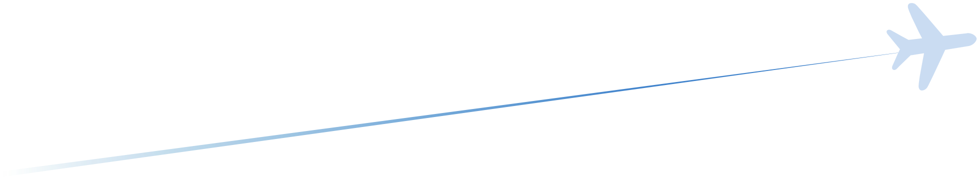


E-UI design doc & demonstrator

Deliverable ID:	**D4.1**
Dissemination Level:	**Public**
Project Acronym:	**MAHALO**
Grant:	**892970**
Call:	**H2020-SESAR-2019-2**
Topic:	**SESAR-ER4-01-2019**
Consortium Coordinator:	**DBL**
Edition Date:	**2 September 2021**
Edition:	**00.02.00**
Template Edition:	**02.00.02**

Founding Members
EUROPEAN UNION
EUROCONTROL

SESAR
JOINT UNDERTAKING

Authoring & Approval

Authors of the document

Name/Beneficiary	Position/Title	Date
Clark Borst/TUD	WP4 Leader	20/05/21

Reviewers internal to the project

Name/Beneficiary	Position/Title	Date
Brian Hilburn/CHPR	WP2 Leader	24/05/2021
Carl Westin/LiU	WP6 Leader	17/05/2021
Erik-Jan van Kampen/TUD	WP5 Leader	17/05/2021
Tiago Nunes/TUD	Project Contributor	17/05/2021
Supathida Boonsong/LFV	Project Contributor	18/05/2021
Magnus Bång/LiU	WP3 Leader	24/05/21
Matteo Cocchioni/DBL	Project Contributor	24/05/21
Stefano Bonelli/DBL	Project Coordinator	24/05/21
Stefano Bonelli/DBL	Project Coordinator	02/09/21

Approved for submission to the SJU By — Representatives of beneficiaries involved in the project

Name/Beneficiary	Position/Title	Date
Stefano Bonelli/DBL	Project Coordinator	28/05/21
Stefano Bonelli/DBL	Project Coordinator	02/09/21

Rejected By - Representatives of beneficiaries involved in the project

Name/Beneficiary	Position/Title	Date

Document History

Edition	Date	Status	Author	Justification
00.00.01	26/04/21	Draft	Clark Borst	Sections drafted
00.00.02	10/05/21	Draft	Clark Borst	Full draft
00.00.03	20/05/21	Draft	Clark Borst & Carl Westin	Revision
00.00.04	24/05/21	Final draft	Clark Borst & Brian Hilburn	Revision
00.01.00	28/05/21	Final release	Stefano Bonelli	Deliverable approved for submission
00.01.01	06/08/21	Final Draft	Clark Borst	Reopen for revision
00.02.00	02/09/21	Final release	Stefano Bonelli	Deliverable approved for submission

MAHALO

MODERN ATM VIA HUMAN / AUTOMATION LEARNING OPTIMISATION

This deliverable is part of a project that has received funding from the SESAR Joint Undertaking under grant agreement No 892970 under European Union's Horizon 2020 research and innovation programme.

Abstract

This document is the *E-UI design* report, deliverable D4.1 of the MAHALO project. This report reflects the output of MAHALO Tasks 4.1 and 4.2, and builds on the earlier D2.1 *Integrated State of the Art Report* and D2.2 *Concept report*.

This report details the MAHALO Ecological User Interface (E-UI), which serves as a common ground/ *shared mental model* between the human and automated machine learning (ML) agents acting in the same airspace environment. In particular, the interface aims to add *domain* and *agent* transparency to the system, which should enable the human controller to understand what the ML agent is doing and manually intervene if necessary or desired. Ecological Interface Design (EID) is used as a design framework for achieving the shared mental model for two reasons. First, EID puts the emphasis on visualising the physical laws and principles governing the ATC work domain (i.e., "domain" transparency), which bounds all actions that can be undertaken by both human and automated agents. Second, previous research initiatives in which several MAHALO consortium members were involved (e.g., SESAR WP-E MUFASA and C-SHARE) produced EID designs for ATC that will serve as starting points for the MAHALO E-UI.

In MAHALO, previous EID designs for "domain" transparency have been adapted and new visualisations have been created to address "agent" transparency, which should explain the inputs, outputs and inner process of how (and when) ML automation makes its decisions. Note that this report contains details regarding the *initial* E-UI design and is subjected to minor changes and/or enhancements in Work Package 5 (*Integration*) efforts.

In addition to this report and as part of the D4.1, a non-interactive video playback of the E-UI for demonstration purposes has been developed. The demonstrator [24] can be found via a link listed in the References.

ACRONYMS

ADS-B	Automatic Dependent Surveillance – Broadcast
AH	Abstraction Hierarchy
AI	Artificial Intelligence
ATC	Air Traffic Control
ATCo	Air Traffic Controller
ATM	Air Traffic Management
BADA	Base of Aircraft Data
CD&R	Conflict Detection and Resolution
CFL	Cleared Flight Level
COPX	Exit change-over point
CTA	Control Task Analysis
CWA	Cognitive Work Analysis
E-UI	Ecological User Interface
EID	Ecological Interface Design
FL	Flight Level
FIM	Flight Information Message
FMS	Flight Management System
GS	Groundspeed
HDG	Heading
IAS	Indicated Airspeed
JCS	Joint Cognitive System
MAHALO	Modernising ATM via Human-Automation Learning Optimisation

ML	Machine Learning
MUAC	Maastricht Upper Airspace Centre
MUFASA	Multidimensional Framework for Advanced SESAR Automation
PVD	Plan View Display
RTE	Route
SSD	Solution Space Diagram
SSR	Secondary Surveillance Radar
STCA	Short-Term Collision Alerting
TAS	True Airspeed
TBO	Trajectory-Based Operations
TSR	Trajectory Space Representation
UI	User Interface
VERA	Verification of Separation and Resolution Advisory
WDA	Work Domain Analysis

Table of Contents

1. Introduction

This document is the E-UI design report, deliverable D4.1 of the MAHALO project. This report reflects the output of MAHALO Tasks 4.1 and 4.2, and builds on the earlier D2.1 Integrated State of the Art Report and D2.2 Concept report.

In MAHALO, the goal of the E-UI is two-fold. First, to provide a realistic simulation platform for integration with the ML system and empirically investigate the balance between the key concepts of Conformance and Transparency. Second, to contribute to the transparency of the proposed ML system, where transparency is defined as "*the extent to which aspects of the automation's inner process underlying a solution can be observed and explained in human terms*." Transparency could help to foster *acceptance* of machine-generated solutions given that the rejection of machine advice has been reported as one of the key factors hindering the transition towards higher levels of automation in ATC [1]. In MAHALO, the main approach to achieve system transparency is by focusing first and foremost on achieving *domain* transparency by adopting design principles of the Ecological Interface Design (EID) framework. Second, *agent* transparency will be added by enhancing the ecological visualisations with information regarding details that disclose the decision-making rationality guiding the behaviour of the (ML) automation.

For air traffic controllers, who like to have a clean and uncluttered radar screen, "minimalistic" visual feedback on the automation's capabilities, limitations, activities and intentions are sought that allow controllers to effectively supervise automation and to decide when to intervene (i.e., take over control). As such, the E-UI design challenge is to find the right balance between complexity and (practical) usability that will lead to high system understanding and acceptance at acceptable workload levels.

This main goal of this report is to present the design details of the *initial* E-UI, in particular how the principles underlying EID have been used to create the visualisations. The visualisations have, for most part, been integrated in the ATC simulator software (i.e., SectorX) that will be used in the *Simulation* trials (WP6). The objective of the simulation trials is to study the impact of the transparency visualisations and personalised advisory systems (in conjunction with the ML models) on acceptance and system understanding.

1.1 Report structure

The report consists of four chapters. Chapter 1 contains a brief introduction to the report and its goals. Chapter 2 provides an overview of the (theoretical) design considerations/requirements underpinning domain and agent transparency. Chapter 3 details the results of a Cognitive Work Analysis that has been conducted for the considered ATC task within the scope of the MAHALO project. Chapter 4 details the initial version of the proposed E-UI design that cover both domain and agent transparency.

2. Domain and Agent Transparency

2.1 Joint Cognitive System

Transparency generally refers to automation's ability to afford understanding and predictions about its behaviour. Within the context of MAHALO, transparency is defined as "*the extent to which aspects of the automation's inner process underlying a solution can be observed and explained in human terms."* As in many transportation domains, the primary source of information regarding the current and future state of a (dynamic) system is a visual interface. Therefore, visual interfaces are commonly seen as ways to provide system transparency. In fact, the higher-level objective of an interface is to serve as the primary communication means between human and computer-based agents acting within the same operational environment [2-5].

As illustrated in Fig 2.1, the human air traffic controller (ATCo) and the computer-based agent, who are both acting within the same air traffic environment (i.e., airspace sector), need to communicate and share information about the (current and future) state of the environment and the tasks that need to be performed. This concept is also known as a Joint Cognitive System (JCS) [6]. To achieve a JCS, it is important to carefully design the contents of the interface, because it must serve as a Shared Mental Model (or, common ground) between human and automated agents [5].

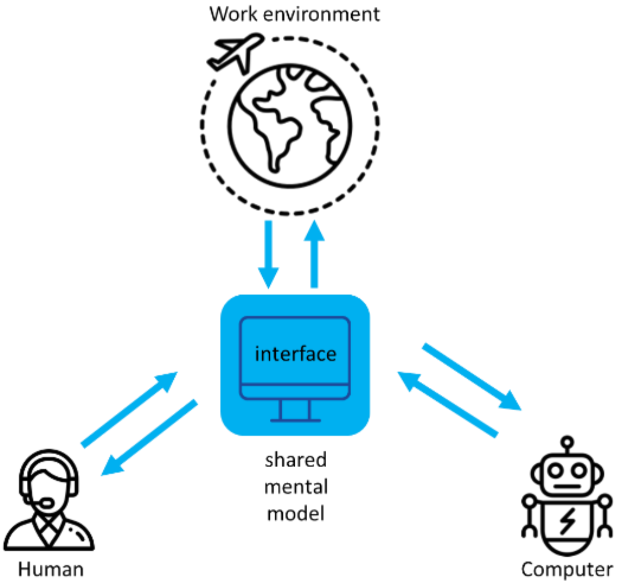


Fig. 2.1: The interface serving as a central element in connecting humans and machines with each other and the operational environment in which they act.

2.2 Domain transparency

In MAHALO, Ecological Interface Design (EID) has been chosen as it is hypothesized to serve as a good starting point for achieving a common ground between human and automated agents [7]. This choice is motivated by the fact that ecology-centred design puts the emphasis on visualising the natural boundaries for actions that are independent of the acting agent. In other words, actions are bounded by the physical laws governing the ATC work (e.g., aircraft climb and turn performances, speed envelopes, etc.), irrespective of being a human or computer-based agent.

Differently from user- and technology-centred approaches that are geared toward simplifying and supplanting human involvement, an ecology-centred approach strives for re-involvement by engineering engaging visual human-automation interfaces that reveal the normally invisible constraints underlying automation and human control expertise. Such constraints are first and foremost grounded in the physical laws that bound the behaviour of *all* agents acting with the work environment, irrespective of those agents being human or computer-based. In other words, ecological interfaces will provide *domain transparency* as it will give insights into the principles of the work domain.

Ecological interfaces typically visualise *solution spaces* that encapsulate and bound all safe actions that can be executed within the system, irrespective of their optimality (see Fig. 2.2). The overall idea is that when computer-based agents will propose (i.e., advisory) or perform an action within the work environment, plotting that advisory or action within the visualised solution space will enable a human operator to monitor the behaviour of the automation and judge the validity of the advice or action. When the operator does not agree with the advisory, he or she can use the same visualised solution space to formulate and execute an alternative action within the safe system envelope.

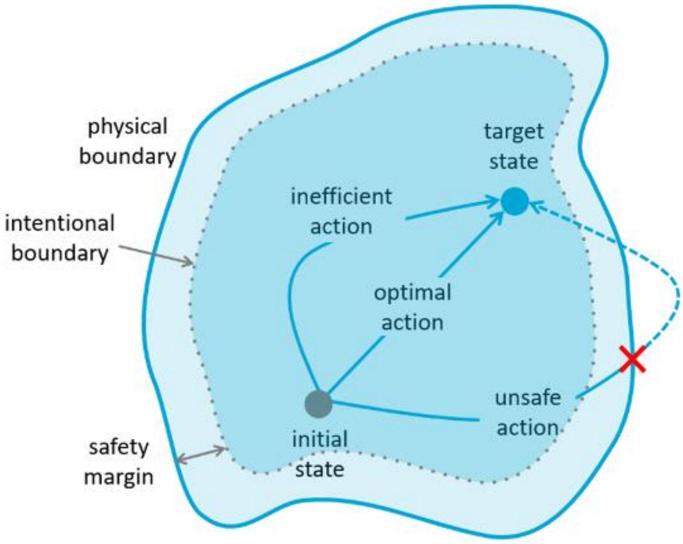


Fig 2.2: Abstract illustration of the solution space, bounded by physical and intentional constraints, encompassing all safe actions.

2.3 Agent transparency

Although the natural boundaries serve as a common ground between humans and automation, both agents may approach their control tasks in many different ways. A limitation of domain transparency is that it does not convey why another agent recommends a specific solution to a specific problem. As illustrated in Fig. 2.3, human actions are often governed by workload management principles that may result in safe solutions that satisfice, whereas computers tend to optimise safe solutions for efficiency purposes. To foster acceptance of automated agents, it is therefore important to also communicate the decision-making rationale underlying computer-generated advice or action. In other words, the interface must also disclose the constraints within the agent that would explain why a certain course of action within the available solution space has been chosen.

