



HAL
open science

State of the Art on Computer-Supported Collaborative Work (CSCW), Situation Awareness (SA) and Human-Automation Teaming (HAT) for the SESAR ER2 AEON (Advanced Engine-Off Navigation) Project

Stéphane Conversy, Jérémie Garcia, Dong-Bach Vo

► To cite this version:

Stéphane Conversy, Jérémie Garcia, Dong-Bach Vo. State of the Art on Computer-Supported Collaborative Work (CSCW), Situation Awareness (SA) and Human-Automation Teaming (HAT) for the SESAR ER2 AEON (Advanced Engine-Off Navigation) Project. Ecole Nationale de l'Aviation Civile. 2021. hal-03930640

HAL Id: hal-03930640

<https://hal.science/hal-03930640v1>

Submitted on 9 Jan 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



D1.3 State of the art

Deliverable ID:	D1.3
Dissemination Level:	PU
Project Acronym:	AEON
Grant:	892869
Call:	H2020-SESAR-2019-2
Topic:	Innovation in Airport Operation
Consortium Coordinator:	ENAC
Edition date:	28 April 2021
Edition:	01.00.00
Template Edition:	02.00.02

Founding Members



EUROPEAN UNION

EUROCONTROL



2 State-of-the-art on collaborative human-machine interactions

In this section, we describe the state-of-the-art regarding collaborative interactions between human operators and automated agents. We start by describing the field of **Computer-Supported Collaborative Work (CSCW)** and presenting its main dimensions that we will use to analyse existing technologies and approaches relevant for the AEON project. We then introduce and discuss the **Situation Awareness (SA)** of all stakeholders so that they can maintain an appropriate mental image of the situation and cooperate to achieve the best possible results. We cover aspects related to **Distributed Situation Awareness (DSA)** that seem particularly relevant given the AEON distributed context to inform our future designs. Finally, in addition to collaboration with humans, operators will rely on partly and fully automated agents such as autonomous tugs, to move aircrafts from parking slots to runways holding points or algorithms, to find optimal solutions. We thus present work focused on **Human-Automation Teaming (HAT)** to identify relevant work related to Human and Automation Collaboration.

2.1 CSCW and Groupware

Computer-Supported Collaborative Work (CSCW) is a subdomain of the science of Human-Computer Interaction (HCI) concerned with the support to all activities involving more than one person.

Collaboration is the umbrella word to describe all aspects related to group activities: communication, coordination, production, edition etc. CSCW is concerned with the technical aspects, but also the social aspects: studies and theories on users' practices and their relationships emerging from their use of CSCW technologies.

"Groupware" is a synonym to "CSCW systems". Groupware are "computer-based systems that support groups of people engaged in a common task (or goal) and that provide an interface to a shared environment" [1]. Groupware can involve software, hardware, services and group process support [2]. General groupware examples include e-mail, chat, audio-video conferencing, mailing lists, discussion groups, group agenda, workflow system, shared editors (including in web browsers), wiki website, argumentations tools, roomware, collaborative buildings, source code management or mediaspaces. In ATC, groupware examples include radio/frequency through which controllers and pilots exchange information, telephones, airport operations centre, radar images through which a pair of controllers, collocated or distant, collaborate, and strip boards where they still exist.

Like single-user tools, groupware should provide a high level of usability (efficiency, efficacy and satisfaction [3] for production. But groupware should also provide the users with interactions that make *collaboration* between users usable.

2.1.1 Group awareness

The main characteristic of CSCW systems is that they should make their users aware that they belong to a group [4]. This need for "group awareness" contrasts with other multi-users systems such as database management systems or distributed operating systems where the goals are rather to hide as

much as possible that multiple users access the shared assets at the same time and ensure the integrity of the shared assets. Figure 3 includes categories of information that are important for group awareness, and questions that a user needs to answer when using a groupware. Group Awareness is to be differentiated from Situation Awareness: group awareness is the specific awareness of others' work through the Human-Machine Interface (HMI), while Situation Awareness is a broader concept that encompasses the whole activity.

Category	Element	Specific questions	Category	Element	Specific questions
Who	Presence	Is anyone in the workspace?	How	Action history	How did that operation happen?
	Identity	Who is participating? Who is that?		Artifact history	How did this artifact come to be in this state?
	Authorship	Who is doing that?	When	Event history	When did that event happen?
What	Action	What are they doing?	Who (past)	Presence history	Who was here, and when?
	Intention	What goal is that action part of?	Where (past)	Location history	Where has a person been?
	Artifact	What object are they working on?	What (past)	Action history	What has a person been doing?
Where	Location	Where are they working?			
	Gaze	Where are they looking?			
	View	Where can they see?			
	Reach	Where can they reach?			

Figure 3: Elements of workspace awareness relating to the present (left) and to the past (right) (from [4]).

2.1.2 Interactive services and concepts for collaboration

The vocabulary used in CSCW systems reflect the shift of concerns compared to single-user systems: 'users' become 'participants', they enter and leave 'sessions' of collaboration instead of just using a tool, etc. The following is an ontology of CSCW important services and concepts, with examples in general applications and in ATC.

Group window. A group window in HMIs is a collection of windows whose instances appear on different display surface.

Telepointer. A telepointer is a cursor that appears on more than one display and that can be moved by different users [4], [5]. When it is moved on one display, it moves on all displays. Although seldomly used in ATC systems, presumably because of implementation difficulties, telepointers are an important means to convey information, notably through gestures.

View. A view is a visual, or multimedia representation of some portion of a shared context. In the En-route control room, the two radar images are views on the same shared context, with different pan & zoom configuration.

Synchronous and asynchronous interactions. In synchronous interactions, such as spoken conversations, people interact in real time. Asynchronous interactions are those in which people interact over an extended period, such as in postal correspondence. In ATC, controllers talk with each other in real-time, and controllers and pilots talk mostly in real-time (depending on the load of the frequency). By contrast, Controller-Pilot Data Link Communications CPDLC enables controllers and pilots to communicate more asynchronously.

Session. A session is a period of synchronous interaction supported by a groupware system. Examples include formal meetings and informal work group discussions. A peculiarity of ATC (and some, but not all CSCW systems) is that it is always on, and all users enter and leave an already-running system. New flights constantly enter airspaces and leave them. ATC controllers work during 'shifts' (around 2-hour

long sessions) and either leave the session without being replaced (sector closure), or being replaced which implies some coordination with the newly arriving controllers.

Change. With an always-on system, users continuously change its content, including while a particular user has left the system. When this user enters the system again (after a few seconds, minutes or hours), s/he might want to be aware and understand the changes brought by other users while s/he was offline [6]. Some visualization allows such a user to see all changes (or the history) [7], [8], or to compare the version s/he knew when s/he left with the new version of the system. The visualization of changes may have different levels of granularity (e.g., for a text editor, a character, a word; it might even be a whole sentence or a paragraph). Visualization of changes can also prove useful in the same session, to remind that a flight got a particular clearance (e.g., level or heading clearance on a strip).

Comments and annotations. Besides the artefact being produced, secondary notations such as comments or annotations help users explain, discuss on and coordinate about the evolution of the artefact. In ATC, a service like a 'Warning' on an image radar or a strip is such an annotation, even if it's an annotation for oneself to be reminded in the near future (the oneself of the future can be considered as a teammate for some CSCW services). Such an annotation is also of help for the teammate who is going to replace the ATCO at the end of the shift (e.g., he writing the actual time of usage of a military area in use in some ACCs).

Conceptual model. A conceptual model is a set of relevant abstractions and their relationships that help humans use a system and predict its behaviour. With CSCW systems, the relevant abstractions are not only those that concern the content (e.g., a text, a drawing), but also those that concern the changes: a *modification*, a *set of modifications*, a *commit* (which is the act of a user to commit her/his changes on a 'working'/temporary version of the content), a *push/pull* (synchronize her/his commits to a shared version with other users), a *conflict*, a *merge*. Most of ATC systems are synchronous, and do not involve such a conceptual model. However, some of them do: ATCOs may prepare a number of clearances (i.e., a set modifications), and perform/send them later with a CPDLC system (commit + push); on a paper strip, the set of handwritten clearances is a visualization of the history that may prove useful to remind a particular context for a flight [9].

Implicit and explicit synchronization. When editing a document, the changes can be implicitly pushed to a shared version, usually as fast as possible to allow other participants to benefit from the most up-to-date version, and thus prevent possible conflicts. However, such a synchronization scheme prevents a participant from preparing (and iterating/designing) a change and committing a final, clean version of the changes. Besides, such synchronization scheme is only available with a living network connection and cannot be performed offline. By contrast, explicit synchronization enables users to decide when their version and the other participants' versions should synchronize. This enables users to change their mind during the preparation, and even for different users to participate to the preparation and the execution of a clearance [10]. However, a user should not forget to explicitly synchronize, and repeating an explicit synchronization interaction can be cumbersome.

Conflicts, concurrency, consistency, lock. As users change the same shared content, they might perform conflicting changes. Some systems detect such conflicts and present them to the users to request their resolution. Some systems are able to automatically resolve a conflict and merge two sets of modifications. To prevent conflicts, some systems force users to lock (either implicitly or explicitly) a subpart of the content. These are 'pessimistic' algorithms since they prepare for the worst to happen. The problem is that the lock mechanism increases the amount of latency between the action performed by a user and its effect on the screen, which results in poor usability. By contrast, optimistic algorithms locally perform as fast as possible the action of a user, before trying to synchronize the

changes. In the event of a conflict, various algorithms (including automatic undo/redo, composition of actions, operational transforms) can resolve the conflict. Optimistic algorithms rely on the fact that conflicts will be rare: if the system provides an efficient HMI that maximizes group awareness, social protocols between users will take place to avoid conflict, before it's necessary for technical protocols to resolve them. In ATC, procedures and role prerogatives prevent conflicts from occurring on a CWP.

Role. A role is a set of privileges and responsibilities attributed to a person, or sometimes to a system module. In ATC, controllers have specific roles and rights. Roles and rights might not be enforced by the system, but rather by established social protocols. By contrast, single-user devices (e.g., a mouse) may raise artificial boundaries eventually leading to rigid roles to enforce an exclusive access to the device. Some systems tentatively break those boundaries, by offering interactions to better distribute micro-tasks among ATCOs [11].

Participants management. Since users may enter and leave sessions, and may have different roles and rights, managers of a system might rely on specific services to add users, change their roles and rights, invite or retire them, or regulate their contributions. For example, a manager may rely on a system to manage sectors and distribute controllers on CWPs.

