Function Allocation between Automation and Human Pilot for Airborne Separation Assurance

Husni Idris* Gabriele Enea**
Timothy A. Lewis***

* TASC an Engility Corporation Company, Billerica, MA 02182 USA
(Tel: 617.780.0987; e-mail: husni.idris@engilitycorp.com)
** TASC an Engility Corporation Company, Billerica, MA 02182 USA
(e-mail:gabriele.enea@engilitycorp.com)
*** NASA Langley Research Center, Hampton, VA 23681 USA
(e-mail: timothy.a.lewis@nasa.gov)

Abstract: Maintaining safe separation between aircraft is a key determinant of the airspace capacity to handle air transportation. With the advent of satellite-based surveillance, aircraft equipped with the needed technologies are now capable of maintaining awareness of their location in the airspace and sharing it with their surrounding traffic. As a result, concepts and cockpit automation are emerging to enable delegating the responsibility of maintaining safe separation from traffic to the pilot; thus increasing the airspace capacity by alleviating the limitation of the current non-scalable centralized ground-based system. In this paper, an analysis of allocating separation assurance functions to the human pilot and cockpit automation is presented to support the design of these concepts and technologies. A task analysis was conducted with the help of Petri nets to identify the main separation assurance functions and their interactions. Each function was characterized by three behavior levels that may be needed to perform the task: skill, rule and knowledge based levels. Then recommendations are made for allocating each function to an automation scale based on their behavior level characterization and with the help of Subject matter experts.

Keywords: Function Allocation, Automation, Air Traffic Management, Separation Assurance, Petri Nets.

1. INTRODUCTION

A principal function of air traffic management is separation assurance, which is responsible for maintaining minimum separation distances between aircraft and from hazardous or restricted airspace. This function is performed predominantly by air traffic controllers based in air traffic control facilities using radar surveillance of aircraft location and voice communication with pilots. Each controller is assigned a volume of airspace with a maximum number of aircraft to control simultaneously thus imposing capacity limits based on their workload. Therefore, currently this function is centralized with ground-based controllers.

With the advent of technologies such as the satellite-based automatic dependent surveillance and broadcast (ADS-B), aircraft can maintain awareness of their own position and share it with their surrounding traffic. Hence, concepts of distributed, airborne-based separation assurance have emerged, where aircraft equipped with ADS-B are delegated the responsibility of maintaining separation with their surrounding traffic, partially or completely. Distributed separation assurance promises to increase airspace capacity by mitigating the centralized workload limitation of the air traffic controller. However, due to pilot workload limitation, it is believed that automation in the aircraft cockpit is needed to enable the new separation responsibilities. NASA has developed a prototype of such automation, called the autonomous operations planner (AOP) and a concept for autonomous flight rules (AFR) (Wing 2011). NASA has also conducted several human-in-the-loop experiments to assess the feasibility of the concept using the AOP prototype (Wing 2010). AOP detects potential violations of the separation requirements between aircraft, called conflicts, based on shared ADS-B surveillance and intent information. AOP advises the pilot of trajectory change maneuvers that resolve these conflicts.

A key design question for airborne-based separation assurance is the allocation of functions between the human pilot and the automation. This question has been addressed implicitly relying primarily on elicitation of subject matter experts, engineering judgment, and human in the loop experiments, which are typically conducted in limited contexts in order to enable high fidelity prototype design and development. In this paper, a more thorough function allocation analysis for airborne-based separation assurance is presented, using AOP as a guiding example, but addressing functions that may not have been considered in the AOP design. A similar analysis was conducted for ground-based separation assurance (Landry 2011), which recommended additional functions such as traffic intensity avoidance.

Landry developed a top-down task analysis approach to identify key separation assurance tasks and then recommended function allocations using the automation levels developed by Sheridan (Sheridan 1992).

The approach of this function allocation assessment consisted of: (1) A task analysis to identify the main functions of separation assurance. (2) Formal modeling with the help of Petri Nets in order to highlight the interactions between the tasks. (3) Characterizing key tasks by the behavior level needed to perform them according to Rasmussen's skill-based, rule-based and knowledge-based levels (Rasmussen 1983) and correspondingly allocating them to an automation scale based on Sheridan's automation levels (Sheridan 1992).

The analysis approach is detailed in the next section including these three components. This is followed by two examples demonstrating the application of the analysis to two main separation assurance functions, conflict detection and conflict resolution. Finally, an overall function allocation analysis of a larger set of key separation assurance functions is presented based on elicitation of a small group of subject matter experts.

2. Analysis Approach

The approach of this function allocation assessment consisted of the following elements:

2.1 Task analysis

The separation assurance tasks were identified in an abstract framework independently from who may perform them to enable identifying possible function allocation schemes. The separation assurance tasks were initially divided into four high level tasks:

(1) Conflict Identification (CI): Identify potential loss of separation (LOS).

(2) Conflict Assessment (CA): Determine the need to resolve a conflict based on its severity.

(3) Resolution Selection (RS): Select a resolution maneuver for the conflict.

(4) Resolution Implementation (RI): Implement the resolution through communication and maneuvering.