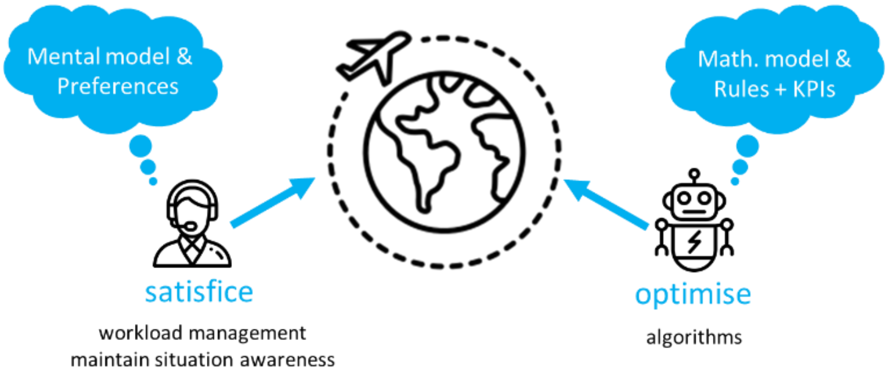


Fig. 2.3: Human and automated agents may approach their shared control tasks in different ways, despite their shared notion of the work domain.

Another way of looking at domain and agent transparency is by considering the stages and levels of automation devised by Parasuraman and colleagues [8]. This framework acknowledges the fact that automation can take place at different levels of information processing stages, similar to the stages of how humans typically make decisions. Before making any decision and action, information first needs to be acquired and then integrated into a comprehensive overview of the system's current, future and target state.

As shown in Fig. 2.4, EID is hypothesised to address the first two stages of automation, where automation acquires and integrates information into solution spaces. The final two stages, thus how a final decision and action will be undertaken, may differ per acting agent. When computer-based agents are not present at the levels of decision selection and action implementation, the human agent is responsible for deciding what to do and implement an action. In ATC, such solutions may already foster acceptance, because the integrated solution space does not dictate specifically what the ATCo should do and how, but leaves those decisions entirely in the hands of the human. Acceptance problems usually arise when automated agents start to mingle in the decision-making and execution process, where underlying agent settings and "cost functions" will steer decisions and actions into a direction that are misaligned with human-like decisions.

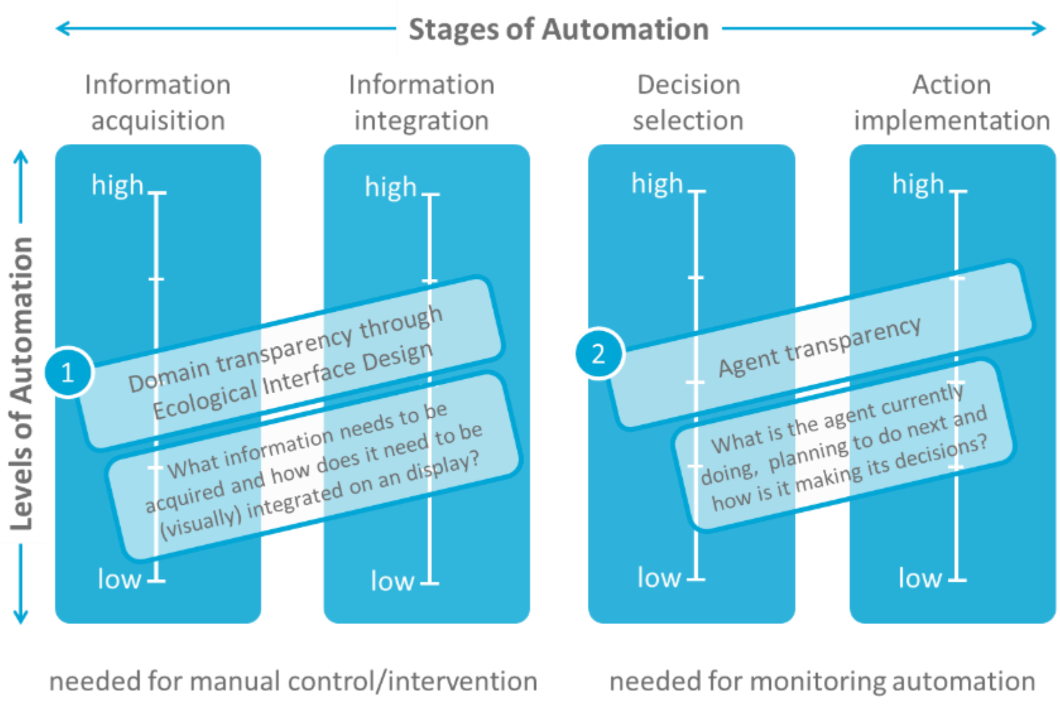


Fig. 2.4: Stages and levels of automation as a framework for domain and agent transparency.

Given that the ATC community wants to exploit the safety and efficiency benefits that computer-based automation can offer, one way to foster acceptance of more optimised solutions is by making the inner process of the automated agent transparent in a way that can be understood by humans. More specifically, when the automated agent could explain its decision-making rationale in the same "language" of a human ATCo, it is expected that acceptance of non-conformal solutions will be gained. Agent transparency can be expected to be most valuable in situations when the automation's decision (or recommended solution) contrasts that of the human ATCo. It follows that human understanding and acceptance of non-conformal solutions can benefit from explanations building on the why the proposed solution is preferred over that of the human. This is not a trivial task as it requires the automation to consider the specific human ATCo's decision making process and strategic preferences underlying a solution to a specific situation (i.e., conformance rationale). A suitable approach for exploring and providing the content of the agent transparency (specifically the conformance rationale and fit between automation's decision and human decision) can be found in ML interpretability methods. It falls on the E-UI to visualise these explanations in a suitable way.

3. Cognitive Work Analysis

The goal of a Cognitive Work Analysis (CWA) is to provide a structured analysis of work that can be used to inform design decisions (e.g., what to show on an interface) [9]. In the context of MAHALO, a CWA of the ATC control task can be used to uncover the shared "language" that will contribute to adding transparency to the system. In this chapter, the results of a partial CWA are reported with the aim to identify the required information and the ways the shared control tasks (i.e., independent of the acting agent) can be approached. Much information regarding the contents of the CWA has been taken from the public FASTI HF document [10] and EUROCONTROL's task analysis for en-route control [11].

3.1 Scope

Before diving into the CWA phases, it is important to define the scope of the system to be analysed. In MAHALO, the focus lies on tactical en-route, upper area control (ACC) with the following simplifying assumptions:

- A single ATCo is responsible for all flights in his/her airspace sector under control;
- Flights follow FMS routes and/or headings;
- Medium- and Short-Term Collision Alerting is available;
- Full ADS-B and SSR Mode S data sharing air-ground (aircraft state, meteo data, etc);
- Flights can only be controlled by issuing altitude and/or heading clearances, thus disregarding speed;
- Trajectory prediction is free of any uncertainties;
- Pilot delays and voice R/T are disregarded (assuming digital data links);
- Pilot requests are disregarded.

The main job of the ACC within the work environment under consideration is mainly *perturbation management*. Although airspace use and route-allocation will be structured and optimised beforehand to achieve optimal system performance in terms of safety, efficiency and productivity (i.e., SESAR objectives), it is the *unforeseen* separation provisions, sequencing, weather and changing airspace constraints which inevitably require (small, tactical) changes in the pre-planned trajectories. Figure 3.1 provides a graphical illustration of such an environment.

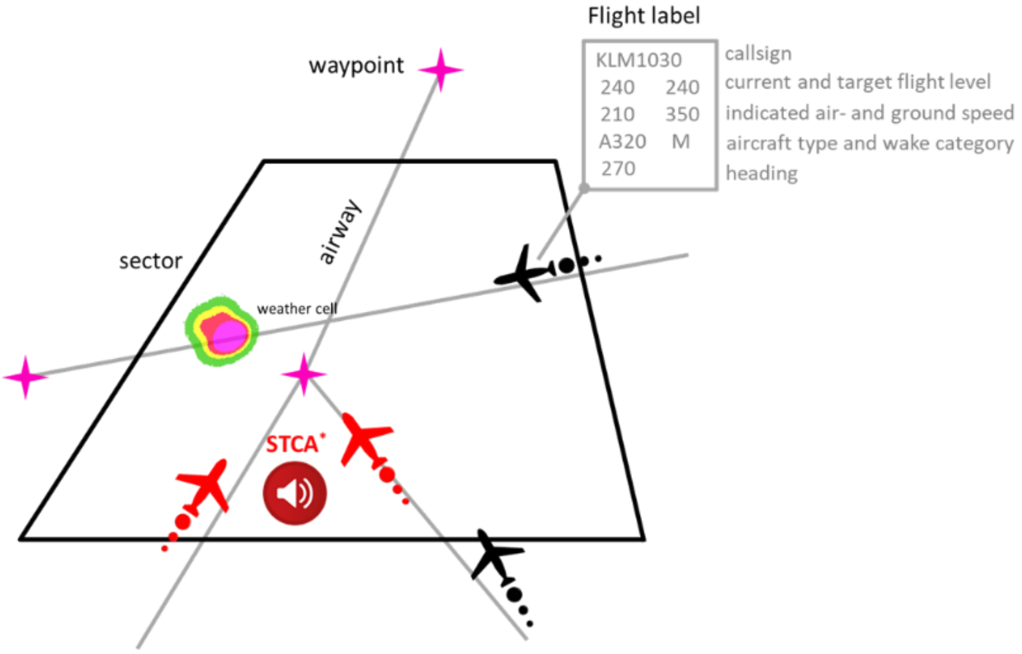


Fig 3.1: MAHALO operational environment.

3.2 Work Domain Analysis

The first step in a CWA is a Work Domain Analysis (WDA). The goal of a WDA is to obtain a structured and functional "map" of the workspace, independent of any acting agent (i.e., human or computer automation). In terms of transparency, the information gathered within the WDA is to contribute to *domain* transparency. The tool to summarise the results of a WDA is Rasmussen's Abstraction Hierarchy (AH), which commonly has five levels:

1. **Function Purpose** – what is the purpose of the system as a whole?
2. **Abstract Function** – what are the underlying laws (of physics) and principles governing the work?
3. **Generalised Function** – what processes and system choices are involved?
4. **Physical Function** – what are the function-bearing components?
5. **Physical Form** – what the locations, sizes and states of the components?

In Fig. 3.2 a generic AH can be seen. According to Rasmussen:

- Each level of the AH fully describes the system, but just at a different abstraction level;
- The hierarchy represents a psychological-relevant way how humans generally solve problems (top-down reasoning, based upon Rasmussen's empirical insights);

- The elements on one level must always be linked to elements found one level above and below (they are linked via `why-what-how’ means-ends relationships);
- It is a powerful critical thinking tool that helps to structure “engineering common sense.”

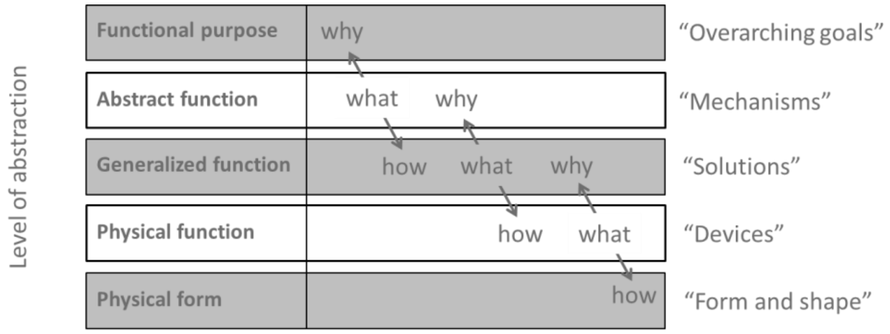


Fig. 3.2: Rasmussen’s Abstraction Hierarchy.

Based upon these principles, documentation on ATC [10-11], previous explorative research [20,23], results from interviews with ATCos and other domain experts [10-12], including taking into account our own experiences and expertise, the conducted WDA has resulted in the AH shown in Fig. 3.3. The importance of conducting the WDA and mapping the discovered constraints within an AH is that the AH helps to specify the contents of the interface in order to achieve domain transparency. In Chapter 4 it will be clarified how each element in the AH is represented in some form or shape onto the geometry of the MAHALO E-UI.

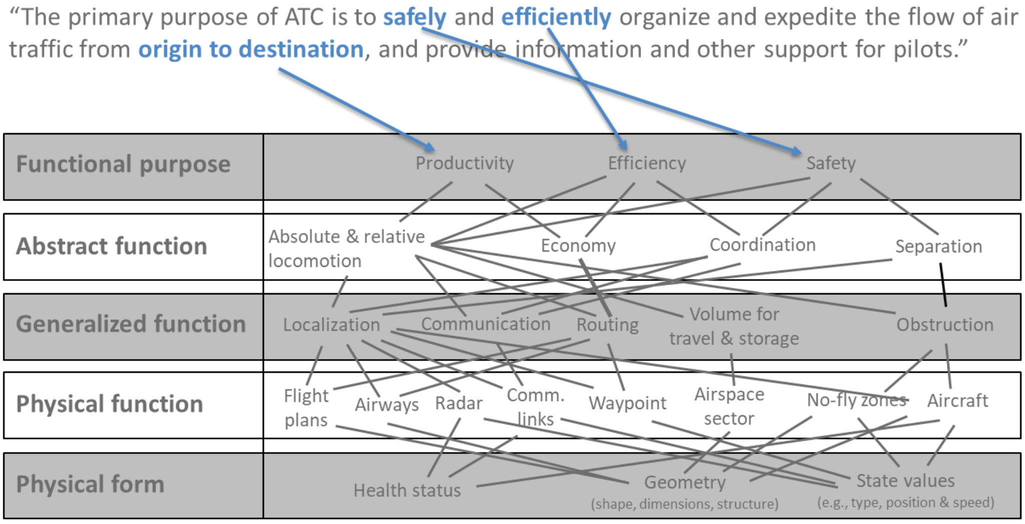


Fig. 3.3: Proposed AH for en-route ACC.

3.3 Control Task Analysis

The goal of the Control Tasks Analysis (CTA), the second step in the CWA, is to analyse the task(s) to perform within the work domain, irrespective of whom is eventually performing the task(s). Using the Cognitive Task Analysis described in the FASTI HF document [10] and given the MAHALO scope as detailed in Work Package 2 deliverables, seven generic control tasks have been identified that are independent of any acting agent:

- **Assume control** – assume control over new aircraft entering the sector;
- **Clear to target flight levels** – put aircraft to their target altitudes;
- **Routing to exit waypoints** – guide aircraft toward destination & minimise travel delays;
- **Conflict detection** – predict separation violation between aircraft & no-fly zones;
- **Conflict resolution** – rerouting by issuing altitude, route and/or heading clearances;
- **Conformance monitoring** – make sure aircraft follow instructions/clearances;
- **Hand over control** – transfer control over aircraft to adjacent sector.