Feedthrough. Feedback is the immediate, perceivable response of an interactive application upon a user action. Feedthrough is a word derived from feedback. Where an artefact is shared, that artefact is not only the subject of communication, but it can also become a medium of communication. As one participant acts upon the artefact, the other observes the effects of the action. This observation by the other participants is called feedthrough [12]. Feedthrough and feedback can be similar, but also different: for example, in a graphical editor, the immediate feedback of a rubber rectangle helps a user control the resizing of an image, but a single, static, 'resizing' icon in other participants' views provides information that an image is currently being resized, without the visual disturbance that would occur when the rectangle continuously resizes itself. In ATC, while a previous sector has transferred a flight and the receiving sector is waiting for the pilot to call, in the receiving sector the label of the aircraft changes its colour and blinks. Colour blinking is at the same time a feedthrough that the aircraft has already been transferred from the previous sector and a highlight to quickly identify the aircraft when its pilot calls.

2.1.3 Taxonomy of CSCW systems and application to AEON

CSCW systems can be described along several dimensions of analysis. The following identifies several relevant dimensions from related work to understand and elicit AEON requirements.

The "clover" model of groupware

Groupware can be analysed according to the type of collaboration services they offer to their users. The "clover" model of groupware [13], [14] (derived from [15]) defines three spaces:

- Coordination: coordinating the production of a shared artefact by multiple users, with respect to time, scheduling, roles, and tasks involved;
- Production: manipulating physical or computerized entities and produce an outcome of the collaboration;

- Communication: exchanging information among users.

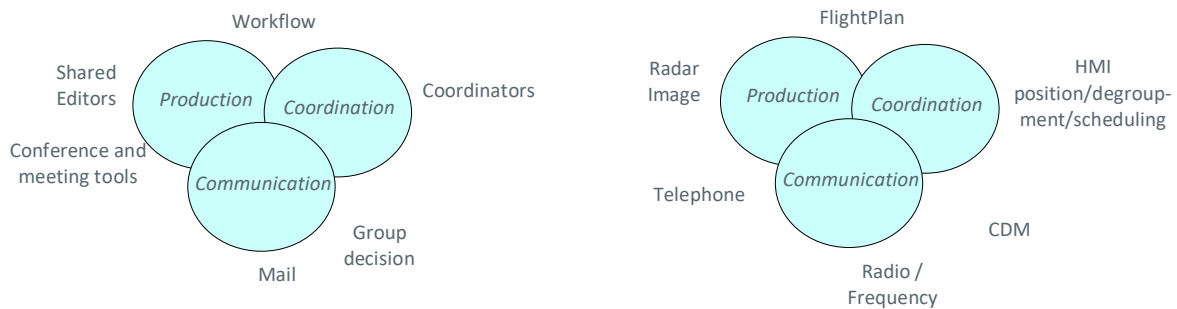


Figure 4: Clover model of groupware

A particular groupware participates with varying degrees to each dimension. For example, Email is mostly about Communication, but can contain some coordination aspects (e.g., “I’ll do that, while you do this”) and even production (e.g., sending a paragraph to be integrated in a shared document). A shared text editor is mostly about Production, but can also contain services for Coordination (e.g., reaching an agreement between users using the ‘comment’ service) or even Communication (e.g., a user can perform deictic gestures with a telepointer in a synchronous editor to communicate the part on the document s/he is talking about).

ATC systems contribute to each three types of space (see Figure 2). Within AEON, it is envisioned that services will likely contribute to communication and coordination, but also production e.g., of clearances.

Space-Time Matrix

Groupware can also be analysed according to the time at which the users interact through the system, and according to their relative physical location [1]. Bringing both dimensions together forms a space-time matrix as presented in Figure 3.

	Same time	Different time
Same place	Face-to-face interaction	Asynchronous interaction
Different place	Synchronous distributed interaction	Asynchronous distributed interaction

Table 2: Space-Time Matrix (from [1])

	Same time	Different time
Same place	TWR position	Shift
Different place	Positions TWR/LCK/towing vehicles	Asynchronous distributed interaction

Table 3: Space-Time Matrix applied to ATC (from [1])

ATC systems range into all categories, though very distant one (inter-continent) might not necessitate much interaction per se. For AEON, we expect to design spatially distributed interactions devices/surfaces, at least belonging to different roles or entities. Regarding time, we envision that interactions would be synchronous as when communicating via the radio, or asynchronous while ATCOs or airlines operations crews explore specific solutions on their own or wait for approval before pursuing. It is possible that ATCOs may attempt to plan say, one hour ahead, but there will be a residue of problems, some of which may require action within seconds. The relatively long-term (or upstream) process is often called ‘planning’ ATC and the short-term (or downstream) process ‘tactical’ ATC [16]. For AEON, we expect to design for both strategic and tactical tasks.

Even if the space-time matrix informs on the category and the difference between systems (e.g. single display groupware [17] are same-time/same-space system), it is deemed as “not very helpful” [18], as space might not be an issue at all if the collaboration support is adequate. It also misses important details due to its coarse granularity and due to it not taking account dynamicity: some interactions might begin asynchronously, eventually turning synchronous (e.g., a chat exchange turning into a voice call when the matter being discussed necessitates a modality with a higher throughput).

Communication

	one	many
one	ATCO => ATCO, ATCO => 1 pilot	ATCO => pilots
many	Pilots => ATCO	

Table 4: Communication Matrix applied to ATC (from [1])

Communication may involve one/many emitters and one/many listeners. In ATC, a single controller is in charge of communicating clearances with each pilot, while a pilot informs the controller on the flight situation. Communication patterns include clearance, request/answer information, or non-requested information by the pilot (e.g., turbulences). The frequency is multiplexed with multiple, short exchanges between different pairs of controller/pilot. Since a frequency is shared by multiple controllers and pilots, exchanges with a particular controller/pilot pair can be listened to by other controllers and pilots and inform on the global situation. In AEON, tugs operators may benefit from this situation awareness.

As discussed earlier, a single ‘communication’ might use multiple modalities according to the changing needs of the activity. This phenomenon is captured by the Multi-scale communication framework [19], and requires that communication systems should support a variable degree of engagement, smooth transitions between degrees and smooth integration with other media or communication systems.

Number of participants

The ATC system involves many participants, but for a project such as AEON the total number of participants is less meaningful than the peak number during a day, or than the patterns of collaboration they are involved in. The set of participants and its number evolve with the traffic (pilots leave and leave ‘sessions’), the time of the day and the traffic load (more controllers reinforcing teammates), and the weather (more controllers might be involved in case of low-visibility conditions or storms). At the scale of the AEON project, we envision that most direct interactions will involve less than 5 people, far from the hundreds that some systems like source code repositories can manage.

2.1.4 Requirements, Design Principles and Evaluation of Groupware

As with any other system, the methodology of HCI system design includes the elicitation of requirements and means to evaluate the product being designed. A peculiarity of HCI system design is the notion of “Design principles” that guide the design to make the product consistent and usable.

Requirements for groupware

After a period of discovery and development of new systems, some researchers gathered together a set of requirements for a platform to effectively support collaboration [18]. A CSCW platform should support:

- informal interaction and provide facilities for unrestricted interaction among users in the form of channels of direct communication. (e.g., a conference function)
- information sharing and exchange, and should enable users to:
 - exchange (send and receive copies of) information objects created by any application
 - share an information object, i.e., to make a particular object (e.g., a file) accessible to other users
 - work in close interaction by giving other users access a particular application window
 - fluidly transition between individual and cooperative activities (the user should be able to publish a file, send a message, display a window, etc. to another user just as he or she can print it from an ordinary workstation.)
- decision making
- coordination and control protocols
- domain directories

Design Principles for groupware

A number of research projects have tackled the problem of designing a digital system that can be updated, while preserving collaboration. They use design principles that guide the choices and the rationale behind the choices.

Use of touch screens

Some research ATC systems rely on the use of one large surface [11] or make use of two touch screens (one per controller) [20]. Their designers argue that touch screens are appropriate tools to support collaboration:

- they increase mutual awareness. Since touch screen-based HMI involve direct manipulation and gestures, seeing what a colleague is doing with his hand (directly or in peripheral vision) on a touch screen provides many cues on his activity.
- unlike mice, touch screens are shareable in a fluid manner: a user can interact on his touch screen as well as on his teammate's.

Free layout and close surfaces

The digitalization of paper strips led to HMIs that resemble actual strip boards. However, for the HMIs to actually convey all information conveyed by stripboards, they also have to mimic the ability of actual strip boards to freely lay out the electronic strips. For example, a planning controller may slightly shift or rotate a strip to the left to make it salient for the tactical controller.

As an alternative to replacing paper flight strips with digital systems, paper strips can be augmented with computing functions. Mackay et al describe how augmented paper strips can provide information to the system, while maintaining paper strips' properties and users' habits [21].

As shown in [22], subtleties in settings can greatly improve collaboration. In an experiment for a new control tool [23], experimenters noticed that a pair of controllers collaborated more when the two radar screens were made closer to one another, and oriented slightly towards the other as opposed to strictly facing the two controllers. Though users could interact with the teammate's screen in DigiStrips, the gap between touch screens prevented fluid passing of objects or the emergence of shared territory [24]. By contrast, MAMMI is a system that uses a shared, multi-touch, multi-users surface since shared surfaces make users close together and enable them to interact simultaneously if the interactions were designed appropriately [11].

Designing Direct Collaboration

The requirements from section 0 were mainly designed for general applications. This has often led to specific tools to support one of the three types of activity described by the clover model. However, clever design might also fulfil some requirements without resorting to explicit tool design.

For example, “direct collaboration” is an HCI design principle to invent smooth and meaningful interactions for collaboration. A direct collaboration system is a “collaborative system in which coordination between users is supported by communication and production tools, and not by dedicated coordination tools” [25]. Three rules might guide the design of direct collaboration system:

- **Integrating Communication Media** - for example avoid a rigid sequence of actions that necessitate to enter a session before editing a document, and rather provide an ‘enter session’ button on a shared document, or provide a communication channel on the application
- **Integrating Activities** - To avoid unintended disruptions of a task, groupware systems can manage pending tasks and let the choice of when to collaborate to the users. For example, a shared workspace can be associated with each user and receive objects representing pending requests such as a “phone call” button.
- **Production Space as a Medium** - The production space can be used as a channel for conveying social hints in the same way as traditional communication channels. This is possible by introducing interaction styles that support prosody in the same way as voice intonation or gestures accompany oral communications, thus reinforcing coordination hints. For example, the authors designed “shared transfolders” in which ATCOs can drop a reification of a clearance. This allows for some form of prosody: dropping a clearance in a remote corner of the working area will probably mean a low need for synchronization, while one dropped next to the phone icon in the transfolder will be interpreted as a request for a phone call.

Direct manipulation and touch-based ATC systems also rely on the reification of actions into objects [11]. Since objects lie on the table, their manipulation may enable accountability [25]; furthermore, they can be passed around and allow for task reallocation.

Foster dynamic task allocation by allowing partial accomplishment of actions

A CSCW system should foster dynamic task allocation to increase capacity [11]. Capacity should increase because users will be able to pick up new tasks to be done as soon as they have completed existing tasks (activity preparation and allocation).

