ergon. Soc. Annu. Meting, **50** (2006), no. 904–908.
24. M. Hou and R. D. Kobierski, Intelligent adaptive interfaces: summary report on design, development, and evaluation of intelligent adaptive interfaces for the control of multiple UAVs from an airborne platform, DRDC-TORONTO-TR-2006-292-Technical Report, DRDC, Tornto, Canada (2006). 2006.
25. T. Chen, Management of multiple heterogeneous unmanned aerial vehicles through transparency capability multiple heterogeneous unmanned aerial vehicles through transparency capability, Ph.D. Dissertation, Queensland University of Technology, Australia, 2016.
26. J. J. Roldán et al., *Multi-robot interfaces and operator situational awareness: Study of the impact of immersion and prediction*, Sensors **17** (2017), no. 8, 1720:1-1720:25.
27. C. Fuchs et al., Adaptive consoles for supervisory control of multiple unmanned aerial vehicles, *Int. Conf. Human-Computer Interact.*, vol. **8007**, Springer, Berlin, Heildelberg, 2013, pp. 678–687.
28. A. Rauschert and A. Schulte, *Cognitive and cooperative assistant system for aerial manned-unmanned teaming missions*, NATO Res. Technol. Agency, Hum. Factors Med. Panel, Task Gr. HFM-170 Superv. Control Mult. Uninhabited Syst. Methodol. Enabling Oper. Interface Technol. RTO-TR-HFM **170** (2012), 1–16.

AUTHOR BIOGRAPHIES

Mahdi Ilbeygi was born in Tehran, Iran in 1984. He received his B.Eng. degree in computer software engineering from the University of Imam Reza, Mashhad, Islamic Rep. of Iran, in 2007, his M.Sc. degree in artificial intelligence from the University of Shahid Beheshti, Tehran, Islamic Rep. of Iran, in 2010. He is currently a Ph.D. candidate in artificial intelligence at the department of computer engineering of Iran University of Science and Technology, Tehran, Islamic Rep. of Iran. His main areas of research interest are human computer/robot interaction, image processing, natural language processing, ontology-based applications, and big-data analytic software.

Mohammad Reza Kangavari received his BS degree in mathematics and computer science from the Sharif University of Technology in 1982, the MS degree in computer science from Salford University in 1989, and the PhD degree in computer science from the University of Manchester in 1994. He is currently a lecturer in the Computer Engineering Department, Iran University of Science and Technology, Tehran, Iran. His research interests include data mining, machine learning, and natural language processing.