Although these control tasks do not always have a fixed order in which they need to be executed (per flight), the consensus is that conflict detection and resolution (CD&R) lies at the centre, as illustrated in Fig. 3.4. That is, other tasks, such as “clear to target altitudes” and “clear to destinations” often involve iterative conflict detection (and resolution) activities to ensure safe and efficient operations at all times. The presentation order as visualised in Fig. 3.4 represents a nominal sequence of control tasks when flights enter, cross and are about to leave the airspace sector.

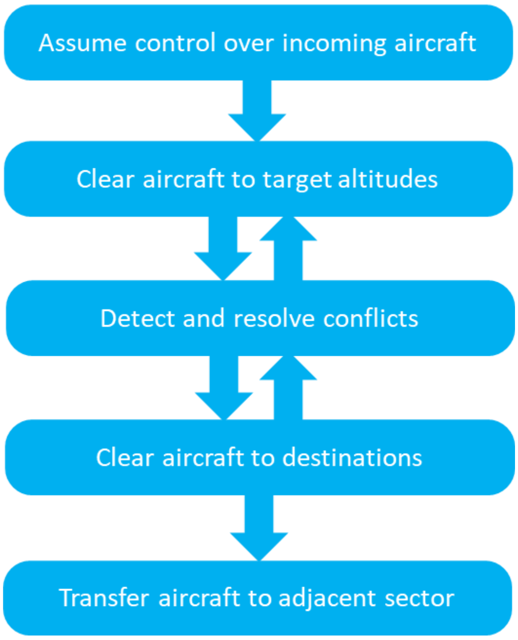


Fig. 3.4: ATC control tasks.

When focussing in more detail on the core CD&R task, several subtasks can be identified [9-10]:

Conflict detection[1]

- (Pairwise) Inspect and compare current, planned and cleared altitudes of flights;
- Focus on converging aircraft pairs;
- Consider crossing flight plans and (sector) merging points;
- Consider speeds and flight directions;
- Predict minimum separation distance (thresholds: 5 nm horizontally and 1,000 ft vertically);
- Interpret temporal information (i.e., urgency of conflict);
- Interpret conflict geometry.

Conflict resolution

- Solve conflicts pairwise;
- Consider solutions that lead to minimal sector disruption;
- Minimise the number of flights to 'move';
- Consider solutions in a hierarchical way: altitude first, then heading or route[2];
- Consider solutions close to the current or target states of flights (i.e., target altitude and sector exit waypoint)
- Consider proven and familiar solutions for specific conflict geometries.

In Chapter 4, details will be provided about the current and proposed supporting tools and visual cues that assist a human ATCo in performing these tasks manually and supervising automation performing these tasks.

3.4 Strategy Analysis

The control tasks discussed in the previous section can be executed in various different ways. The goal of the Strategies Analysis is to analyse how tasks within the work domain can be performed, again irrespective of who is performing the task. Strategies within certain tasks can be visualised by flow charts that, for example, illustrate a sequence of steps to be undertaken.

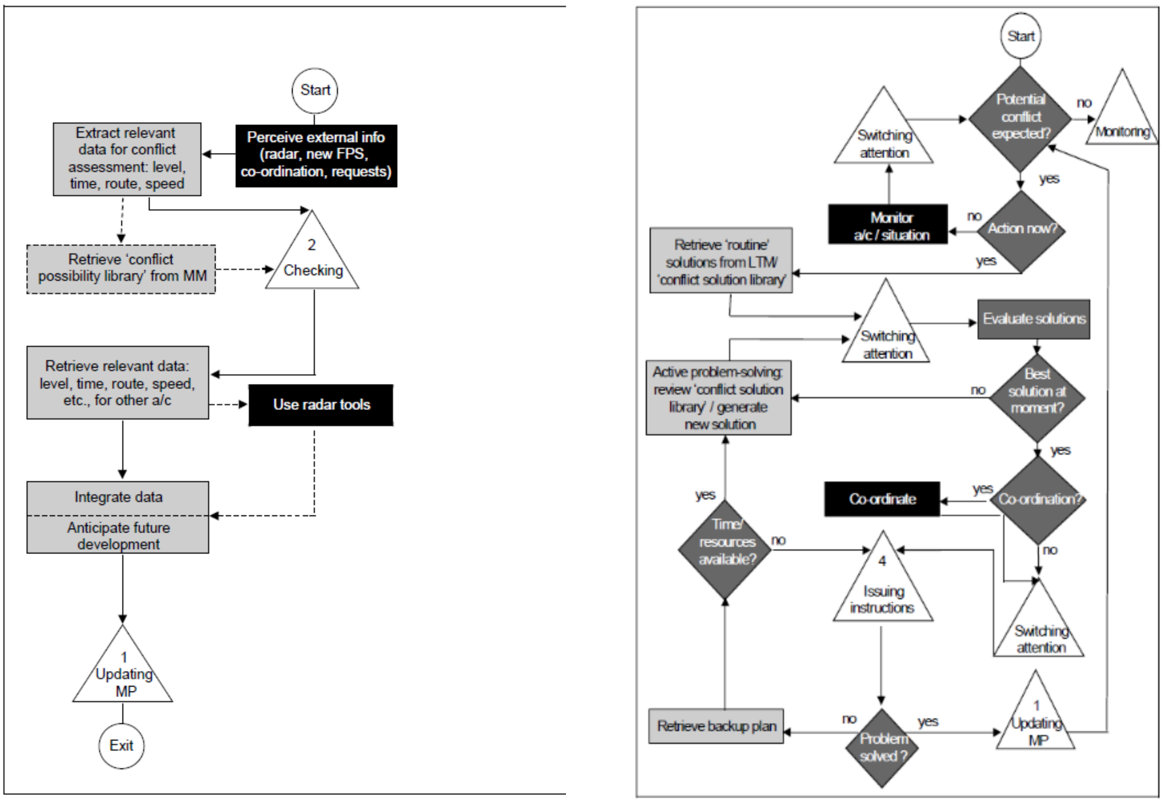
In 1999, EUROCONTROL conducted a very detailed integrated task analysis for en-route area control [11]. Although that analysis was approached from the perspective of a human ATCo, and was therefore agent specific, the document contains useful flow charts for analysing strategies. For example, the flow

---

[1] Within the scope of MAHALO, uncertainty in trajectory and wind predictions are not taken into account.

[2] Within the scope of MAHALO, speed solutions are not considered.

charts for the core ATC tasks (i.e., conflict detection and resolution) are shown in Fig. 3.5. The value of these flow charts is that if automation needs to communicate its CD&R activities to human ATCos in order to make the system understandable, it would make sense that automation approaches the detection and resolution activities in a way that is similar to human strategies.



a) conflict detection

b) conflict resolution

Fig. 3.5: Conflict detection and resolution flow charts, taken from [11].

The purpose of this document is not to repeat EUROCONTROL's task and strategy analysis, but rather to complement it with more specific strategy information, such as ATCo conflict resolution heuristics that could help to make automation more understandable and therefore more acceptable. A study conducted by Fothergill and Neal revealed 13 conflict resolution heuristics for en-route control, of which eight are considered relevant within the MAHALO context [12]:

Lateral heuristics (vectoring):

- **Point/vector behind other aircraft** (see Fig. 3.6). This heuristic involves vectoring one aircraft (preferably the slower aircraft or the one that is farthest in distance to the pair's crossing point) either directly at or just behind a potentially conflicting aircraft (that flies at a faster speed or is closer in distance to the pair's crossing point).
- **Direct away from potential conflicts.** This heuristic involves assigning an aircraft a short trajectory deviation from its planned route.

- **Parallel Track**. This heuristic involves assigning an aircraft a new trajectory, which is parallel to its current route.
- **Take out for five miles, then put back on track**. This heuristic involves assigning an aircraft a new trajectory, which will keep the aircraft five miles either right or left of its current route.

Vertical heuristics (level change):

- **Cut off at nearest available level on climb.** This heuristic involves amending an aircraft's cleared flight level to the nearest available level on its climb.
- **Cut off at highest possible level on climb** (see Fig. 3.6). This heuristic involves assigning a new cleared flight level to an aircraft, which is the highest vacant level the aircraft can reach by the crossing point with the potentially conflicting aircraft.
- **Descend to nearest available level**. This heuristic involves assigning an aircraft to the nearest available level (in intervals of 1,000 ft). This heuristic poses a penalty for the aircraft. By descending to a lower level, the aircraft will use more fuel than it would at higher altitudes.
- **Step climb/descent** (see Fig. 3.6). This heuristic involves incrementally amending an aircraft's cleared flight level in intervals of 1,000 ft to 'step' it past the potentially conflicting aircraft using vertically separated levels.

Although these heuristics may be applied independently from any acting agent, their impact on human workload will differ. For example, the 'vector behind' heuristic is considered a "set and forget" strategy that will require low workload in both execution (only one heading change) and required monitoring (after the heading change). The 'step climb/descent' heuristic is more labour intensive, requiring multiple clearances and more frequent monitoring cycles.

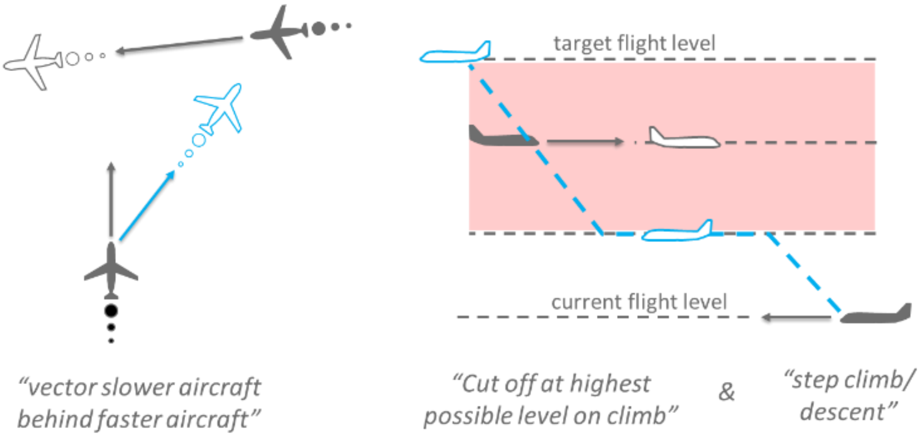


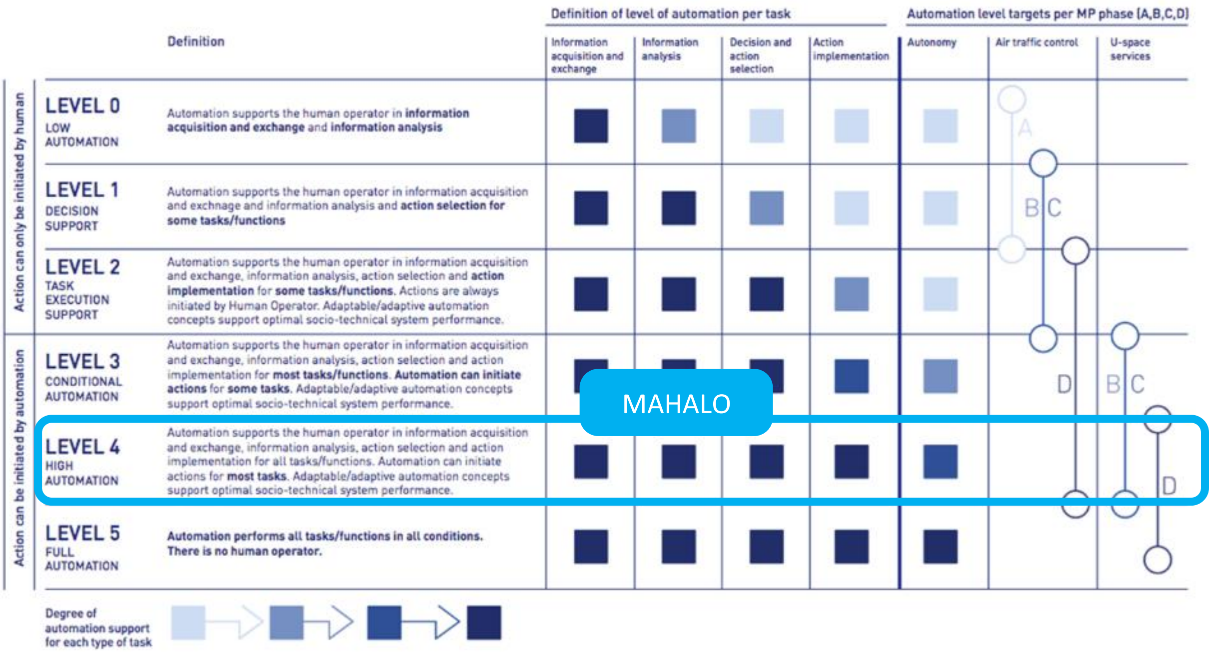
Fig. 3.6: Example conflict resolution heuristics.

3.5 Social Organisation and Cooperation Analysis

The en-route ATC control tasks and strategy heuristics can be done by either a human or computer-based agent, or shared between the two and/or a team of human ATCos (e.g., the planner and executive controller). Regardless of the specific team setting, a Social Organisation and Cooperation Analysis (SOCA) can help to allocate tasks between team members. In MAHALO, the focus on human-automation teaming lies on a single executive controller collaborating with automation.

In human-automation teaming, it is often considered advantageous that routine tasks, featuring low levels of uncertainty and complexity, are allocated to automation and let humans focus on the real, challenging work to do what they are the best at, namely creativity and adaptivity/flexibility in control [7]. In the light of the control tasks mentioned in Section 3.3, routine tasks are "assuming" and "transferring" flights, whereas CD&R may require creative problem-solving activities in non-standard traffic situations. Assuming and transferring flights can therefore be allocated to computer automation, whereas CD&R activities may occasionally need to be handled by humans, warranted by situational demands.