Such fostering can be accomplished by carefully subdividing tasks into sub-tasks. For example, an action can be separately prepared, checked and accomplished. Different users can perform each subtask, thus offering seamless workload allocation, as long as the accomplishment status of sub-tasks is visible and as long as the subdivision fulfils operational requirements (especially regulations).

Provide as much feedthrough as possible

Since activities must be accountable, it is important that appropriate feedback provide an opportunity for teammates to observe one another's actions [11]. Carefully designed feedthrough is one way to provide information.

For example, interactions and graphic rendering techniques such as position remanence in radar images or in traditional cursor-based HMIs, Mnemonic Rendering or Phosphor might provide a sense of the history of teammates' actions. Other group activities may benefit from specific visualizations of social protocols, such as social proxies [29], out-of-screen location indicators such as wedges [1], or specific interactions such as the Telepresence interfaces for Mediaspaces [30].

Evaluation of groupware

Some researchers sought to provide specific evaluation criteria to assess CSCW systems. They notably defined a set of heuristics to evaluate CSCW systems, tailored to the groupware genre of shared visual workspaces [31]. As always, evaluation and ideation are the two sides of the system design coin, and heuristics for evaluation can also be used as heuristics for ideation.

- **Heuristic 1:** Provide the means for intentional and appropriate verbal communication (chat, audio, video)
- **Heuristic 2:** Provide the means for intentional and appropriate gestural communication (telepointer, avatars, video)
- **Heuristic 3:** Provide consequential communication of an individual's embodiment (capture and transmit both the explicit and subtle dynamics that occur between collaborating participants)
- **Heuristic 4:** Provide consequential communication of shared artefacts (i.e., artefact feedthrough)
- **Heuristic 5:** Provide Protection (access control, concurrency control, undo, version control, and turn-taking)
- **Heuristic 6:** Management of tightly and loosely coupled collaboration (relaxed-WYSIWIS (What You See Is What I See, overviews)
- **Heuristic 7:** Allow people to coordinate their actions
- **Heuristic 8:** Facilitate finding collaborators and establishing contact (Being available, Knowing who is around and available, Establishing contact, Working together)

Problems in the design and evaluation

Several problems in the design and evaluation of CSCW systems have been identified [32]:

Problem 1. The disparity between who does the work and who gets the benefit.

Many CSCW applications will directly benefit certain users, often managers, while requiring additional work from others. A traditional method of coping with such a problem is to create new jobs or "redesign" existing jobs -- in short, to require people to do the additional work.

Problem 2. The breakdown of « intuitive » decision-making [about designing a CSCW system]

Decision-makers in a position to commit the resources to application development projects rely heavily on intuition. Not surprisingly, the decision-maker is drawn to applications that selectively benefit one subset of the user population: managers. Managers tend to overlook or underestimate the downside, the extra work that might be required of other users to maintain the application.

Problem 3. The underestimated difficulty of evaluating CSCW applications.

Task analysis, design, and evaluation are never easy, but they are considerably more difficult for CSCW applications than for single-user applications.

2.1.5 Conclusion on CSCW

CSCW concerns are important for any collaborative activity, all the more for ATC. Collaboration is already known as key to safety and capacity in ATC. In the context of AEON, the quality of the support to collaboration should also foster better optimization of the use of resources and means to fulfil AEON's sustainability goals. AEON introduces more automation and a new role (towing vehicle manager). The main AEON research questions related to CSCW are: what support for collaboration is needed to consider the three envisioned main roles (pilot, controller, towing vehicle manager)? What support for collaboration is needed to consider both humans and automation aspects? These questions are closely related to distributed situation awareness, which is the topic of the next subsection.

2.2 Distributed Situation Awareness among participants

In cognitive science, Situation Awareness (SA) describes the ability to “knowing what’s happening”, during collaborative work. In this section, we describe existing models of SA and in particular Distributed Situation Awareness (DSA) that seems most adequate for the AEON project due to the distributed nature of ground handling tasks. We will also describe existing design requirements for other aviation systems and evaluation methods that could be useful to assess the effectiveness of tools to support SA.

2.2.1 Situation Awareness Concepts

Endsley distinguishes the term Situation Awareness, as a state of knowledge, from the processes used to achieve that state, as acquiring or maintaining SA [33]. She defines SA 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*”. Figure 5 shows a three-level model of situation awareness. The levels are:

1. Perception (level 1 SA): The basis of SA is formed by the perception of relevant elements in the environment. This involves monitoring or cue detection to enable an awareness of situational elements (objects, events, people, systems...).
2. Comprehension (level 2 SA): A synthesis of the elements through pattern recognition, interpretation and evaluation forms the comprehension of the current situation which is compared to the operators' goals

3. Projection (level 3 SA): The last level is the ability to project future states of the environment to determine how it will affect future states of the operational environments. This level is particularly important for decision making.

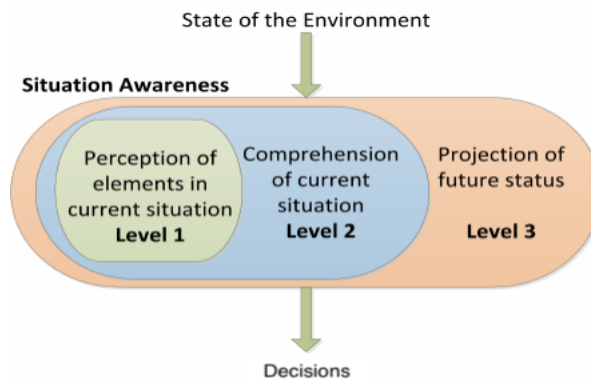


Figure 5: Three levels of situation awareness

For Air Traffic Control, Ruitenbergh defines SA elements for en-route ATCOs [34] as presented in Figure 6. In the scope of the AEON project, we will need to investigate whether these elements are relevant for ground controllers and possibly propose an updated version. We will also need to consider the other stakeholders involved such as airlines crew, pilots, tug drivers and airport teams.

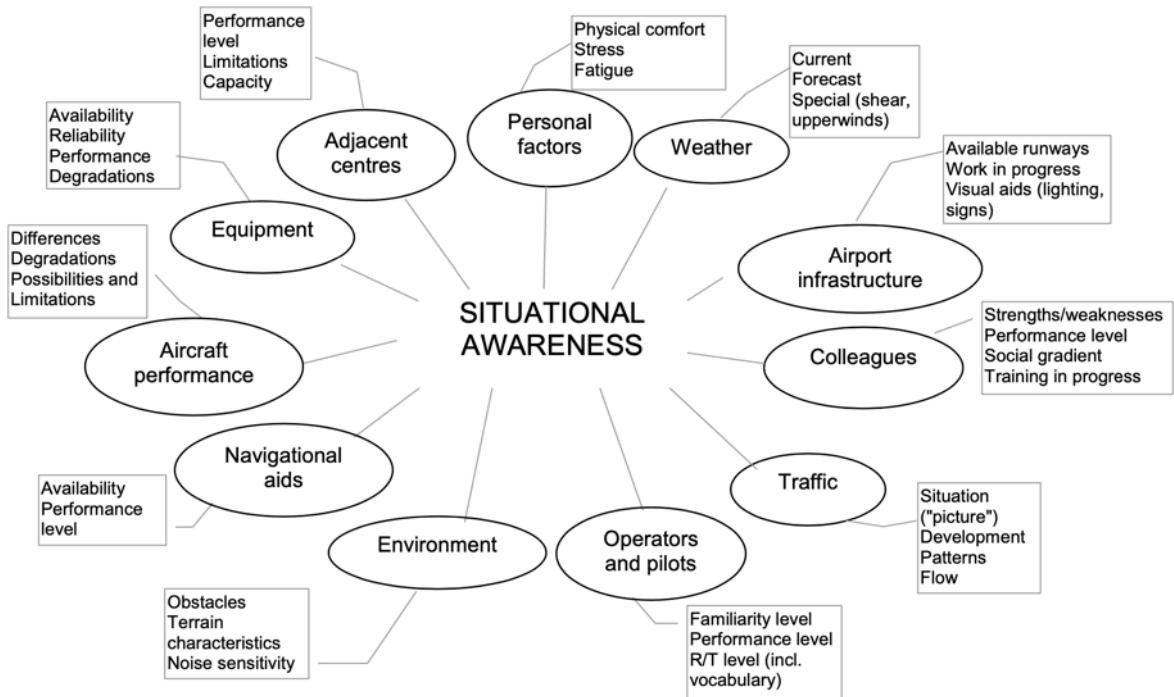


Figure 7: Elements of SA for ATCOs (from [34]).

In many organizational environments, especially complicated ones, a team of individuals takes charge of the operational tasks [35]. The traditional and dominant view emphasizes Team SA on a shared understanding of the situation, that is, the team members should have a common picture. Endsley raised a Team SA model, in which a set of circles overlaps with each other [33] (see Figure 7 left). Each circle represents a team member’s SA elements related to his or her specific role. The overlaps of the circles represent shared SA, and the union of the circles represents Team SA. Team SA is defined as “the degree to which every team member possesses the SA required for his or her responsibilities” [33]. According to this model, the success or failure of a team depends on the success or failure of each of its team members.

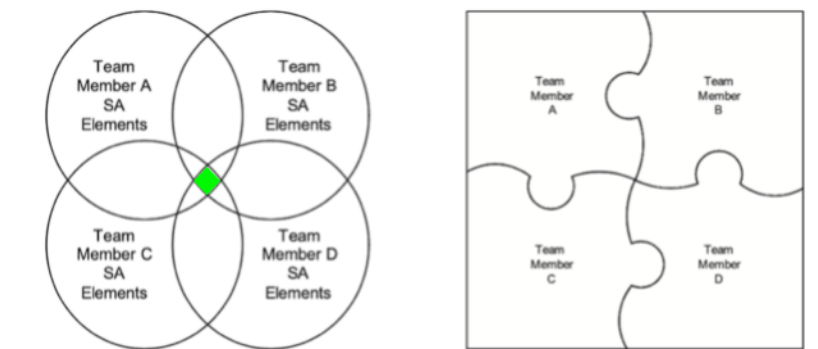


Figure 6: Shared Situation Awareness (left). Compatible Situation Awareness (right) (figure from [36])

Another approach considers Compatible Situation Awareness instead of Shared SA [36]. Compatible SA is based on the notion that no individual working within a collaborative system will hold exactly the same perspective on a situation. Compatible SA therefore suggests that, due to factors such as individual roles, goals, tasks, experience, training and schema, each member of a collaborative system has a unique level of SA that is required to satisfy their particular goals. Each team member does not

need to know everything, rather they possess the SA that they need for their specific task but are also aware of what other team members need to and do know as illustrated in Figure 7 (right). Although different team members may have access to the same information, their resultant awareness of it is not shared, since the team members often have different goals, roles, experience and tasks (and thus different schema) and so view the situation differently based on these factors. We believe that the AEON project falls within the Compatible SA approach as most stakeholders (ATCOs, pilots, truck drivers or marshallers) will likely have some shared elements but different goals, experiences and tasks.

Distributed Situation Awareness

Building upon the Compatible SA that we identify as most relevant for the AEON project, Stanton et al [30] propose that SA is distributed amongst the humans and non-human artefacts in the socio-technical system. In their view of Distributed Situation Awareness (DSA), SA no longer exists solely in the individuals, but is an emergent property of the system. A system analysis cannot be accounted for by summing independent individual analyses. The basis of their theory is described in the six following propositions [36]:

- *SA is held by human and non-human agents. Automated agents as well as human operators have some level of SA in the sense that they are holders of contextually relevant information. For instance, automated tugs do have SA through their sensors.*
- *Different agents have different views on the same scene. This emphasizes the role of past experience, memory, training and perspective. Also, autonomous and automated technology may be able to learn about their environment and evolve over time.*
- *Whether or not one agent's SA overlaps with that of another depends on their respective goals. Different agents could actually be representing different aspects of SA.*
- *Transactions between agents may be verbal and non-verbal behavior, customs and practice. Technologies transact through sounds, signs, symbols via the HMIs.*
- *SA holds loosely coupled systems together. It is argued that without this coupling the systems performance may collapse. Dynamical changes in system coupling may lead to associated changes in DSA.*
- *One agent may compensate for degradation in SA in another agent. This represents an aspect of the emergent behavior associated with complex systems.*

Stanton et al. [36] reports that the application of Distributed SA has led to encouraging results. It promotes higher performance in teams than shared SA. Distributed SA theory offers explanations of the behaviours of complex socio-technical systems in a wide range of domains such as energy distribution or digital mission planning for military operations.

Figure 8 illustrates the models of DSA with a compatible SA model at the center made of Agents and Artefacts. The human and non-human agents and the artefacts exchange situational data via SA transactions.

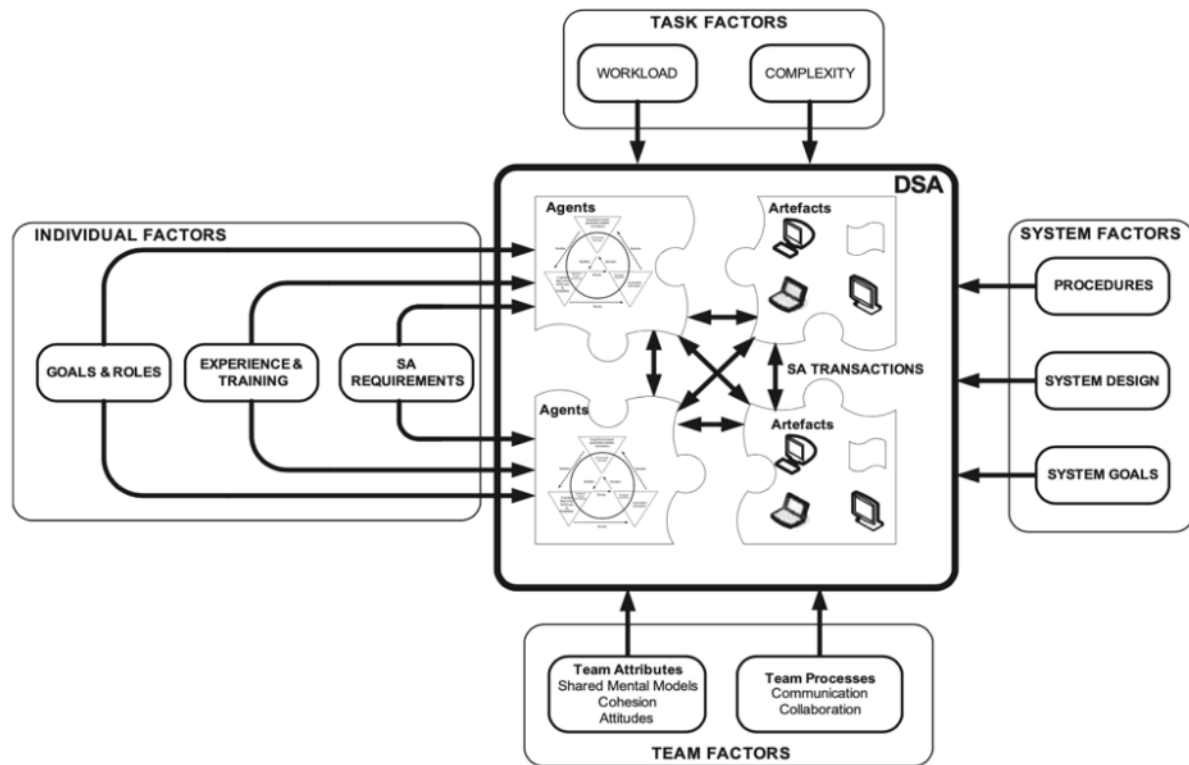


Figure 8: Distributed Situation Awareness Model (from [36])

Within the AEON project, multiple users will collaborate through various systems to achieve their respective goals, i.e., handling outbound and inbound aircrafts. This implies that all stakeholders must build a distributed SA to maximize the effectiveness of the system and maintain both safety and capacity. A design challenge is to offer support for maintaining distributed situation awareness during the operations. In addition to current elements, we will also have to maintain an ecological performance of the whole system. In the next section, we will review the Affecting factors described in this model and relevant design guidelines to support SA in such a distributed system.

Affecting factors

According to Salmon et al. [36], numerous factors affect and influence SA according to task, individual, system and team factors as presented in Figure 8. We describe below some of their components and give details with respect to the specific challenges we may face within the scope of the AEON project. Then, we build upon existing work to describe specific factors related to ATCOs SA.

Task factors concerns the characteristics of the tasks being performed by teams that can either facilitate or inhibit team performance. Factors such as task design, complexity, workload, time pressure, task allocation and familiarity with the task can all potentially affect the DSA acquired during performance of the task in question [37]. For example, the level of workload experienced by team members is a key element in the safety, reliability and efficiency of complex sociotechnical systems such as ATC [38]. Inappropriate levels of workload (either too high or too low) are likely to lead to reduced levels of DSA. Endsley points out that a major factor creating a challenge for operator SA is the increasing complexity of many systems and suggests that complexity can negatively affect SA via factors such as increased system components, the degree of interaction between components and the dynamics or rate of change of the components.

Individual factors concern mostly factors related to how an agent's goals and roles within the system, its experience and training as well as the availability of required data, are adequate to the entire system. AEON will likely introduce minimizing fuel consumption as a new goal for each role and will introduce needs for accessing relevant data to inform decisions.

System factors are related with the design of the system to support development and maintenance of the SA. For instance, if the system fails at presenting the appropriate information to the operator who requires it because it is incomplete or erroneous. The structure of the network of agents involved and the communications channels that are available to the different agents comprising the system are also likely to have an impact on the quality of the system's DSA. Communication links are one of the critical factors in the acquisition and maintenance of DSA and it is important that the appropriate communication links are present within a system and are maintained throughout task performance. In the case of AEON, an important effort will concern the design of communication mechanisms, whether they are explicit (radio, text messages) or implicit (noise of important activity, emotions). Procedures that enforce the communication of critical DSA-related information, such as instructions, work progress and situational updates are particularly important. Stone and Posey [39], for example, suggest that each member's awareness of the current situation could be significantly reduced if communication is not appropriate among members. One approach typically adopted by distributed teams is closed loop communication [38], which involves the initiation of communication by a sender, acknowledgement of receipt of the information by the receiver and then a follow up by the sender to check that the message was interpreted as intended.

Team Factors concerns team attributes and processes. For example, according to Salas et al. [40], there are five main processes: leadership, mutual performance monitoring, back up behavior, adaptability and team orientation, that are necessary for improved team performance. Lack of shared mental models between team members can lead to confusion regarding who held the information to update the situation awareness. Similarly, lack of communication between agents can lead to an out-of-date DSA of the system. Since AEON may introduce a new role in the existing operations workflow, supporting the new Team organization and these five processes need to be considered.

Jeannot [41] offers very interesting results in terms of factors leading to loss of SA and strategies used to recover SA for en-route ATCO as presented in Figure 9 . During AEON, we will keep these factors in mind and explore interaction supporting users in recovering SA if required.

Factors leading to loss of SA	Strategies used to recover SA
<ul style="list-style-type: none"> • Time pressure • Focusing on non-pertinent, or less pertinent information • Focusing on a subset of relevant information but missing the evolution of other information • Becoming reactive rather than proactive • Reduction in room to manoeuvre, • Increased occurrence of non-safe situations • Noise/distraction other people • Mental and/or physical fatigue • Volume of traffic; unexpected and sudden variation of traffic load • Number of phone calls • Lack of strip information at the right time • Lack of adequate feedback • Too much happening and having to process too much information • Traffic building up • Unusual or unexpected events (e.g. aircraft calling in too early) 	<ul style="list-style-type: none"> • Increase in delay between pilot calls and controller answer • Check consistency between strips and radar • Force themselves to speak slowly and precisely, return to strict phraseology • Request help: ask for sector splitting, if possible, to decrease load • Analyse closely all the strip • Force themselves not to spend too much time on a single problem • Change principle of strip classification, re-organise strips according to new criteria (e.g. by entry or exit point) • Physically manipulate the strips (i.e. re-positioning the strips aids concentration) • Prioritise work and “forget” less important tasks • Disregard the strips of ‘hello-goodbye’ a/c (in the case of too many strips) • Always prioritise new strips

Figure 9: SA factors for ATCOs from [41]

Another contribution from the same work concerns indicators of good and reduced SA as presented in Figure 10. Some of the indicators of reduced or impaired SA such as delay between pilot calls and controller answer or the need to check the same information several times can be monitored and represented by the AEON solution to alert a specific operator as well as the others on possible risk of reduced SA. Using indicators as the use of exclusive language such as 'but' and 'except', and the use of second person pronouns has been used to automatically assess viable team interactions for classrooms [42] and could be adapted to the AEON project.

Indicators of good SA	Indicators of reduced/impaired SA
<ul style="list-style-type: none"> • Anticipating events • Being able to predict next a/c call • Managing resources (technical system , internal, team ...) • Managing time • Feeling of being in control; able to implement elegant solutions • Taking the right decision at the best moment, managing traffic in a safe and expeditious way • Detecting mismatches 	<ul style="list-style-type: none"> • Increase in delay between pilot calls and controller answer • Inconsistency in communication with pilots, colleagues, adjacent sectors or centre • Sudden and unexpected variation of workload • Confusion • Need to check same information several times

Figure 10: Indicators of good or reduced SA from [41]

Measuring SA

Several techniques exist to measure situation awareness of individual members as well as Distributed Situation Awareness as reviewed in [36]. The most popular technique is the Situation Awareness Global Assessment Technique (SAGAT) that consists in interrupting simulation at random times and asks the crew to state their assessment of the current situation based on Endsley's three levels model. There is a specific version of the SAGAT for Air Traffic Control that could be relevant to use within the AEON project. Other tools such as Situation Awareness Rating Technique (SART) or Situation Awareness Subjective Workload Dominance (SA-SWORD) are also used. These techniques are often used for assessing pilot's situation awareness, but some work also studied more collaborative contexts and ATC contexts.