In MAHALO, however, we seek a more highly-automated ATC environment where humans need to primarily monitor the behaviour of automation performing all tasks, whether routine or non-routine. This focus is aligned with the "Level 4" solution according to the latest ATM Master Plan (see Fig. 3.7). By excluding complexity factors such as uncertainties in trajectory predictions, weather forecasts and pilot responses, the focus of MAHALO will be exclusively on studying the impact of domain and agent transparency on the acceptance, trust, workload and system understanding of a highly-automated ATC environment.



	Definition	Information acquisition and exchange	Information analysis	Decision and action selection	Action implementation	Autonomy
LEVEL 0 LOW AUTOMATION	Automation supports the human operator in **information acquisition and exchange** and **information analysis**	█	▒	░	░	░
LEVEL 1 DECISION SUPPORT	Automation supports the human operator in information acquisition and exchnage and information analysis and **action selection for some tasks/functions**	█	█	▒	░	░
LEVEL 2 TASK EXECUTION SUPPORT	Automation supports the human operator in information acquisition and exchange, information analysis, action selection and **action implementation** for **some tasks/functions.** Actions are always initiated by Human Operator. Adaptable/adaptive automation concepts support optimal socio-technical system performance.	█	█	█	▒	░
LEVEL 3 CONDITIONAL AUTOMATION	Automation supports the human operator in information acquisition and exchange, information analysis, action selection and action implementation for **most tasks/functions. Automation can initiate actions** for **some tasks.** Adaptable/adaptive automation concepts support optimal socio-technical system performance.	█	█	█	▓	▒
LEVEL 4 HIGH AUTOMATION	Automation supports the human operator in information acquisition and exchange, information analysis, action selection and action implementation for all tasks/functions. Automation can initiate actions for **most tasks.** Adaptable/adaptive automation concepts support optimal socio-technical system performance.	█	█	█	█	▓
LEVEL 5 FULL AUTOMATION	**Automation performs all tasks/functions in all conditions. There is no human operator.**	█	█	█	█	█

Degree of automation support for each type of task: ░ → ▒ → ▓ → █

Definition of level of automation per task

Fig. 3.7: Levels of Automation, taken from the 2020 ATM Master Plan [13].

4. Visualisations and Implementations

The purpose of the Ecological User Interface (E-UI) is to provide both domain transparency and (automated) agent transparency in a visual way. By following the principles underlying the EID framework, the E-UI will first and foremost portray the constraints and their relationships within the ATC work domain. In terms of visualisations and implementations, MAHALO performs evolutionary design by enhancing/augmenting existing interfaces that have proven their value in both industry and academia.

4.1 Baseline Plan View Display

The ATCo's primary source of information is Plan View Display (PVD), the electronic radar screen. The PVD has been chosen as a baseline interface, because it already portrays relevant *domain* information. For example, it portrays the geometry of the sector and the positions and states of flights by the plotted blips and flight label information, respectively. Over the past decades, industry and academia have integrated many useful decision-support tools in the PVD to assist controllers in their control tasks.

For MAHALO, SectorX[3] has been adapted to mimic the state-of-the-art PVD of a controller working position (CWP) found at Maastricht Upper Area Control (MUAC), see Fig. 4.1. For MAHALO purposes, a reduced MUAC toolset has been integrated in SectorX that are deemed critical in supporting the ATC control tasks as discussed in Section 3.3:

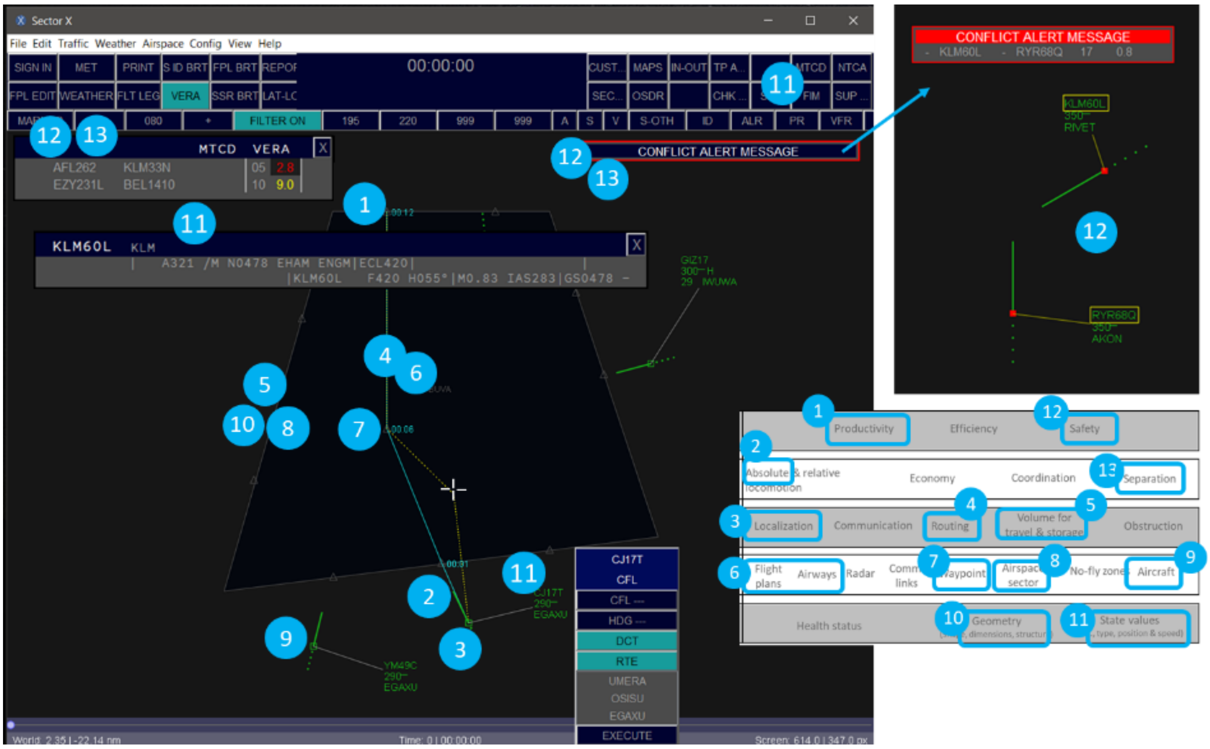
- **VERA** (Verification of Separation and Resolution Advisory): used for inspecting current and future separation properties (i.e., time- and distance-to-closest-point-of approach) between flight pairs.
- **FIM** (Flight Information Message): used for gathering additional flight information that is not shown in the flight labels (e.g., flight type, indicated airspeed and ground speed, wake category, origin and destination airport, etc.).
- **Conflict alert table**: STCA information regarding time and closest approach distance between *urgent* conflict pairs.

In Fig. 4.2, the results of an analysis are shown that indicate how each visualised PVD element represents the work domain constraints found in the AH portrayed in Fig. 3.3.

---

[3] JAVA-based medium fidelity ATC research simulator, developed by the TUD MAHALO consortium members.

Fig. 4.1: State-of-the-art Plan View Display (PVD), as implemented in SectorX.



AFL262 KLM33N 05 2.8
EZY231L BEL1410 10 9.0

KLM60L KLM
| A321 /M N0478 EHAM ENGM|ECL420| |
|KLM60L F420 H055°|M0.83 IAS283|GS0478 -

CONFLICT ALERT MESSAGE
- KLM60L - RYR68Q 17 0.8

Fig. 4.2: Analysis of the PVD elements in terms of (work) domain representation.

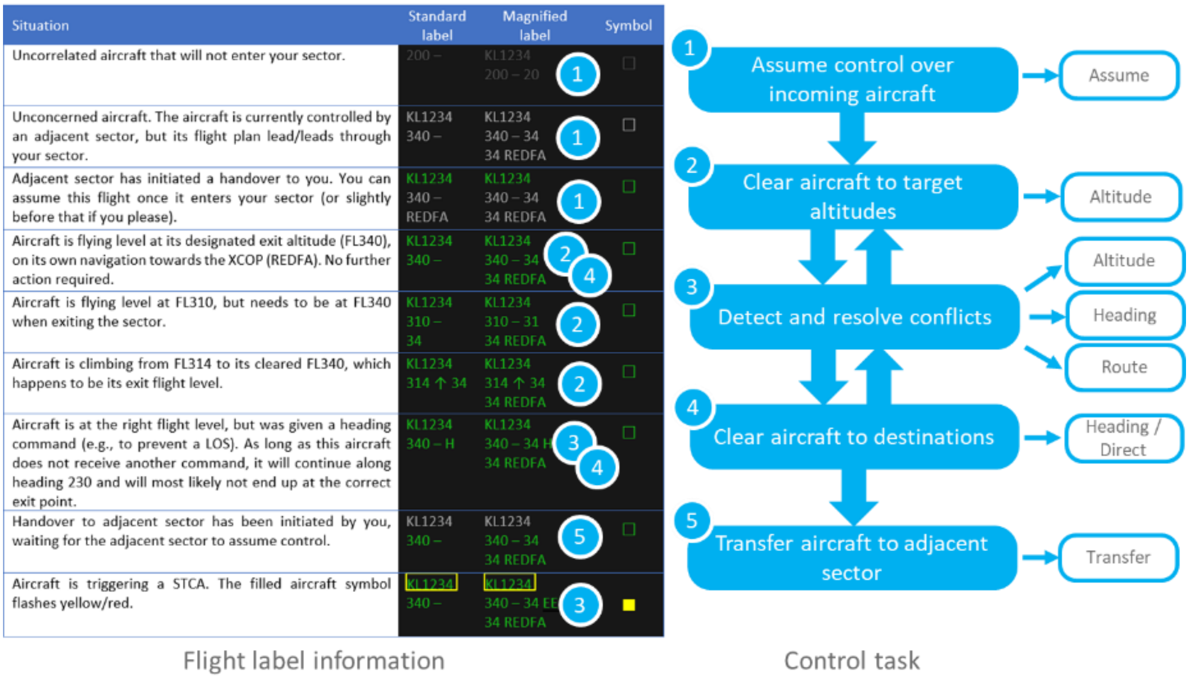
From Fig. 4.2 it is clear that many *domain* elements are already visualised on a state-of-the-art PVD, which makes it an ideal candidate to augment it for any missing links that are not yet fully represented. Such missing links will be addressed in Section 4.2. Note that certain domain elements are beyond scope of the MAHALO project, for example, the 'health status' of systems, quality of communication links, etc.

Regarding the support for ATC control tasks discussed in Section 3.3, Fig. 4.3 portrays how 'blip' and label information and colours serve as (action) triggers for specific control tasks. The information and colours allow an ATCo to quickly scan the sector for action items. In general, a richer flight label (in terms of information and colours) corresponds to (more) outstanding action items per flight. When an action is required, for example when an aircraft is not yet cleared to its target flight level (e.g., see the label state in the fifth row in the table shown in Fig. 4.3), the ATCo is expected to clear that aircraft to its target altitude. Obviously, the ATCo will first need to judge whether it is safe to change altitude before issuing the clearance. That will involve conflict detection activities and subsequently a strategy (e.g., a "step climb") needs to be devised on how to get the aircraft (close) to its target altitude.

Once the ATCo has devised a plan and strategy to execute, control inputs can be given by clicking on the interactive label items that will open up the clearance menu. As shown in Fig. 4.4, the clearance menu is responsive to the specific label item that has been clicked:

- **Callsign**: opens a menu allowing the ATCo to initiate the VERA tool starting from that flight, assume control over the flight (when transferred from previous sector) or transfer it to the next sector;
- **Cleared Flight Level (CFL)**: opens the flight level menu and the mouse cursor will automatically snap to the target flight level in case it deviates from the current cleared level;
- **Heading (HDG)**: opens the heading clearance menu, allowing a controller input an absolute heading or a relative heading (i.e., nudging the flight certain degrees to the left or right of the current heading);
- **Exit change-over waypoint (COPX)**: opens the Route (RTE) menu, allowing a controller to modify the current route by inserting intermediate waypoints. Note that editing a route is only possible when the aircraft is flying on a route. If not (i.e., flying on a heading), clicking that label item will open the HDG menu instead. The RTE menu can also be used to put a flight on a direct trajectory toward the COPX, the so called Direct-To (DCT) clearance.

Given the simplifying assumptions made for MAHALO, where we disregard radio R/T and pilot response delays, it is assumed that ATCo label inputs are automatically uplinked to the aircraft via digital data links and are immediately executed by the autopilot. The manoeuvring behaviour of aircraft (i.e., climb/descent, turn and acceleration) is simulated with BADA performance models.



Situation	Standard label	Magnified label	Symbol
Uncorrelated aircraft that will not enter your sector.	200 –	KL1234 200 – 20 ①	☐
Unconcerned aircraft. The aircraft is currently controlled by an adjacent sector, but its flight plan lead/leads through your sector.	KL1234 340 –	KL1234 340 – 34 34 REDFA ①	☐
Adjacent sector has initiated a handover to you. You can assume this flight once it enters your sector (or slightly before that if you please).	KL1234 340 – REDFA	KL1234 340 – 34 34 REDFA ①	☐
Aircraft is flying level at its designated exit altitude (FL340), on its own navigation towards the XCOP (REDFA). No further action required.	KL1234 340 –	KL1234 340 – 34 34 REDFA ② ④	☐
Aircraft is flying level at FL310, but needs to be at FL340 when exiting the sector.	KL1234 310 – 34	KL1234 310 – 31 34 REDFA ②	☐
Aircraft is climbing from FL314 to its cleared FL340, which happens to be its exit flight level.	KL1234 314 ↑ 34	KL1234 314 ↑ 34 34 REDFA ②	☐
Aircraft is at the right flight level, but was given a heading command (e.g., to prevent a LOS). As long as this aircraft does not receive another command, it will continue along heading 230 and will most likely not end up at the correct exit point.	KL1234 340 – H	KL1234 340 – 34 H 34 REDFA ③ ④	☐
Handover to adjacent sector has been initiated by you, waiting for the adjacent sector to assume control.	KL1234 340 –	KL1234 340 – 34 34 REDFA ⑤	☐
Aircraft is triggering a STCA. The filled aircraft symbol flashes yellow/red.	KL1234 340 –	KL1234 340 – 34 EE 34 REDFA ③	■

Fig. 4.3: Connections between flight label (and blip) states to control tasks.