For instance, the Situation Present Assessment Method [44] is a real-time probe technique that was developed to assess air traffic controllers' awareness. The technique was improved and used by Eurocontrol with SASHA [41] for the assessment of air traffic controller SA in automated systems. SASHA comprises two techniques, SASHA_L (real-time probe technique) and SASHA_Q (post-trial questionnaire). SASHA_L is based on the Situation Present Assessment Method [44] and involves probing the participant on-line using real-time SA related queries. The response content and response time is recorded. Once the trial is completed, the participant completes the SASHA_Q questionnaire, which consists of 10 questions designed to elicit subjective participant ratings of SA. This technique might be used to assess the SA support of the AEON solution. However, question will need to be adapted to ground controllers as most of the existing work focused on en-route Air Traffic Control and did not consider other operators in the loop. In our case, we might need to assess SA of each such as airlines operations, or airport management.

Design guidelines

In this section, we review some related work that offers guidelines for designing collaborative systems able to support Situation Awareness. As pointed out, traditional Human Factor design guidelines are inadequate for achieving the SA required in complex systems [33]. Stanton et al. [36] built upon several project to offer the following guidelines. We describe some of them that seemed the most important within the scope of AEON.

Clearly define and specify SA requirements: The collaborative system design process should begin with a clear definition and specification of the DSA requirements of the overall system and of the different operators working in the system in question.

Design to support compatible SA requirements: Rather than present everything to everyone, or use common operational picture displays, collaborative systems should be designed so that users are not presented with information, tools and functionality that they do not explicitly require. Systems should therefore be designed to support the roles, goals and SA requirements of each of the different users involved in the process in question. This might involve the provision of different displays, tools and functions for the different roles and tasks involved, or might involve the use of customisable interfaces and displays.

Design to support SA transactions: SA transactions are means by which DSA is developed and maintained during collaborative tasks by the agents. Systems and interfaces that present information to team members should therefore be designed so that they support SA transactions where possible. This might involve presenting incoming SA transaction information in conjunction with other relevant information (i.e. information that the incoming information is related to and is to be combined with) and also providing users with clear and efficient communications links with other team members. Similarly, procedures can be used to support SA transactions; this might involve incorporating certain pieces of information into procedural communications between team members in order to support SA transactions.

Use multiple interlinked systems for multiple roles and goals: role specific systems might be more appropriate to support DSA development and maintenance. When a team is divided into distinct roles, team members have very different goals and informational requirements; it may therefore be pertinent to offer separate (but linked) support systems.

Ensure that the information presented to users is accurate at all times: The information presented by any collaborative system should therefore be highly accurate and system designers need to ensure that the information presented by all aspects of the system is accurate at all times.

Ensure information is presented to users in a timely fashion and that the timeliness of key information is represented: SA-related information should therefore be presented to users in a timely manner, without any delay, at all times. Further, the timeliness of information should be represented on interfaces and displays, allowing users to determine the latency of information.

Use procedures to facilitate DSA: procedures are an effective means of facilitating DSA acquisition and maintenance through SA transactions. It is therefore recommended that procedures should be used to support SA transactions via encouraging the continual communication of DSA-related information around collaborative systems and also by structuring communications so that related information is communicated together.

Conclusion on Distributed Situation Awareness

DSA gives interesting direction to design groupware systems able to support operators understanding their current environment and making informed decisions. Building from work presented in this section, we will involve participants from the early design phases to gather SA requirements for ground ATCOs and supervisors of the towing vehicles. We will also explore interactions to achieve efficient

and effective SA transactions. We will try to assess the DSA status of the operators during the design and experimental phases of the project.

2.3 Human Automation Teaming

Human Automation Teaming (HAT) can be defined as a group of human and autonomous agents, performing activities and achieving outcomes together towards a common goal. In particular in HAT, the autonomous agents work alongside humans performing essential tasks and teamwork functions that a human would [45]. This new teamwork configuration is partially shifting the role of human agents to supervisors, adding new tasks to their workload. These tasks include assessing situations for taking decisions, correcting agents' errors or even managing agents' failures. However as autonomous systems become more and more reliable, operators tend to detach from automation making the ability to takeover autonomous control challenging. Therefore, it is necessary to keep human operators attentive to autonomous agents' activities and actions. This is highly relevant to AEON as autonomous solutions for taxiing have already started to be deployed in airports around the globe. In this section, we discuss the challenges to support collaboration between human and autonomous agents, and to design reliable and efficient HAT interactions which will allow AEON's stakeholders to perform their activity in an optimal way.

Human-automation is not new. Its genesis can be dated to the early 70s with research on conversational agents [46]. In his research, Carbonnel introduced SCHOLAR a novel type of computer assisted instruction system capable of reviewing the knowledge of a student in a given context [46]. In this system, students are prompted by the agent which can communicate students' request statuses, detect misspelling, answer students' questions with acceptable English or generate test questions and evaluate students' test answers. SCHOLAR was the first system to propose and maintain a mixed-initiative man computer dialog. Mixed-initiative interaction is defined as a flexible interaction strategy in which each agent (human or computer) contributes to what it is best suited at the most appropriate time [47]. Horvitz describes mixed initiative support as an efficient, natural interleaving of contributions by users and automated services aimed at converging on solutions to problems [47]. In his vision, not only these systems take advantage of combining the power of direct manipulation and potentially valuable automated reasoning, but they also facilitate collaboration between users and intelligent services to achieve their goal.

The last decades have seen the rapid development of sensors and computational power. Autonomous agents can now perform sophisticated functions with no or little intervention of humans [48]. Depending on the function performance, reliability and importance, different levels of autonomy can be applied. However, higher level of autonomy contributes to better team performance by increasing communication efficiency, improving coordination and reducing workload [49].

In HATs, autonomous agents perform complex tasks which require to engage with other teammates to achieve team objectives. Previous research has established that task difficulty has an effect on team performance, especially the tasks switching frequency and the workload increasing with the difficulty [50], [51]. As the difficulty increases, more effort and time are needed to deal with the tasks, causing the communication and engagement between agents to reduce [50].

Teams composed of humans alone tend to outperform teams with autonomous agents [52]. While autonomous agents are able to manage workload better, human operators can adapt to new situations

better [53], thanks to the way human agents communicate to each other. Mc Neese et al. have shown that humans provide more status updates and request less information between them than when collaborating with autonomous agents [45]. Therefore, in order to support the supervising role of human agents in HATs, communication is key. In fact, the more reliable and robust the automation is, the less likely human operators will be aware of critical information and will take manual control when needed [54]. This is very relevant to AEON as any taxi failure will critically impact the airport runways schedules.

In a recent effort to summarise the research on level of automation and situation awareness, Endsley has proposed a Human-Autonomy System Oversight (HASO) model that takes into consideration the impact of the level of automation on human performance and cognition for automation task stages. These stages include monitoring and information presentation, generation of options, decision making and implementation of actions [54]. The specificity of HASO is the implementation of strategies to reduce workload and improve human engagement towards automation through level of automation, adaptive automation and grain of automation control. As a result, a list of guidelines to support operator situation awareness and autonomy oversight has been proposed (Table 5).

These guidelines, which complete previous guidelines on situation awareness (see section 0. Design guidelines), ensure that (1) the designed interfaces will provide a high level of transparency promoting understandability and predictability of the system, (2) will communicate agent's reliability or robustness to the operator at all times, and (3) will allow the operators to develop a mental model of the autonomy and perform the appropriate actions to keep the system in line with their goals.

<i>Guidelines</i>	<i>Summary</i>
Automate only if necessary and avoid operators to detach from automation as much as possible	Automation should be avoided unless its assistance is really needed as it can lead to significant problems such as lack of understanding, system complexity, decision biasing and out-of-the-loop performance issues.
Use automated assistance for carrying out routine tasks rather than higher-level cognitive functions	Automation that carries out the action in a routine task is highly beneficial for reducing manual workload, however decisional automation should be avoided as it can create out-of-the-loop problems.
Provide system awareness support rather than decisions	Systems that provide situation awareness through well designed information presentation to operators, integration and projection yield better performances and robustness.
Keep the operator in control and in the loop	Increasing operator involvement and control will improve engagement in task performance. Ensure that the operator maintains control over the automation, and design strategies that include human decision in the task flow.
Avoid the proliferation of automated modes	Keeping autonomy modes low will help operators to develop a good mental model of how the system works and will facilitate the autonomy mode tracking and learning.

Make modes and system states salient	The current mode of automation should be made salient to the operator (including mode transitions back to manual operation). The current state of the automation should be salient so that any violation of operator expectations will be readily apparent.
Enforce automation consistency	Consistency in the terminology, information placement, and functionality of the system between modes should be enforced to minimize errors in working with system autonomy.
Avoid advanced queuing of tasks	Approaches that maintain operator involvement in the decisions associated with execution of tasks will avoid potential failures with no immediate support that could interrupt the task flow.
Avoid the use of information cuing	Automatic highlighting of information should be avoided in favour of approaches that allow people to use their own senses more effectively. Providing strategies to declutter unwanted information or improving picture clarity are preferable.
Use methods of decision to support that create human/system symbiosis	Encourage people to consider multiple possibilities and perform contingency planning that can help project future states and provide systems that support humans to consider alternate interpretation of the data.
Provide automation transparency	Make clearly apparent what the system is currently doing, why it is doing it and what it will do next to improve transparency and observability of the system.
Ensure logical consistency across features and modes	Differences in operational logic, information presentation and sequences of input not necessary for the operation should be reduced to limit system complexity.
Minimize logic branches	Reduce linkages and conditional operations in the autonomy and modes as much as possible to minimize system complexity.
Map system functions to goals and mental models of users	Use a clear mapping between user goal and system functions to limit the degree to which operators need to understand the underlying software or hardware linkages to operate or oversee the autonomy.
Minimize task complexity	Limit the number of actions needed to perform desired tasks to reduce the sequence errors and cognitive load in interacting with the autonomy.

Table 5: Guidelines for supporting human understanding of autonomous systems and the reduction of complexity in autonomous system (from [54])

Transparency of autonomous agents is critical to maintain situation awareness and to allow reliable supervision by human operators. Building upon [33], Chen et al. have proposed the *Situation Awareness-based Agent Transparency* (SAT) model to identify the information autonomous agents need to provide to ensure a clear communication with human operators [55]. Matching Endsley's situation awareness levels [33] (Section 2), they have proposed 3 levels of information that should be accessible to operators for effective supervision control. While the first level provides basic information about the current agent's state and goals, intentions and plans, the second level informs about the agent's reasoning process and all the considerations when planning its actions. Finally, the third level communicates future states, predicted results and consequences, and uncertainties about projections. A summary of the model is provided in Figure 11. In AEON, stakeholders may have to collaborate with autonomous taxiing agents. Therefore, we will identify the relevant information at all SAT levels required by each stakeholder for a reliable and efficient collaboration.

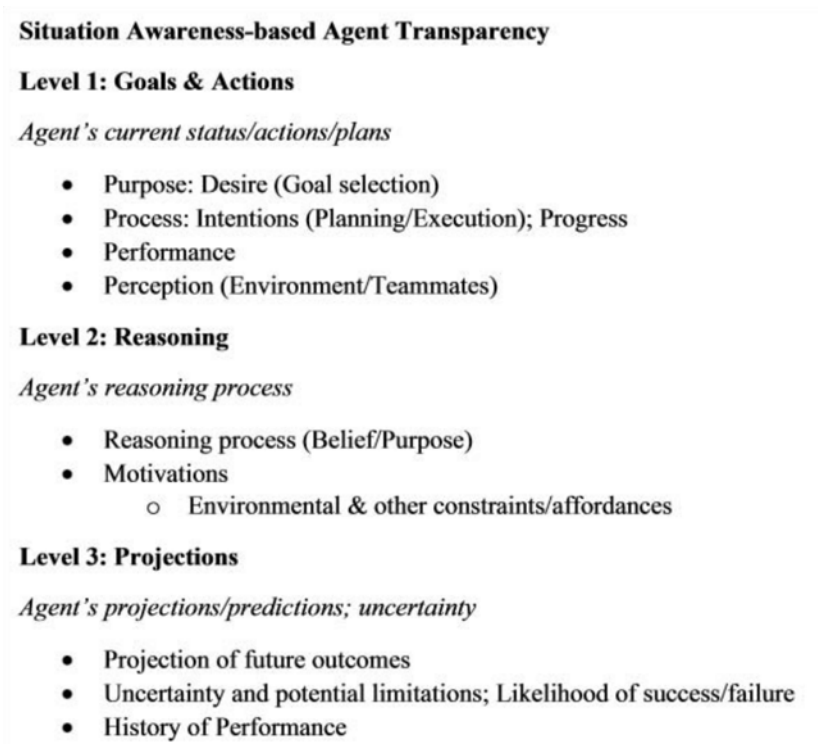


Figure 11: The situation awareness-based agent transparency model (from [55])

Although avoiding errors in automation is crucial, not all errors can be prevented. Automation errors such as missed and false alarms or misdiagnoses can be defined as events that make “the automation behaves in a manner that is inconsistent with the true state of the world” [56]. In the context of AEON, these errors, which can yield severe consequences, could create disruptions in airport ground operations resulting in queuing delays on airport runways. Furthermore, external and unexpected challenges may degrade the condition in which human operators need to perform their tasks, and affect the performance of the automation [57]. To mitigate the detrimental consequences of imperfect automation, it will be essential to provide operators with control that allows flexibility and resilience of the automation [57], such as in adaptable systems where human operators are involved in the decision of what and when to automate [58].

High level of automation, low human engagement and teaming will require elaborated human-machine interfaces that support shared awareness between human and the autonomy to coordinate

actions towards shared goals. The guidelines introduced in this section will serve as a design frame to elaborate novel and compelling interaction techniques that will ensure safe and efficient oversee of autonomous systems within AEON. However, the research on designing effective shared situation awareness in HAT is still young. The research and interaction design in AEON will contribute to establish novel shared situation awareness representations to enable effective collaboration between AEON stakeholders, and between humans and autonomy.

2.4 Summary and research directions

This section offered a comprehensive set of definitions, examples, guidelines and techniques to support the design of collaborative HMIs with the AEON project. We started by reviewing GroupWare and CSCW concepts, tools and examples in Air Traffic Control. We then investigated Situation Awareness and Distributed Situation Awareness to cover how information needs to be exchanged and maintained between all participants so that the teamwork is effective. We concluded with a review of Human Automation Teaming as the collaboration in AEON will involve not only humans but also intelligent agents such as planning algorithms or autonomous towing vehicles. Such context introduces subtle changes and provide additional guidelines to support the design of an effective solution. Below we summarize several Research Directions that we need to explore during the AEON project.

Research Directions:

In order to provide efficient and reliable interaction techniques that will support collaboration between humans and autonomous agents optimally, the research related to Human-Machine collaboration aspects in AEON will be twofold.

The first research axis will aim at understanding the role and the contribution of each stakeholder in taxiing operations at airports. Alongside with identifying and documenting all the tasks performed by the operators, a specific focus will be given to investigate all the explicit and tacit communication processes that come into play to support situation awareness and limit workload for each operator. We will also examine the effect of plane towing automation on operators' activities in order to identify the mechanisms linking the input and the output that will change for each operator, and the new issues that will need to be addressed in the future of airports ground operations.

The second research axis will aim at providing usable interaction techniques and interactive systems that will support distributed situation awareness in human-automation teaming and facilitate rapid decision and tactical collaboration. This state-of-the art will provide us with a framework to emphasise situation awareness support between all the human and autonomous workers in our design approach. In addition, we will tailor our approach to the specific context of AEON through user-centred design in order to create and evaluate innovative and efficient representations of information and interactive controls in stakeholders' digital monitoring tools that will take advantage of human senses and humans' ability to understand complex situation rapidly through multimodal and feedback. This will allow operators to maximise taxiing operations flow and manage fleet of taxis and aircrafts.

References

- [1] C. A. Ellis, S. J. Gibbs, and G. Rein, "Groupware: some issues and experiences," *Commun. ACM*, vol. 34, no. 1, pp. 39–58, 1991, doi: 10.1145/99977.99987.
- [2] J. Rama and J. Bishop, "A Survey and Comparison of CSCW Groupware Applications," in *Proceedings of the 2006 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists on IT Research in Developing Countries, ZAF*, 2006, pp. 198–205, doi: 10.1145/1216262.1216284.
- [3] 14:00-17:00, "ISO 9241-210:2019," *ISO*. <https://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/07/75/77520.html> (accessed Apr. 13, 2021).
- [4] K. J. Lynch, J. M. Snyder, D. R. Vogel, and W. K. McHenry, "The arizona analyst information system: Supporting collaborative research on international technological trends," in *Proceedings of the IFIP WG 8.4 confernece on multi-user interfaces and applications*, USA, 1990, pp. 159–174.
- [5] D. Englebart, "The augmented knowledge workshop," in *Proceedings of the ACM Conference on The history of personal workstations*, Palo Alto, California, USA, 1986, pp. 73–83, doi: 10.1145/12178.12184.
- [6] S. Greenberg, "Sharing views and interactions with single-user applications," in *Proceedings of the conference on Office information systems -*, Cambridge, Massachusetts, United States, 1990, pp. 227–237, doi: 10.1145/91474.91546.
- [7] J. Tam and S. Greenberg, "A framework for asynchronous change awareness in collaborative documents and workspaces," *Int. J. Hum.-Comput. Stud.*, vol. 64, no. 7, pp. 583–598, Jul. 2006, doi: 10.1016/j.ijhcs.2006.02.004.
- [8] W. C. Hill, J. D. Hollan, D. Wroblewski, and T. McCandless, "Edit wear and read wear," in *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '92*, Monterey, California, United States, 1992, pp. 3–9, doi: 10.1145/142750.142751.
- [9] C. Mertz, S. Chatty, and J.-L. Vinot, "Pushing the limits of ATC user interface design beyond S&M interaction: the DigiStrips experience," 2000.
- [10] C. Gutwin, "Traces: Visualizing the Immediate Past to Support Group Interaction," *Proc. Graph. Interface 2002*, vol. Calgary, p. 8 pages, 258.41 KB, 2002, doi: 10.20380/GI2002.06.
- [11] S. Conversy, H. Gaspard-Boulinç, S. Chatty, S. Valès, C. Dupré, and C. Ollagnon, "Supporting air traffic control collaboration with a TableTop system," in *Proceedings of the ACM 2011 conference on Computer supported cooperative work*, 2011, pp. 425–434.
- [12] S. Valès et al., "MAMMI Phase 3-Exploring collaborative workspaces for air traffic controllers in the scope of SESAR," in *INO 2008, 7th Eurocontrol Innovative Research Workshop & Exhibition*, 2008, p. pp-15.
- [13] A. Dix, "Computer-supported cooperative work - a framework," in *Design Issues in CSCW*, Springer Verlag, pp. 23–37.
- [14] D. Salber, "De l'interaction homme-machine individuelle aux systèmes multi-utilisateurs. L'exemple de la communication homme-homme médiatisée," Université Joseph-Fourier - Grenoble I, 1995. [Online]. Available: <https://tel.archives-ouvertes.fr/tel-00005060>.
- [15] G. Calvary, J. Coutaz, and L. Nigay, "From single-user architectural design to PAC*," in *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems*, 1997, pp. 242–249, doi: 10.1145/258549.258717.
- [16] S. Ratcliffe, "Strategic or Tactical Air Traffic Control?," *J. Navig.*, vol. 31, no. 3, pp. 337–347, 1978, doi: 10.1017/S0373463300041886.