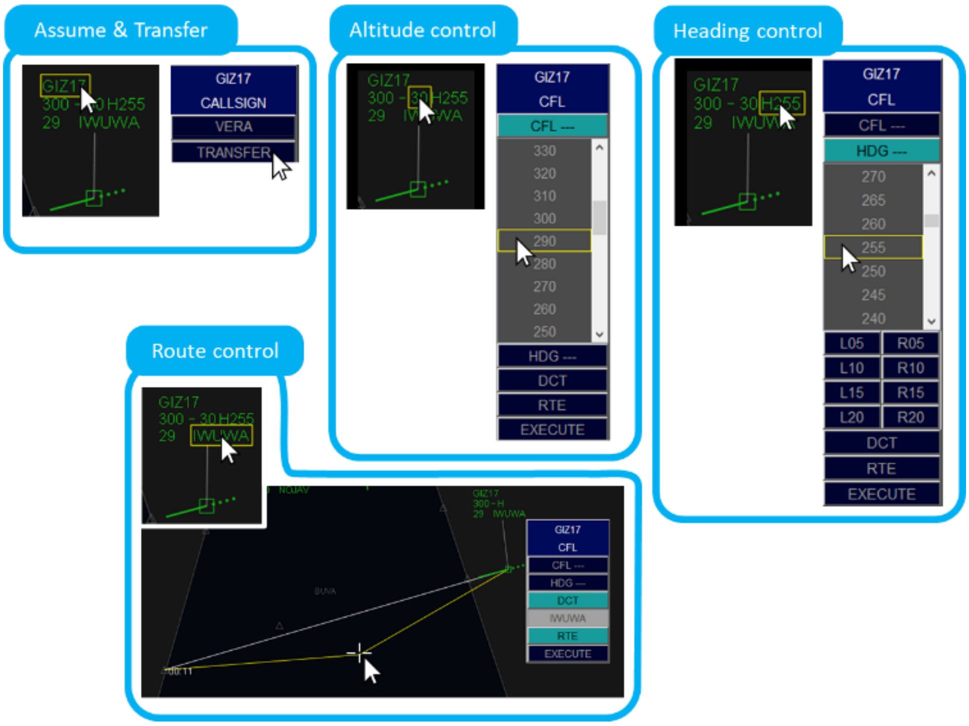


Fig. 4.4: The means to execute the ATC control tasks supported in MAHALO.

4.2 Domain transparency

As discussed in Section 4.1, the baseline PVD already represents many elements that add domain transparency and assist controllers in performing the different control tasks. Given that more domain insight regarding routine tasks, such as assuming and transferring flights, is not deemed critical, more domain insight could be given regarding the core CD&R task. Support in terms of conflict detection is offered by using the VERA tool on specific flight pairs and STCA alerting, the main focus of enhanced domain transparency lies in supporting conflict resolution. This requires integrating low-level flight state information (e.g., positions, headings and speeds) into (higher-level) integrated solution spaces that portray the boundaries for all actions that can be undertaken. Formulating such solution spaces directly into the ATCo's input state-space (i.e., altitude, heading and route) would enable a controller to gain a deeper insight into the structure of conflicts and opportunities for solutions.

4.2.1 Altitude domain

Conflicts and solutions in the altitude dimension are difficult to spot on the PVD, given the 2D representation. Altitude information is only conveyed in the flight labels. When aircraft are flying at the same flight level and are one crossing paths, or when STCA has issued an alert, ATCos will first direct their attention to those aircraft pairs. For aircraft that are on different flight levels, but need to be cleared to other flight levels, the ATCo wants to know if it is safe to clear aircraft to their target levels.

In SectorX, problems and solutions in the altitude dimension can be inspected by opening the CFL clearance menu. In Fig. 4.5 it is shown that the CFL menu has been augmented with red shadings that indicate problematic flight levels (i.e., altitudes along the route at which other aircraft are flying). Determining which flight levels should be shaded is the result of conflict prediction calculations as implemented in the STCA models devised by EUROCONTROL [14]. The predictions used for altitude shading, however, adopt a longer look-ahead time than STCA. Given trajectory prediction uncertainties and no consideration of wind effects, extending the look-ahead time is considered unproblematic.

For additional decision support, linking problematic flight levels to the flights that cause those problems is important. Such relationships would represent the 'means-ends' links between AH elements that allow a controller to perform top-down reasoning: diving deeper into the work domain structure to identify which element found at the lower level of the AH is responsible for blocking the locomotion state-space in the altitude domain. This linking is supported by the hovering the mouse cursor over the problematic flight levels, which will highlight the callsigns of the problematic flights.

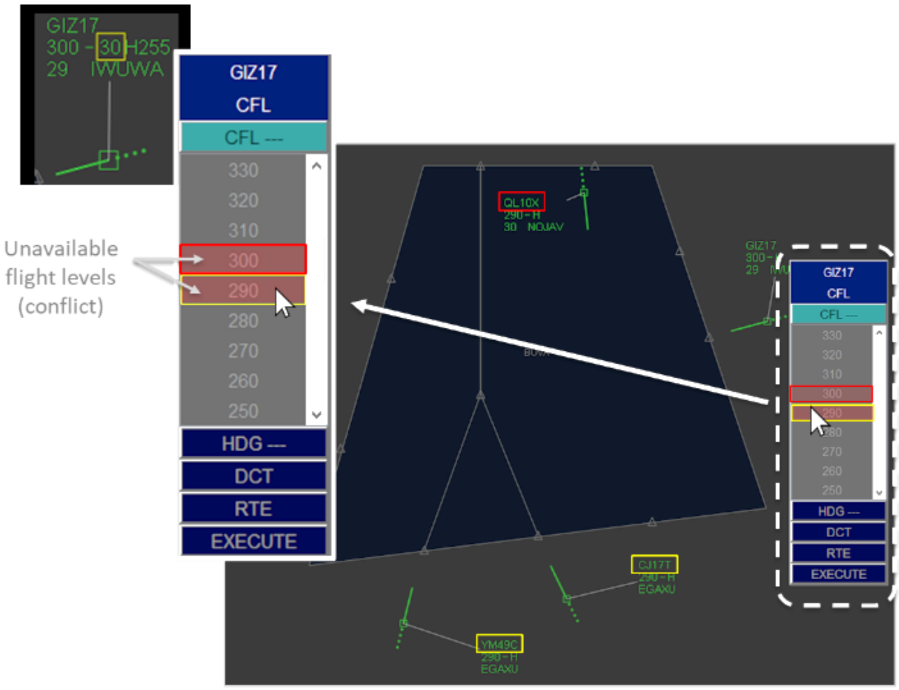


Fig. 4.5: Altitude solution space.

4.2.2 Heading domain

Similar to the blocked flight levels in the CFL menu shading, problematic headings can be shaded in the HDG menu as shown in Fig. 4.6. Additionally, activating the HDG menu will make the Solution Space Diagram (SSD) visible [15-19]. The SSD is centred around the controlled flight and portrays *all* problematic heading and speed states within the entire speed envelope of the controlled flight. This way, the ATCo can simultaneously observe the “go” and “no-go” states and thus use this information to formulate and implement a solution.

The problematic (combined) heading and speed states have the visual form of a triangle due to the geometric nature of the problem, as shown in Fig. 4.7. Given that the relative velocity between flight pairs is encoded inside the triangle, times at which separation will be lost (t_{los}) can be calculated using the distance between the aircraft. With this information, conflict urgency can be conveyed by shading the triangle with different colours (e.g., red represents t_{los} between now and 60 seconds, amber represents t_{los} between 60 and 120 seconds, etc.). Also note that other relevant CD&R information can be extracted from the SSD, such as: the width (α) of the triangle relates to the proximity between the controlled and observed flights, placing the speed vector outside the triangle solves the conflict and gives information about the controlled aircraft flying either behind or in front of the observed aircraft, etc.

TUD has developed and studied the SSD and its usefulness as a decision-support tool in ATC for several years. For more information about the SSD design and empirical results, see [15-19].

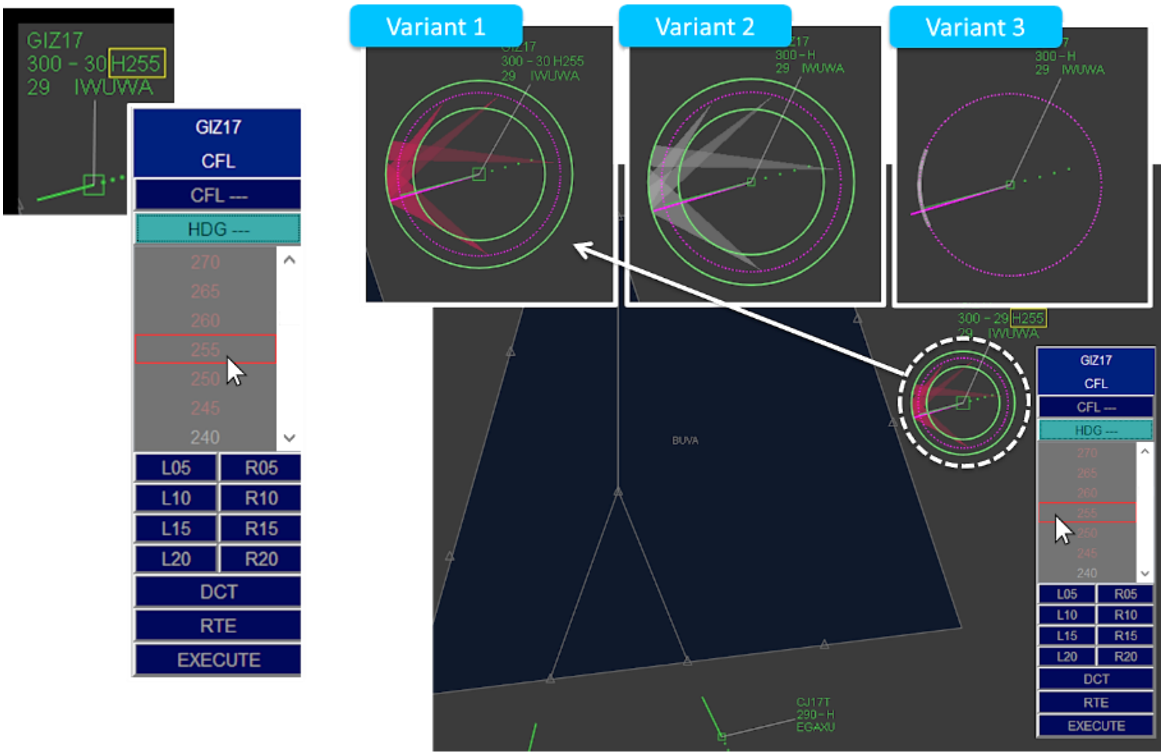


Fig. 4.6: Heading solution space – Solution Space Diagram (SSD). Variant 1 shades the triangular conflict zones with one colour, losing information about conflict urgency. Variant 2 shades the conflict zones according to time thresholds, restoring urgency information. Variant 3 only portrays conflict information in the heading domain, simplifying the SSD at the cost of enhanced information extraction (e.g., width and orientation of the conflict zones are lost, which can be used to judge aircraft proximity and flight directions).

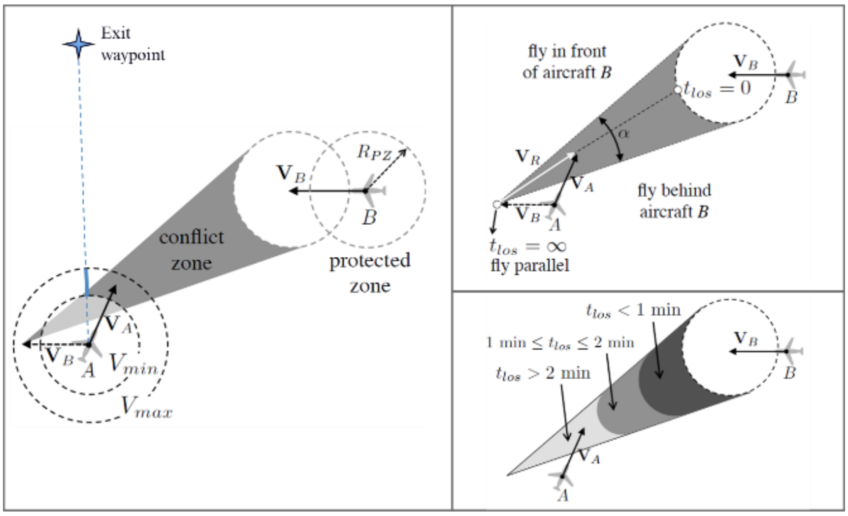
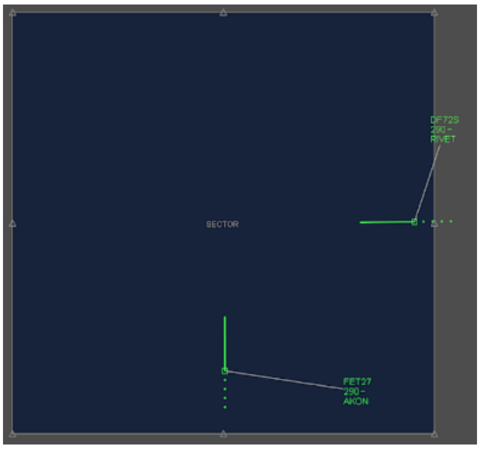


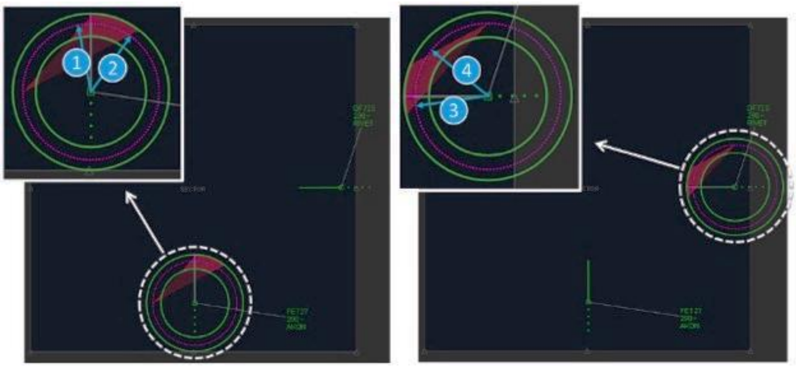
Fig. 4.7: SSD design and embedded domain information.