- [17] J. Stewart, B. B. Bederson, and A. Druin, "Single display groupware: a model for co-present collaboration," in *Proceedings of the SIGCHI conference on Human factors in computing systems the CHI is the limit - CHI '99*, Pittsburgh, Pennsylvania, United States, 1999, pp. 286–293, doi: 10.1145/302979.303064.
- [18] K. Schmidt and T. Rodden, "Chapter 11 Putting it all together: Requirements for a CSCW platform," in *Human Factors in Information Technology*, vol. 12, Elsevier, 1996, pp. 157–175.
- [19] N. Roussel and S. Gueddana, "Beyond 'beyond being there': towards multiscale communication systems," in *Proceedings of the 15th international conference on Multimedia - MULTIMEDIA '07*, Augsburg, Germany, 2007, p. 238, doi: 10.1145/1291233.1291283.
- [20] C. Mertz and R. Benhacène, "Users Bandwidth in Air Traffic Management: an Analysis from the HMI Point of View," 2002.
- [21] W. E. Mackay, "Is paper safer? The role of paper flight strips in air traffic control," *ACM Trans. Comput.-Hum. Interact.*, vol. 6, no. 4, pp. 311–340, Dec. 1999, doi: 10.1145/331490.331491.
- [22] H. Ishii, M. Kobayashi, and J. Grudin, "Integration of Interpersonal Space and Shared Workspace: ClearBoard Design and Experiments," *ACM Trans. Inf. Syst. TOIS*, vol. 11, no. 4, pp. 349–375, 1993, doi: 10.1145/159764.159762.
- [23] M.-C. Bressolle, B. Pavard, and M. Leroux, "The role of multimodal communication in cooperation: The cases of air traffic control," in *International Conference on Cooperative Multimodal Communication*, 1995, pp. 326–343.
- [24] C. Mertz and R. Benhacène, "Users Bandwidth problems in ATC: How DigiStrips, DigiListes and HMI techniques address it," 2001.
- [25] S. Sire, S. Chatty, H. Gaspard-Boulin, and F.-R. Colin, "How Can Groupware Preserve our Coordination Skills? Designing for Direct Collaboration," in *Human-Computer Interaction INTERACT '99: IFIP TC13 International Conference on Human-Computer Interaction, Edinburgh, UK, 30th August-3rd September 1999*, 1999, pp. 304–312.
- [26] C. Hurter, S. Conversy, J.-L. Vinot, and Y. Jestin, "Représentations écologiques de données temporelles: exemples et apports," in *Proceedings of the 21st International Conference on Association Francophone d'Interaction Homme-Machine*, New York, NY, USA, Oct. 2009, pp. 33–42, doi: 10.1145/1629826.1629832.
- [27] A. Bezerianos, P. Dragicevic, and R. Balakrishnan, "Mnemonic rendering: an image-based approach for exposing hidden changes in dynamic displays," in *Proceedings of the 19th annual ACM symposium on User interface software and technology*, New York, NY, USA, Oct. 2006, pp. 159–168, doi: 10.1145/1166253.1166279.
- [28] P. Baudisch *et al.*, "Phosphor: explaining transitions in the user interface using afterglow effects," in *Proceedings of the 19th annual ACM symposium on User interface software and technology*, New York, NY, USA, Oct. 2006, pp. 169–178, doi: 10.1145/1166253.1166280.
- [29] T. Erickson, D. N. Smith, W. A. Kellogg, M. Laff, J. T. Richards, and E. Bradner, "Socially translucent systems: social proxies, persistent conversation, and the design of 'babble'," in *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, New York, NY, USA, May 1999, pp. 72–79, doi: 10.1145/302979.302997.
- [30] S. Gustafson, P. Baudisch, C. Gutwin, and P. Irani, "Wedge: clutter-free visualization of off-screen locations," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, New York, NY, USA, Apr. 2008, pp. 787–796, doi: 10.1145/1357054.1357179.
- [31] J. R. Cooperstock, K. Tanikoshi, G. Beirne, T. Narine, and W. A. S. Buxton, "Evolution of a reactive environment," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, USA, May 1995, pp. 170–177, doi: 10.1145/223904.223926.
- [32] K. Baker, S. Greenberg, and C. Gutwin, "Heuristic Evaluation of Groupware Based on the Mechanics of Collaboration," in *Proceedings of the 8th IFIP International Conference on Engineering*

- for *Human-Computer Interaction*, Berlin, Heidelberg, May 2001, pp. 123–140, Accessed: Mar. 23, 2021. [Online].
- [33] J. Grudin, “Why CSCW applications fail: problems in the design and evaluation of organizational interfaces,” in *Proceedings of the 1988 ACM conference on Computer-supported cooperative work - CSCW '88*, Portland, Oregon, United States, 1988, pp. 85–93, doi: 10.1145/62266.62273.
- [34] M. R. Endsley, “Toward a theory of situation awareness in dynamic systems,” *Hum. Factors*, vol. 37, no. 1, pp. 32–64, 1995.
- [35] B. Ruitenber, “SITUATIONAL AWARENESS IN ATC -- A MODEL,” *The Controller*, Mar. 1997, Accessed: Mar. 26, 2021. [Online]. Available: <https://trid.trb.org/view/565595>.
- [36] M. She and Z. Li, “Team situation awareness: A review of definitions and conceptual models,” in *International Conference on Engineering Psychology and Cognitive Ergonomics*, 2017, pp. 406–415.
- [37] N. A. Stanton, “Distributed situation awareness,” *Theor. Issues Ergon. Sci.*, vol. 17, no. 1, pp. 1–7, Jan. 2016, doi: 10.1080/1463922X.2015.1106615.
- [38] A. Gregoriades and A. Sutcliffe, “Workload prediction for improved design and reliability of complex systems,” *Reliab. Eng. Syst. Saf.*, vol. 93, no. 4, pp. 530–549, 2008.
- [39] E. Salas, C. S. Burke, and S. N. Samman, “Understanding command and control teams operating in complex environments,” *Inf. Knowl. Syst. Manag.*, vol. 2, no. 4, pp. 311–323, 2001.
- [40] N. J. Stone and M. Posey, “Understanding coordination in computer-mediated versus face-to-face groups,” *Comput. Hum. Behav.*, vol. 24, no. 3, pp. 827–851, May 2008, doi: 10.1016/j.chb.2007.02.014.
- [41] “Is there a ‘Big Five’ in Teamwork? - Eduardo Salas, Dana E. Sims, C. Shawn Burke, 2005.” <https://journals.sagepub.com/doi/abs/10.1177/1046496405277134> (accessed Mar. 25, 2021).
- [42] E. Jeannot, C. Kelly, and D. Thompson, *The Development of Situation Awareness Measures in ATM Systems*. 2003.
- [43] H. Cao *et al.*, “My Team Will Go On: Differentiating High and Low Viability Teams through Team Interaction,” *Proc. ACM Hum.-Comput. Interact.*, vol. 4, no. CSCW3, pp. 1–27, Jan. 2021, doi: 10.1145/3432929.
- [44] M. R. Endsley, “Situation awareness global assessment technique (SAGAT),” in *Proceedings of the IEEE 1988 national aerospace and electronics conference*, 1988, pp. 789–795.
- [45] F. T. Durso, C. A. Hackworth, T. R. Truitt, J. Crutchfield, D. Nikolic, and C. A. Manning, “Situation awareness as a predictor of performance for en route air traffic controllers,” *Air Traffic Control Q.*, vol. 6, no. 1, pp. 1–20, 1998.
- [46] N. J. McNeese, M. Demir, N. J. Cooke, and C. Myers, “Teaming With a Synthetic Teammate: Insights into Human-Autonomy Teaming,” *Hum. Factors*, vol. 60, no. 2, pp. 262–273, 2018, doi: 10.1177/0018720817743223.
- [47] J. R. Carbonell, “AI in CAI: An Artificial-Intelligence Approach to Computer-Assisted Instruction,” *IEEE Trans. Man-Mach. Syst.*, vol. 11, no. 4, pp. 190–202, Dec. 1970, doi: 10.1109/TMMS.1970.299942.
- [48] J. E. Allen, C. I. Guinn, and E. Horvitz, “Mixed-initiative interaction,” *IEEE Intell. Syst. Their Appl.*, vol. 14, no. 5, pp. 14–23, Sep. 1999, doi: 10.1109/5254.796083.
- [49] R. Parasuraman, T. B. Sheridan, and C. D. Wickens, “A model for types and levels of human interaction with automation,” *IEEE Trans. Syst. Man Cybern. - Part Syst. Hum.*, vol. 30, no. 3, pp. 286–297, May 2000, doi: 10.1109/3468.844354.
- [50] M. Johnson, J. M. Bradshaw, P. Feltovich, C. Jonker, B. van Riemsdijk, and M. Sierhuis, “Autonomy and interdependence in human-agent-robot teams,” *IEEE Intell. Syst.*, vol. 27, no. 2, pp. 43–51, Mar. 2012, doi: 10.1109/MIS.2012.1.
- [51] M. C. Wright and D. B. Kaber, “Effects of Automation of Information-Processing Functions on Teamwork,” *Hum. Factors*, vol. 47, no. 1, pp. 50–66, 2005, doi: 10.1518/0018720053653776.

- [52] X. Fan, M. McNeese, and J. Yen, “NDM-Based Cognitive Agents for Supporting Decision-Making Teams,” *Human–Computer Interact.*, vol. 25, no. 3, pp. 195–234, 2010, doi: 10.1080/07370020903586720.
- [53] T. O’Neill, N. McNeese, A. Barron, and B. Schelble, “Human–Autonomy Teaming: A Review and Analysis of the Empirical Literature,” *Hum. Factors*, p. 0018720820960865, Oct. 2020, doi: 10.1177/0018720820960865.
- [54] D. A. Grimm, M. Demir, J. C. Gorman, and N. J. Cooke, “Team Situation Awareness in Human-Autonomy Teaming: A Systems Level Approach,” *Proc. Hum. Factors Ergon. Soc. Annu. Meet.*, vol. 62, no. 1, pp. 149–149, Sep. 2018, doi: 10.1177/1541931218621034.
- [55] M. R. Endsley and S. A. Technologies, “From Here to Autonomy : Lessons Learned From Human – Automation Research,” doi: 10.1177/0018720816681350.
- [56] J. Y. C. Chen, S. G. Lakhmani, K. Stowers, A. R. Selkowitz, J. L. Wright, and M. Barnes, “Situation awareness-based agent transparency and human-autonomy teaming effectiveness,” *Theor. Issues Ergon. Sci.*, vol. 19, no. 3, pp. 259–282, May 2018, doi: 10.1080/1463922X.2017.1315750.
- [57] S. E. McBride, W. A. Rogers, and A. D. Fisk, “Understanding human management of automation errors,” *Theor. Issues Ergon. Sci.*, vol. 15, no. 6, pp. 545–577, Nov. 2014, doi: 10.1080/1463922X.2013.817625.
- [58] D. Grimm, M. Demir, J. C. Gorman, and N. J. Cooke, “Systems Level Evaluation of Resilience in Human-Autonomy Teaming under Degraded Conditions,” in *2018 Resilience Week (RWS)*, Denver, CO, Aug. 2018, pp. 124–130, doi: 10.1109/RWEEK.2018.8473561.
- [59] C. A. Miller and R. Parasuraman, “Designing for Flexible Interaction Between Humans and Automation: Delegation Interfaces for Supervisory Control,” *Hum. Factors J. Hum. Factors Ergon. Soc.*, vol. 49, no. 1, pp. 57–75, Feb. 2007, doi: 10.1518/001872007779598037.
- [60] M. D. Weerd, A. Mors, and C. Witteveen, “Multi-agent Planning An introduction to planning and coordination,” pp. 1–32, 2005.
- [61] E. H. Durfee, “Distributed problem solving and planning,” in *ECCAI Advanced Course on Artificial Intelligence*, 2001, pp. 118–149.
- [62] T. W. Malone, K. Crowston, and others, “Toward an interdisciplinary theory of coordination,” 1991.
- [63] K. Fines, “Decentralized Control for Resilient Airport Surface Movement Operations,” 2018.
- [64] A. Barrett and D. S. Weld, “Partial-order planning: evaluating possible efficiency gains,” *Artif. Intell.*, vol. 67, no. 1, pp. 71–112, 1994.
- [65] K. S. Decker and V. R. Lesser, “Generalizing the partial global planning algorithm,” *Int. J. Intell. Coop. Inf. Syst.*, vol. 1, no. 02, pp. 319–346, 1992.
- [66] M. M. Veloso, J. Biswas, B. Coltin, and S. Rosenthal, “CoBots: Robust Symbiotic Autonomous Mobile Service Robots,” in *IJCAI*, 2015, p. 4423.
- [67] R. Morris *et al.*, “Planning, scheduling and monitoring for airport surface operations,” 2016.
- [68] E. Erdem, D. G. Kisa, U. Oztok, and P. Schüller, “A general formal framework for pathfinding problems with multiple agents,” *Proc. 27th AAAI Conf. Artif. Intell. AAAI 2013*, pp. 290–296, 2013.
- [69] R. Luna and K. E. Bekris, “Push and swap: Fast cooperative path-finding with completeness guarantees,” *IJCAI Int. Jt. Conf. Artif. Intell.*, pp. 294–300, 2011, doi: 10.5591/978-1-57735-516-8/IJCAI11-059.
- [70] D. Silver, “Cooperative Pathfinding,” *AIIDE*, vol. 1, pp. 117–122, 2005.
- [71] T. Standley, “Finding optimal solutions to cooperative pathfinding problems,” *Proc. Natl. Conf. Artif. Intell.*, vol. 1, pp. 173–178, 2010.
- [72] M. Barer, G. Sharon, R. Stern, and A. Felner, “Suboptimal variants of the conflict-based search algorithm for the multi-agent pathfinding problem,” *Proc. 7th Annu. Symp. Comb. Search SoCS 2014*, vol. 2014-Janua, no. SoCS, pp. 19–27, 2014.

- [73] Z. Bnaya and A. Felner, "Conflict-oriented windowed hierarchical cooperative A*," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 3743–3748.
- [74] E. Boyarski *et al.*, "ICBS: Improved conflict-based search algorithm for multi-agent pathfinding," *IJCAI Int. Jt. Conf. Artif. Intell.*, vol. 2015-Janua, no. Ijcai, pp. 740–746, 2015.
- [75] L. Cohen and S. Koenig, "Bounded suboptimal multi-agent path finding using highways," *IJCAI Int. Jt. Conf. Artif. Intell.*, vol. 2016-Janua, pp. 3978–3979, 2016.
- [76] S. S. Chouhan and R. Niyogi, "DiMPP: a complete distributed algorithm for multi-agent path planning," *J. Exp. Theor. Artif. Intell.*, vol. 29, no. 6, pp. 1129–1148, 2017, doi: 10.1080/0952813X.2017.1310142.
- [77] K. H. C. Wang and A. Botea, "Fast and memory-efficient multi-agent pathfinding," *ICAPS 2008 - Proc. 18th Int. Conf. Autom. Plan. Sched.*, no. Icaps, pp. 380–387, 2008.
- [78] M. Ryan, "Constraint-based multi-robot path planning," pp. 922–928, 2010.
- [79] P. Surynek, "A Novel Approach to Path Planning for Multiple Robots in Bi-connected Graphs," 2009.
- [80] G. Wagner and H. Choset, "M*: A complete multirobot path planning algorithm with optimality bounds," *Lect. Notes Electr. Eng.*, vol. 57 LNEE, pp. 3260–3267, 2011, doi: 10.1007/978-3-642-33971-4_10.
- [81] G. Sharon, R. Stern, A. Felner, and N. R. Sturtevant, "Conflict-based search for optimal multi-agent pathfinding," *Artif. Intell.*, vol. 219, pp. 40–66, 2015, doi: 10.1016/j.artint.2014.11.006.
- [82] T. Standley and R. Korf, "Complete algorithms for cooperative pathfinding problems," *IJCAI Int. Jt. Conf. Artif. Intell.*, pp. 668–673, 2011, doi: 10.5591/978-1-57735-516-8/IJCAI11-118.
- [83] B. de Wilde, A. W. ter Mors, and C. Witteveen, "Push and Rotate: Cooperative Multi-agent Path Planning," in *Proceedings of the 2013 International Conference on Autonomous Agents and Multi-agent Systems*, Richland, SC, 2013, pp. 87–94, [Online]. Available: <http://dl.acm.org/citation.cfm?id=2484920.2484938>.
- [84] M. M. Khorshid, R. C. Holte, and N. Sturtevant, "A polynomial-time algorithm for non-optimal multi-agent pathfinding," *Proc. 4th Annu. Symp. Comb. Search SoCS 2011*, pp. 76–83, 2011.
- [85] Q. Sajid, R. Luna, and K. E. Bekris, "Multi-agent pathfinding with simultaneous execution of single-agent primitives," *Proc. 5th Annu. Symp. Comb. Search SoCS 2012*, pp. 88–96, 2012.
- [86] B. DeWilde, A. W. Ter Mors, and C. Witteveen, "Push and Rotate: A complete Multi-agent Pathfinding algorithm," *J. Artif. Intell. Res.*, vol. 51, pp. 443–492, 2014, doi: 10.1613/jair.4447.
- [87] M. Ryan, "Exploiting subgraph structure in multi-robot path planning," *J. Artif. Intell. Res.*, vol. 31, pp. 497–542, 2008, doi: 10.1613/jair.2408.
- [88] P. Surynek, "On propositional encodings of cooperative path-finding," *Proc. - Int. Conf. Tools Artif. Intell. ICTAI*, vol. 1, pp. 524–531, 2012, doi: 10.1109/ICTAI.2012.77.
- [89] P. Surynek, "Makespan Optimal Solving of Cooperative Path-Finding via Reductions to Propositional," pp. 1–40, 2016.
- [90] P. Surynek, A. Felner, R. Stern, and E. Boyarski, "Modifying optimal SAT-based approach to multi-agent path-finding problem to suboptimal variants," *Proc. 10th Annu. Symp. Comb. Search SoCS 2017*, vol. 2017-Janua, pp. 169–170, 2017.
- [91] G. Wagner, *Subdimensional Expansion: A Framework for Computationally Tractable Multirobot Path Planning*. 2015.
- [92] G. Sharon, R. Stern, M. Goldenberg, and A. Felner, "The Increasing Cost Tree Search for Optimal Multi-Agent Pathfinding," *Proc. 4th Annu. Symp. Comb. Search SoCS 2011*, vol. 2, no. i, pp. 150–157, 2011.
- [93] G. Sharon, R. Stern, A. Felner, and N. Sturtevant, "Conflict-based search for optimal multi-agent path finding," *Proc. 5th Annu. Symp. Comb. Search SoCS 2012*, pp. 97–104, 2012.

- [94] H. Ma, D. Harabor, P. J. Stuckey, J. Li, and S. Koenig, "Searching with consistent prioritization for multi-agent path finding," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019, vol. 33, no. 01, pp. 7643–7650.
- [95] L. Cohen, T. Uras, and S. Koenig, "Feasibility Study: Using Highways for Bounded-Suboptimal Multi-Agent Path Finding," *Proc. 8th Annu. Symp. Comb. Search SoCS 2015*, vol. 2015-Janua, pp. 2–8, 2015.
- [96] L. Cohen, T. Uras, T. K. Satish Kumar, H. Xu, N. Ayanian, and S. Koenig, "Improved solvers for bounded-suboptimal multi-agent path finding," *IJCAI Int. Jt. Conf. Artif. Intell.*, vol. 2016-Janua, pp. 3067–3074, 2016.
- [97] J. Li, A. Tinka, S. Kiesel, J. W. Durham, T. K. S. Kumar, and S. Koenig, "Lifelong Multi-Agent Path Finding in Large-Scale Warehouses," *AAAI*, Mar. 2021, [Online]. Available: <http://arxiv.org/abs/2005.07371>.
- [98] S. M. LaValle, "Rapidly-exploring random trees: A new tool for path planning," 1998.
- [99] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *Int. J. Robot. Res.*, vol. 30, no. 7, pp. 846–894, 2011.
- [100] M. Cáp, P. Novák, J. Vokřínek, and M. Pechoucek, "Multi-agent RRT*: Sampling-based Cooperative Pathfinding," *CoRR*, vol. abs/1302.2, 2013, [Online]. Available: <http://arxiv.org/abs/1302.2828>.
- [101] M. Soltani, S. Ahmadi, A. Akgunduz, and N. Bhuiyan, "An eco-friendly aircraft taxiing approach with collision and conflict avoidance," *Transp. Res. Part C Emerg. Technol.*, vol. 121, p. 102872, Dec. 2020, doi: 10.1016/j.trc.2020.102872.
- [102] J. Y. Du, J. O. Brunner, and R. Kolisch, "Obtaining the optimal fleet mix: A case study about towing tractors at airports," *Omega*, vol. 64, pp. 102–114, Oct. 2016, doi: 10.1016/j.omega.2015.11.005.
- [103] X. Han, P. Zhao, Q. Meng, S. Yin, and D. Wan, "Optimal scheduling of airport ferry vehicles based on capacity network," *Ann. Oper. Res.*, vol. 295, no. 1, pp. 163–182, Dec. 2020, doi: 10.1007/s10479-020-03743-0.
- [104] P. X. Zhao, W. Q. Gao, X. Han, and W. H. Luo, "Bi-Objective Collaborative Scheduling Optimization of Airport Ferry Vehicle and Tractor," *Int. J. Simul. Model.*, vol. 18, no. 2, pp. 355–365, Jun. 2019, doi: 10.2507/IJSIMM18(2)CO9.
- [105] M. Schiffer and G. Walther, "The electric location routing problem with time windows and partial recharging," *Eur. J. Oper. Res.*, vol. 260, no. 3, pp. 995–1013, Aug. 2017, doi: 10.1016/j.ejor.2017.01.011.
- [106] G. Hiermann, J. Puchinger, S. Ropke, and R. F. Hartl, "The Electric Fleet Size and Mix Vehicle Routing Problem with Time Windows and Recharging Stations," *Eur. J. Oper. Res.*, vol. 252, no. 3, pp. 995–1018, Aug. 2016, doi: 10.1016/j.ejor.2016.01.038.
- [107] P. C. Roling and H. G. Visser, "Optimal airport surface traffic planning using mixed-integer linear programming," *Int. J. Aerosp. Eng.*, vol. 2008, no. 1, p. 1:1-1:11, Jan. 2008, doi: 10.1155/2008/732828.
- [108] E. v. Baaren and P. C. Roling, "Design of a zero emission aircraft towing system," in *AIAA Aviation 2019 Forum*, 0 vols., American Institute of Aeronautics and Astronautics, 2019.
- [109] S. Zaninotto, J. Gauci, G. Farrugia, and J. Debattista, "Design of a Human-in-the-Loop Aircraft Taxi Optimisation System Using Autonomous Tow Trucks," in *AIAA Aviation 2019 Forum*, 0 vols., American Institute of Aeronautics and Astronautics, 2019.
- [110] W. Hönl, S. Kiesel, A. Tinka, J. Durham, and N. Ayanian, "Conflict-based search with optimal task assignment," 2018.