The SSD can also be used to formulate solutions that are efficient and also in line with common ATCo strategy heuristics mentioned in Section 3.4. Consider the traffic situation portrayed in Fig. 4.8(a). When selecting flight FET27, the SSD shows that the heading/speed solution space to the left of the current heading is richer than to the right. An efficient solution would be ①, which is both closer to the current heading and exit waypoint direction, thus requiring a small heading change. Solution ② requires a larger heading change and can therefore be marked as less efficient.

When selecting aircraft DF72S, other solutions to solve the conflict are indicated by ③ and ④. There, solution ③ is more efficient than solution ④ (requiring a smaller deviation). Now, considering the "point behind" heuristic described in Section 3.4, the SSD sort of explains why that strategy works well. In the traffic situation shown in Fig. 4.8(a), both aircraft are flying at the same speed and the portrayed situation shows that FET27 is closer to the flights' crossing point than is DF72S. Following the heuristic, an ATCo would typically leave FET27 alone and would solve the conflict by vectoring the other aircraft, DF72S, behind. Solution ③ is aligned with that heuristic and vectoring behind is indeed more efficient (i.e., small heading change) and more robust (i.e., most available solution space). Thus, the domain transparency offered by the SSD enables an ATCo to better understand why certain heuristics work well. Note that choosing solution ① would result in the same traffic pattern as solution ③: DF72S will pass behind FET27.



(a) traffic situation



(b) Potential solutions

Fig. 4.8: Horizontal conflict situation and the potential HDG solutions.

4.2.3 Route domain

The SSD provides information about whether aircraft pairs are in conflict (when the tip of the speed vector lies within a triangle) and all “go” and “no-go” heading and speed states that would result in new conflicts or solves them altogether, respectively. Thus, the SSD enables the ATCo to observe two states: the current state and the solution state(s). The SSD does not, however, allow the controller to anticipate the next possible state, such as when would be a good time to steer the aircraft back to its original trajectory toward the exit waypoint and how much time delay is introduced as a result of departing from the original route.

When integrating the SSD, and assuming the speed remains constant at the currently flown speed, a trajectory solution space can be calculated. As shown in Fig 4.9, the Trajectory Space Representation (TSR) reveals a family of ellipses (shaded in grey) where each point on the outline of a certain ellipse represents a constant flown distance toward the exit waypoint. Here, each point on the ellipse represents a potential location of an intermediate waypoint that will break up the trajectory is two segments. Using the current ground speed of the aircraft, the time delay at which the aircraft would arrive at the exit waypoint (relative to the original direct track) can be calculated. In Fig. 4.9 and Fig. 4.10, each shaded area corresponds to a certain time delay margin in seconds.

With the TSR, the ATCo is able to modify the route of the selected aircraft. Upon opening the RTE clearance menu, the TSR will become visible and the ACTo can place intermediate waypoint locations by moving and clicking the mouse cursor somewhere inside the editable area. As shown in Fig. 4.10, the editable area (i.e., candidate locations for intermediate waypoints) is limited to be at least 10 nm away from the current aircraft position and exit waypoint location and the angle θ between the two route segments should be larger than 120 degrees. These limitations were implemented based on feedback from controllers, who typically do not want aircraft to make sharper turns.

Note that the TSR is based upon an adaptation of previous designs from TUD research into 4D Trajectory-Based Operations [20-21]. In that research, the TSR encompasses waypoint locations in conjunction with speed targets to ensure the aircraft arrives at the exit waypoint at the original planned time, thus without any delay. The upper bound of the TSR is then limited by the maximum operating speed of the aircraft. In MAHALO, speed is assumed to remain constant, such that the TSR portrays areas resulting in arrival time delays.

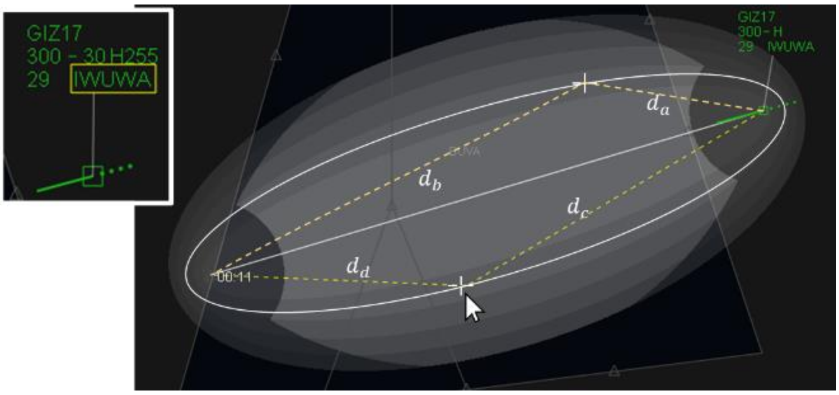


Fig. 4.9: Route solution space – Trajectory Space Representation (TSR). Each point on the white ellipse outline results in equal distances of two-segment trajectories. Thus here, $d_a + d_b = d_c + d_d$.

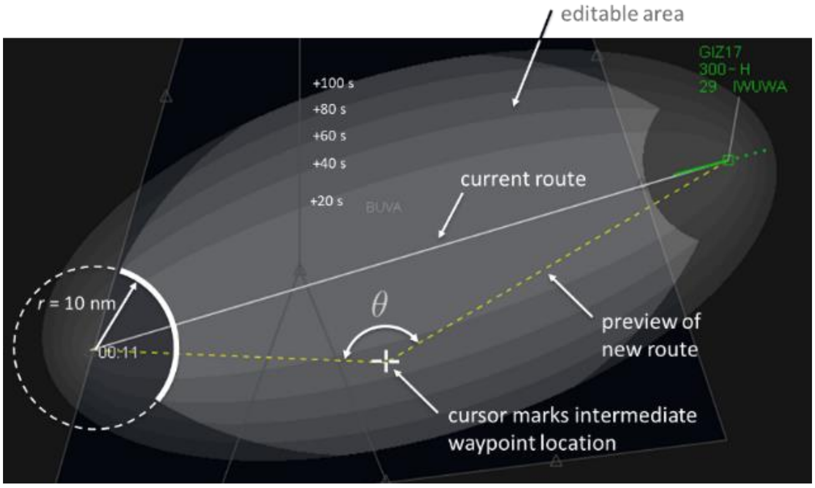


Fig. 4.10: TSR details.

Given that each "pixel" in the TSR represents a possible waypoint location that connects two route segments, each route segment can be probed for conflicts using STCA conflict detection logic. When at least one of the two segments results in a time and 5 nm distance overlap with another aircraft's flight plan, that 'pixel' in the TSR will be coloured red, marking an invalid waypoint location. When the distance overlap is between 5 nm and 7 nm, the pixel will be coloured amber, representing a cautionary 2 nm separation margin.

In Fig. 4.11, the same traffic situation as in Fig. 4.8 is depicted, but now showing the TSRs of both aircraft instead of the SSDs. It can be seen that the TSR tells the same story as the SSDs: making aircraft DS72S fly behind FET27 is more favourable as it features a richer solution space. The ATCo could also optimise a solution for efficiency by placing the intermediate waypoint close to the original route, resulting in less time delay.

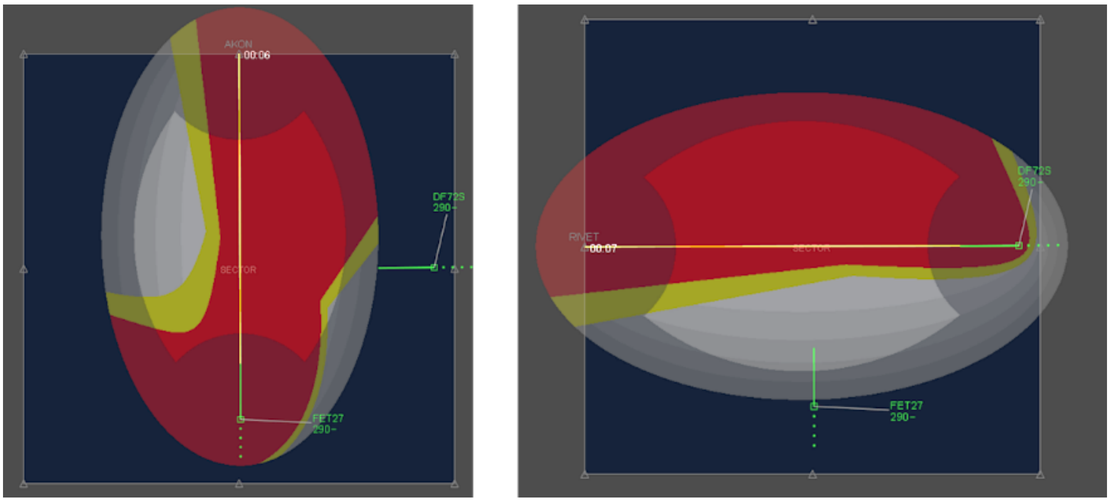


Fig. 4.11: TSR showing the red "no-go" space for waypoint locations that would result in a separation loss. Left: TSR for selected aircraft FET27. Right: TSR for selected aircraft DS72S.

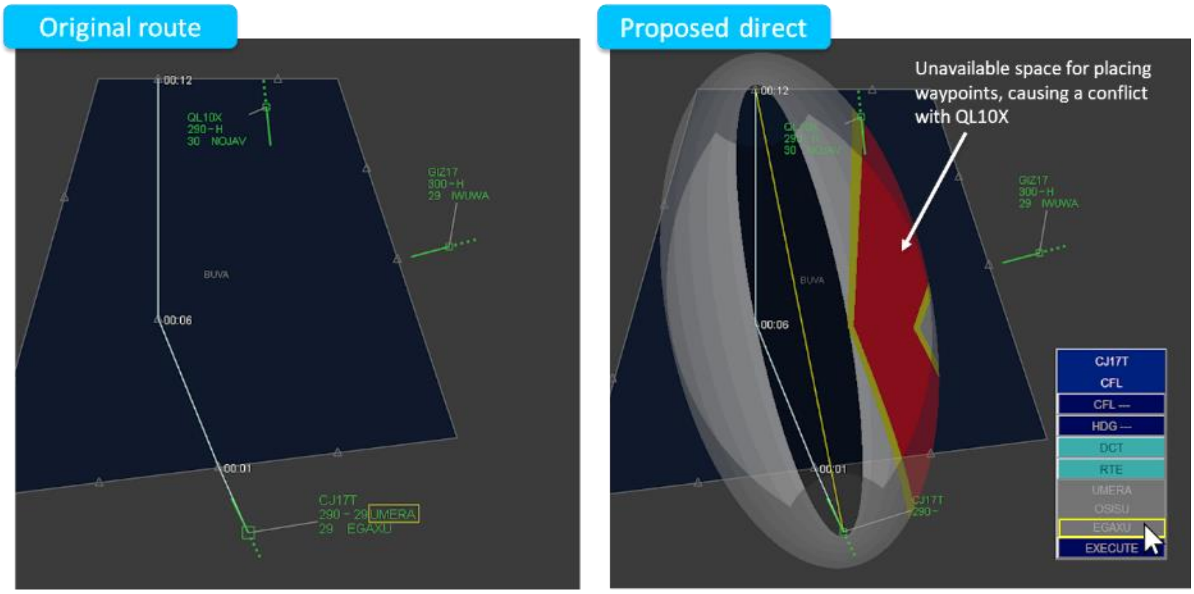


Fig. 4.12: Flying direct to the last route point makes the aircraft arrive earlier than originally planned, indicated by the “black hole” within the delay ellipses.

The TSR is not only useful for solving conflicts and make an aircraft steer back to the original track, it can also be used to fly more direct routes and potentially arrive earlier than planned. In Fig. 4.12 it is shown that waypoint locations that make the aircraft arrive earlier are indicated by “black holes” inside the elliptical TSR.

4.3 Agent transparency

Domain transparency provides a common ground for human-automation teamwork and is independent from any particular agent taking actions. That is, humans can use the visualised solution spaces to formulate any preferred action that is safe. Automation can use the solution spaces to calculate a solution that is both safe *and* efficient. Automation can either directly implement that solution or communicate the solution as an advisory that can be either accepted or rejected, as illustrated in Fig. 4.13.

When automation plots its solutions within the available solution spaces, the machine’s output can be evaluated by the ATCo in terms of its safety and efficiency [4]. As the solution spaces portray the input state-space for aircraft, the ATCo can observe what clearances the automation will issue to the aircraft. As such, this provides a level of transparency regarding the constraints that bound the behaviour of the automated agent [5].

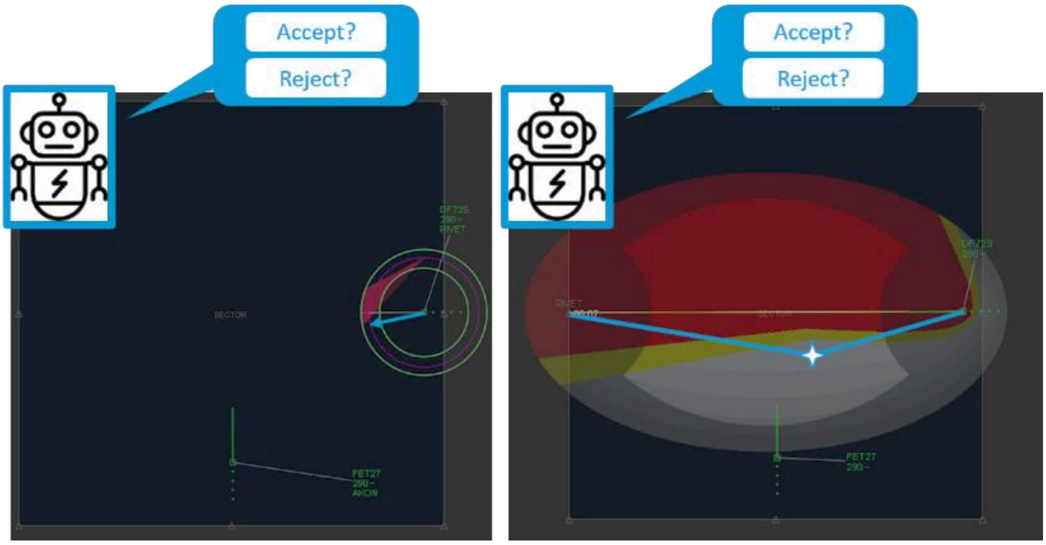


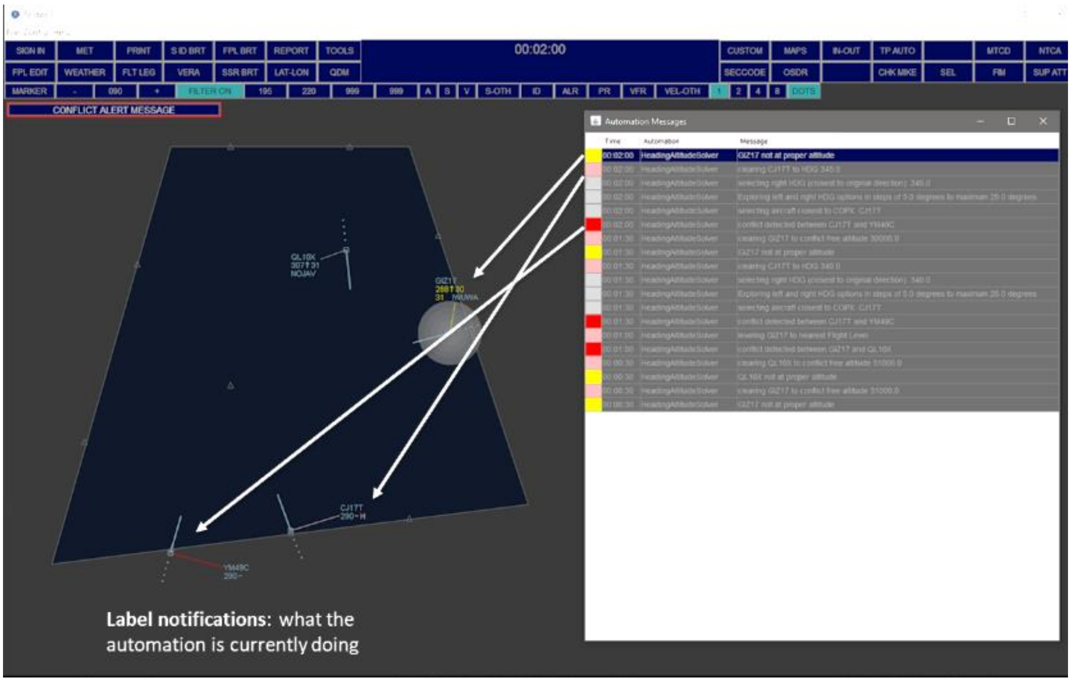
Fig. 4.13: Automation portraying its outputs within the visible solution spaces as advisories.

Although the machine's output can be plotted within the solution spaces, it does not necessarily explain much about the inner process of how the machine arrived at its decision. This requires the machine to expose more information about its *inner* decision-making process of how it evaluates the factors that would steer the solution in a particular direction.

Irrespective of the type of algorithms that guide the automation's behaviour (i.e., machine learning (ML) models developed in WP3 or classic rule-based algorithms), a first prototype of a generic visual and text-based language for conveying an algorithm's inner process have been developed. As shown in Figs. 4.14 and 4.15, colour-coded automation event notifications, displayed in flight labels, are connected to text-based messages displayed in a message table. In the development of this design, rule-based automation was implemented as a placeholder that will be replaced by the ML models developed in WP3.

The conceptual idea is that the automation is able to disclose information about its inner decision-making process by expressing that in the "language" that has meaning to the human ATCo. The automation events displayed in the message table are colour-coded and these colours match the highlighted elements in the flight labels. This allows the ATCo to visually observe on the PVD what automation is currently doing (and planning to do) with what aircraft. The message table can then be consulted to gain more insight into the reasoning of the automated agent.

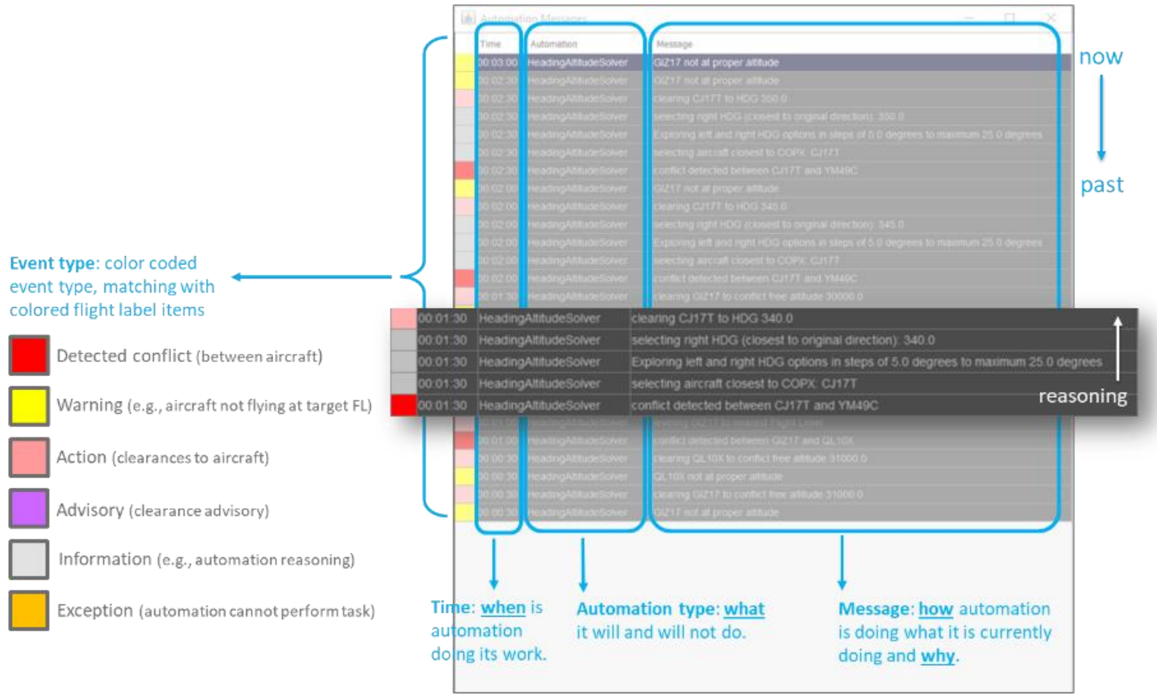
Together with domain transparency, a controller can monitor what the automation is doing by observing the notifications and the solution spaces for a deeper understanding of the conflict situation. As illustrated Fig. 4.16, the solution spaces can also help the controller to predict when automation would clear the aircraft to the exit waypoint. When a controller would disagree with the automated action or advisory, he or she can intervene by assuming manual control over a flight and implementing a clearance to his/her own preference using the solution space diagram as a decision-support tool.



Time	Automation	Message
00:02:00	HeadingAltitudeSolver	GIZ17 not at proper altitude
00:02:00	HeadingAltitudeSolver	clearing CJ17T to HDG 345.0
00:02:00	HeadingAltitudeSolver	selecting right HDG (closest to original direction): 345.0
00:02:00	HeadingAltitudeSolver	Exploring left and right HDG options in steps of 5.0 degrees to maximum 25.0 degrees
00:02:00	HeadingAltitudeSolver	selecting aircraft closest to COPX: CJ17T
00:02:00	HeadingAltitudeSolver	conflict detected between CJ17T and YM49C
00:01:30	HeadingAltitudeSolver	clearing GIZ17 to conflict free altitude 30000.0
00:01:30	HeadingAltitudeSolver	GIZ17 not at proper altitude
00:01:30	HeadingAltitudeSolver	clearing CJ17T to HDG 340.0
00:01:30	HeadingAltitudeSolver	selecting right HDG (closest to original direction): 340.0
00:01:30	HeadingAltitudeSolver	Exploring left and right HDG options in steps of 5.0 degrees to maximum 25.0 degrees
00:01:30	HeadingAltitudeSolver	selecting aircraft closest to COPX: CJ17T
00:01:30	HeadingAltitudeSolver	conflict detected between CJ17T and YM49C
00:01:00	HeadingAltitudeSolver	lowering GIZ17 to nearest Flight Level
00:01:00	HeadingAltitudeSolver	conflict detected between GIZ17 and QL10X
00:00:30	HeadingAltitudeSolver	clearing QL10X to conflict free altitude 31000.0
00:00:30	HeadingAltitudeSolver	QL10X not at proper altitude
00:00:30	HeadingAltitudeSolver	clearing GIZ17 to conflict free altitude 31000.0
00:00:30	HeadingAltitudeSolver	GIZ17 not at proper altitude

Label notifications: what the automation is currently doing

Fig. 4.14: Overview of agent transparency.



Time	Automation	Message
00:03:00	HeadingAltitudeSolver	GIZ17 not at proper altitude
00:02:30	HeadingAltitudeSolver	GIZ17 not at proper altitude
00:02:30	HeadingAltitudeSolver	clearing CJ17T to HDG 350.0
00:02:30	HeadingAltitudeSolver	selecting right HDG (closest to original direction): 350.0
00:02:30	HeadingAltitudeSolver	Exploring left and right HDG options in steps of 5.0 degrees to maximum 25.0 degrees
00:02:30	HeadingAltitudeSolver	selecting aircraft closest to COPX: CJ17T
00:02:30	HeadingAltitudeSolver	conflict detected between CJ17T and YM49C
00:02:00	HeadingAltitudeSolver	GIZ17 not at proper altitude
00:02:00	HeadingAltitudeSolver	clearing CJ17T to HDG 345.0
00:02:00	HeadingAltitudeSolver	selecting right HDG (closest to original direction): 345.0
00:02:00	HeadingAltitudeSolver	Exploring left and right HDG options in steps of 5.0 degrees to maximum 25.0 degrees
00:02:00	HeadingAltitudeSolver	selecting aircraft closest to COPX: CJ17T
00:02:00	HeadingAltitudeSolver	conflict detected between CJ17T and YM49C
00:01:30	HeadingAltitudeSolver	clearing GIZ17 to conflict free altitude 30000.0
00:01:30	HeadingAltitudeSolver	clearing CJ17T to HDG 340.0
00:01:30	HeadingAltitudeSolver	selecting right HDG (closest to original direction): 340.0
00:01:30	HeadingAltitudeSolver	Exploring left and right HDG options in steps of 5.0 degrees to maximum 25.0 degrees
00:01:30	HeadingAltitudeSolver	selecting aircraft closest to COPX: CJ17T
00:01:30	HeadingAltitudeSolver	conflict detected between CJ17T and YM49C
00:01:00	HeadingAltitudeSolver	lowering GIZ17 to nearest Flight Level
00:01:00	HeadingAltitudeSolver	conflict detected between GIZ17 and QL10X
00:00:30	HeadingAltitudeSolver	clearing QL10X to conflict free altitude 31000.0
00:00:30	HeadingAltitudeSolver	QL10X not at proper altitude
00:00:30	HeadingAltitudeSolver	clearing GIZ17 to conflict free altitude 31000.0
00:00:30	HeadingAltitudeSolver	GIZ17 not at proper altitude

Event type: color coded event type, matching with colored flight label items

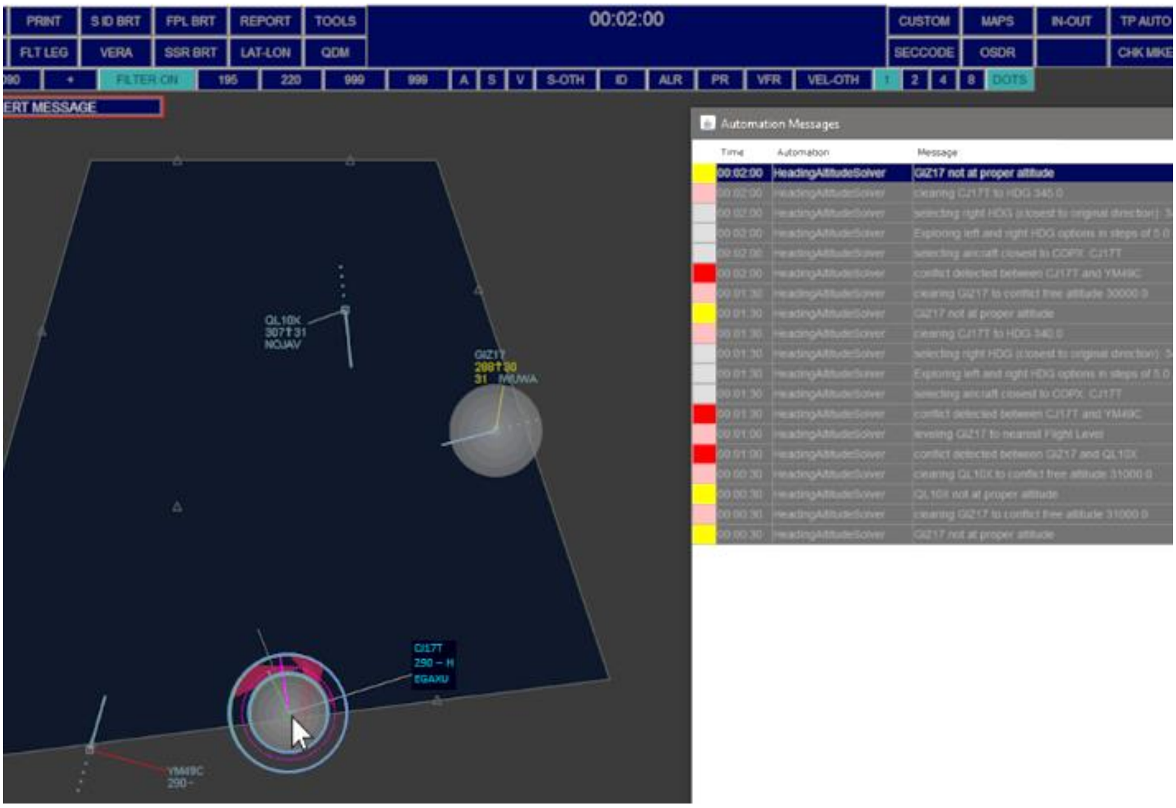
- 🟥 Detected conflict (between aircraft)
- 🟨 Warning (e.g., aircraft not flying at target FL)
- 🟥 Action (clearances to aircraft)
- 🟪 Advisory (clearance advisory)
- ⬜ Information (e.g., automation reasoning)
- 🟧 Exception (automation cannot perform task)

Time: when is automation doing its work.

Automation type: what it will and will not do.

Message: how automation is doing what it is currently doing and why.

Fig. 4.15: Automation message table provides a way to follow the reasoning of automation activities in a language that is expected to be meaningful and understandable to the ATCo.



Time	Automation	Message
00:02:00	HeadingAltitudeSolver	GIZ17 not at proper altitude
00:02:00	HeadingAltitudeSolver	clearing CJ17T to HDG 345.0
00:02:00	HeadingAltitudeSolver	selecting right HDG (closest to original direction):
00:02:00	HeadingAltitudeSolver	Exploring left and right HDG options in steps of 5.0
00:02:00	HeadingAltitudeSolver	selecting aircraft closest to COPX: CJ17T
00:02:00	HeadingAltitudeSolver	conflict detected between CJ17T and YM49C
00:01:30	HeadingAltitudeSolver	clearing GIZ17 to conflict free altitude 30000.0
00:01:30	HeadingAltitudeSolver	GIZ17 not at proper altitude
00:01:30	HeadingAltitudeSolver	clearing CJ17T to HDG 340.0
00:01:30	HeadingAltitudeSolver	selecting right HDG (closest to original direction):
00:01:30	HeadingAltitudeSolver	Exploring left and right HDG options in steps of 5.0
00:01:30	HeadingAltitudeSolver	selecting aircraft closest to COPX: CJ17T
00:01:30	HeadingAltitudeSolver	conflict detected between CJ17T and YM49C
00:01:00	HeadingAltitudeSolver	leveling GIZ17 to nearest Flight Level
00:01:00	HeadingAltitudeSolver	conflict detected between GIZ17 and QL10X
00:00:30	HeadingAltitudeSolver	clearing QL10X to conflict free altitude 31000.0
00:00:30	HeadingAltitudeSolver	QL10X not at proper altitude
00:00:30	HeadingAltitudeSolver	clearing GIZ17 to conflict free altitude 31000.0
00:00:30	HeadingAltitudeSolver	GIZ17 not at proper altitude

Fig. 4.16: Domain combined with agent transparency.

5. Positioning within MAHALO

In MAHALO, the E-UI will eventually be used to make the (proposed) actions made by ML models more explainable and understandable in order to foster acceptance. As shown in Fig. 5.1, the E-UI has relationships with parallel WP3 (ML model development) and WP5 (Integration) efforts. Eventually, the concept will be empirically tested in WP6 (Simulations) using human-in-the-loop simulations. In this chapter, the connections with WP3 and WP5 will be briefly explained.

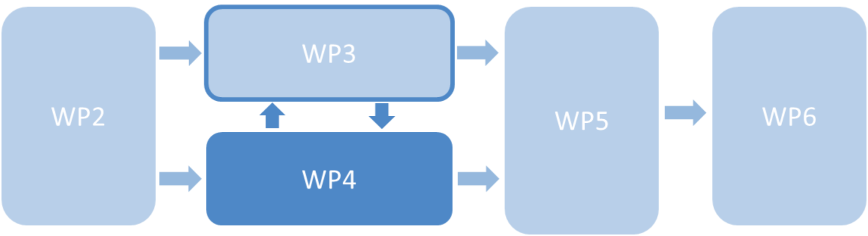


Fig. 5.1: Positioning of WP4 within the other technical work packages.

5.1 Machine Learning models (WP3)

The machine learning models developed in WP3 aim to cover both a Personalised (and Group) Prediction Model and an Optimised Prediction model. In an effort to further align domain and agent transparency, WP3 (and WP5) will experiment with using images of the solution spaces as inputs to the machine learning algorithms. The expected benefits are two-fold.

First, the advantage of using an image to represent the state of traffic conflicts is that its size (in pixels) is independent of the number of aircraft present in the airspace sector. That is, regardless of having 10 or 20 aircraft in the sector, the pixel size of the SSD for the selected aircraft always remains, for example, 128 x 128 pixels [22]. Capturing the traffic state in an image is therefore expected to benefit the speed of learning.

Second, when the ML models and the human ATCo use the same visual representations as inputs to formulate solutions, a true **shared mental model** will be achieved [22]. It is expected that this will further help to foster acceptance. When the ML models will then provide reasoning messages on why it decided for a range of solutions with various probabilities, and display that in the message table, it may help the ATCo to better understand the ML agent since they “speak” the same language. The goal is thus to create a digital assistant that behaves and reasons like a human colleague.

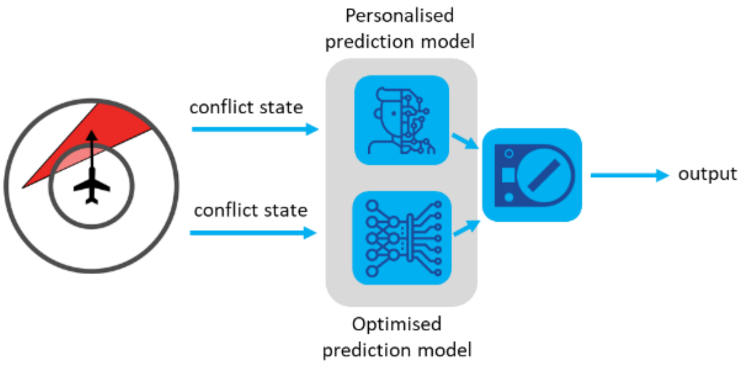


Fig. 5.2: Using solution space images as inputs to the ML models.

5.2 Integration (WP5)

Looking forward to WP5 (integration work package), connecting the E-UI and ML models will likely raise challenges that may require further interface enhancements and/or (minor) modifications. The developed E-UI described in this document therefore serves as a foundation to which more information could be added that may benefit transparency. The current E-UI aims to provide a nominal balance between 'too much' and 'too little' information by allowing the ATCo to effectively monitor the behaviour of the automation and manually intervene whenever required or desired.

5.3 Simulation (WP6)

In the real-time, human-in-the-loop simulation trials it is not yet decided at which stage and level of automation the ML will assist the human ATCo. There are, for example, two ways to implicitly evaluate the acceptance and trust in the automated system: either by accepting (and rejecting advisories) or by allocating flights to automation (or taking back control), as shown in Fig. 5.3. In MAHALO, both approaches will be supported. In the simulation, domain and agent transparency will be manipulated together with the conformance of advisories (comparing conformal vs optimal solutions).

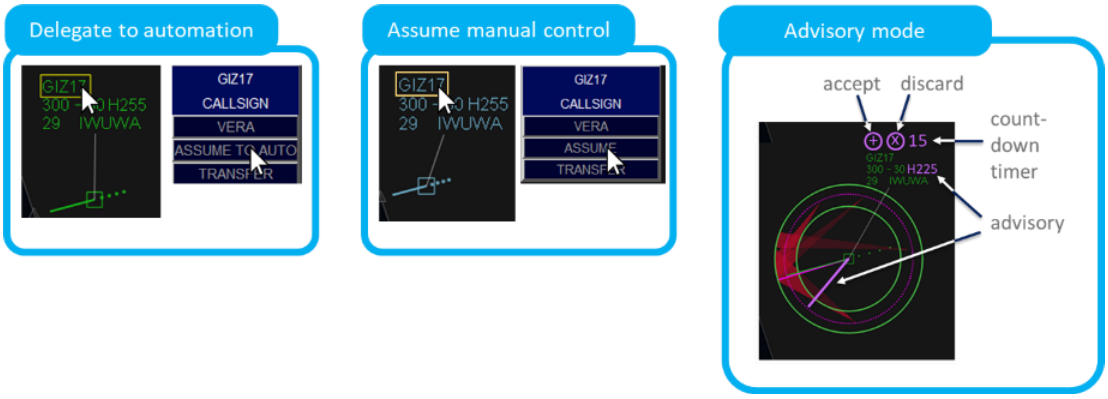


Fig. 5.3: Manual flights (green) can be allocated to automation and manual control can be re-assumed over flights currently fully controlled by automation (cyan). Automation can also act as an advisory system [23].

REFERENCES

[1] Westin, C., Borst, C., & Hilburn, B. (2016). Strategic Conformance: Overcoming Acceptance Issues of Decision Aiding Automation? IEEE Transactions on Human-Machine Systems, 46(1), 41–52. https://doi.org/10.1109/THMS.2015.2482480.

[2] Bhaskara, A., Skinner, M., & Loft, S. (2020). Agent Transparency: A Review of Current Theory and Evidence. IEEE Transactions on Human-Machine Systems, 50(3), 215–224. https://doi.org/10.1109/THMS.2020.2965529

[3] Christoffersen, K., & Woods, D. D. (2002). How to make automated systems team players. In E. Salas (Ed.), Advances in Human Performance and Cognitive Engineering Research (2nd ed., Vol. 2, pp. 1–12). Emerald Group Publishing Limited.

[4] Inagaki, T. (2008). Smart collaboration between humans and machines based on mutual understanding. Annual Reviews in Control, 32(2), 253–261. https://doi.org/10.1016/j.arcontrol.2008.07.003.

[5] Borst, C. (2016). Shared Mental Models in Human-Machine Systems. 13th IFAC Symposium on Analysis, Design, and Evaluation of Human-Machine Systems, 1–6.

[6] Woods, D. D., & Hollnagel, E. (2006). Joint cognitive systems: Patterns in cognitive systems engineering. Boca Raton: CRC/Taylor & Francis.

[7] Borst, C., Flach, J. M., & Ellerbroek, J. (2015). Beyond Ecological Interface Design: Lessons From Concerns and Misconceptions. IEEE Transactions on Human-Machine Systems, 45(2), 164–175. https://doi.org/10.1109/THMS.2014.2364984.

[8] Parasuraman, R., Sheridan, T. B., & Wickens, C. D. (2000). A model for types and levels of human interaction with automation. IEEE Transactions on Systems, Man and Cybernetics, Part A, 30(3), 286–297. http://www.ncbi.nlm.nih.gov/pubmed/11760769.

[9] Vicente, K. J. (1999). Cognitive Work Analysis: Towards Safe, Productive, and Healthy Computer-Based Work. L. Erlbaum Associates Inc., USA.

[10] Organisation, E., The, F. O. R., Of, S., Navigation, A. I. R., Air, E., & Management, T. (n.d.). First ATC Support Tools Implementation (FASTI) – Cognitive Task Analysis. October, October 2007.

[11] Kallus, K. W., Van Damme, D., & Dittmann, A. (1999). Integrated Task and Job Analysis of Air Traffic Controllers - Phase 2 : Task Analysis of En-route Controllers. Eurocontrol, 1–98.

[12] Fothergill, S., & Neal, A. (2013). Conflict-Resolution Heuristics for En Route Air Traffic Management. Proceedings of the Human Factors and Ergonomics Society 57th Annual Meeting, 57(1), 71–75. https://doi.org/10.1177/1541931213571018.

[13] SESAR [2019]. European ATM Master Plan. Digitising Europe's Aviation Infrastructure. Executive View. 2020 Edition. SESAR Joint Undertaking, Luxembourg: Publications Office of the EU, Dec. 17.

[14] EUROCONTROL (2017). EUROCONTROL Guidelines for short term conflict alert (STCA) – parts I, II and III. January 2017, Edition 1.0.

[15] Mercado Velasco, G. A. ., Borst, C., van Paassen, M. M., & Mulder, M. (2021). Solution Space Decision Support for Reducing Controller Workload in Route Merging Task. Journal of Aircraft, 58(1), 125–137. https://doi.org/10.2514/1.C035852

[16] Borst, C., Visser, R. M., Paassen, M. M. Van, & Mulder, M. (2019). Exploring Short-Term Training Effects of Ecological Interfaces : A Case Study in Air Traffic Control. IEEE Transactions on Human-Machine Systems, PP(PP), 1–10. https://doi.org/10.1109/THMS.2019.2919742

[17] Van Paassen, M. M., Borst, C., Ellerbroek., J., Mulder, M., & Flach, J. M. (2018). Ecological Interface Design for Vehicle Locomotion Control. IEEE Transactions on Human-Machine Systems, 48(5), 541–555. https://doi.org/10.1109/THMS.2018.2860601

[18] Borst, C., Bijsterbosch, V. A., van Paassen, M. M., & Mulder, M. (2017). Ecological interface design: supporting fault diagnosis of automated advice in a supervisory air traffic control task. Cognition, Technology & Work, 19(4), 545–560. https://doi.org/10.1007/s10111-017-0438-y

[19] Mercado Velasco, G. A., Borst, C., Ellerbroek, J., van Paassen, M. M., & Mulder, M. (2015). The Use of Intent Information in Conflict Detection and Resolution Models Based on Dynamic Velocity Obstacles. IEEE Transactions on Intelligent Transportation Systems, 16(4), 2297–2302. https://doi.org/10.1109/TITS.2014.2376031.

[20] Klomp, R., Borst, C., van Paassen, R., & Mulder, M. (2015). Expertise Level, Control Strategies, and Robustness in Future Air Traffic Control Decision Aiding. IEEE Transactions on Human-Machine Systems, 46(2), 255–266. https://doi.org/10.1109/THMS.2015.2417535

[21] Klomp, R. E., Riegman, R., Borst, C., Mulder, M., & Paassen, M. M. Van. (2019). Solution Space Concept: Human-Machine Interface for 4D Trajectory Management. Thirteenth USA/Europe Air Traffic Management Research and Development Seminar, 1–9.

[22] Van Rooijen, S. J., Ellerbroek, J., Borst, C., & van Kampen, E. (2020). Toward Individual-Sensitive Automation for Air Traffic Control Using Convolutional Neural Networks. Journal of Air Transportation, 28(3), 105–113. https://doi.org/10.2514/1.D0180

[23] Hilburn, B., Westin, C., & Borst, C. (2014). Will Controllers Accept a Machine That Thinks Like They Think? The Role of Strategic Conformance in Decision Aiding Automation. Air Traffic Control Quarterly, 22(2), 115–136.

[24] Demonstrator of the E-UI: https://drive.google.com/drive/folders/17ZF8hifStxvbd49CTW8Px-GMlQoEGnLw?usp=sharing