Then, these tasks were divided into subtasks gradually whenever a function was too complex to be allocated to the human or to the automation. Scenarios were used to provide context where AOP and the AFR concept were used as an example automation instantiation. However, additional tasks that AOP did not consider were identified. Two scenarios, one for conflict detection and one for conflict resolution, are presented as examples in the next two sections.

2.2 Petri Net Modeling

Petri nets were used to provide a formal representation of the functions, and the information flows and interactions between them. Petri nets (Fig. 1) consist of places (circles) that represent conditions, transitions (rectangles) representing tasks, and arrows that lead from input places to transitions and from transitions to output places. Tokens (small circles that may have multiple colors as identities) are placed inside places when the corresponding conditions are true. Transitions fire (i.e., tasks are performed) once tokens are present in their input places, which results in removing tokens from the input places and adding tokens to the output places. When a transition fires the net moves to a new state (i.e., configuration of tokens in places). Using Petri nets it is possible to identify issues associated with allocating the separation assurance tasks among agents, their information sharing, and the timeliness and synchronization of their actions. In this paper, the human or automation agents are represented as tokens: If a task is allocated to the human, the automation, or both, then a human token, an automation token, or both, respectively, are needed in input places for it to fire. This representation enables modeling dynamic allocations, where if a task is allocated to either the human or the automation, it may be executed by one or the other dynamically based on which resource is available at the time.

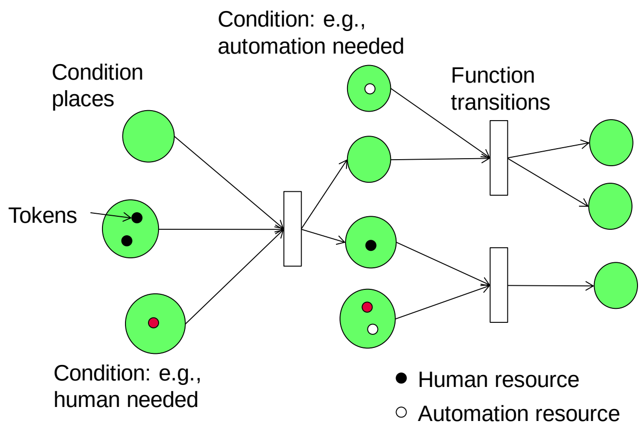

Fig. 1. Petri nets basic components.

2.3 Behavior and Automation Level Analysis

The criteria used to guide function allocation between the human and the automation started from Fitts' 1951 list of men are better at – machines are better at (MABA-MABA). More recently, Sheridan proposed in his supervisory control theory a systematic approach where each function is characterized along two dimensions: physiological locus (consisting of sensory, cognitive or response activities) and behavior level, based on Rasmussen's knowledge-based, rule-based and skill-based model (Sheridan 1992). Sheridan suggested that skill-based functions be allocated to task-interactive automation, rule-based functions be allocated to a human-interactive computer, while Knowledge-based functions requiring experience are allocated to the human supervisor. Recently (Cummings 2014) suggested the addition of an expert level of behavior and related these levels to the uncertainty involved in a task (Fig. 2). She also suggested allocating functions to the human and the automation according to these behavior levels: A skill-based function involves reliable state and sensor information and is most suitable for automation. A rule-based function is a good candidate for automation is the rule set if well established and tested. A knowledge-based function can be partially automated to assist with data manipulation to support human decisions. Finally, human reasoning is needed for expert-based function with possible help from the automation.

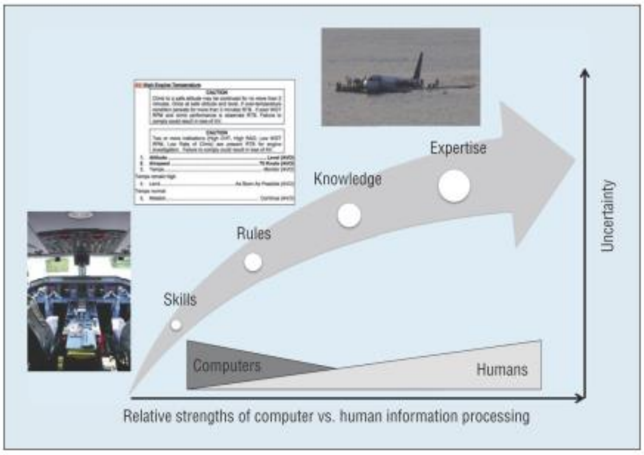

Table 3. Degree of automation as a function of a desired behavior.

Cognitive behavior/task	Degree of automation
Skill-based	Best candidate for automation, assuming reliable sensors for state and error feedback
Rule-based	Possible candidate for automation, if rule set is well-established and tested
Knowledge-based	Some automation can be used to help organize, filter, and synthesize data
Expertise	Human reasoning is superior, but can be aided by automation as a teammate

Fig. 2. Human behavior and automation (Cummings 2014).

Because of the low concept maturity, the tasks in this analysis are not defined in sufficient detail to be characterized based on the physiological locus (for example, making decisions based on aural versus visual capabilities of the human versus the automation in a sensory activity). Therefore, the tasks were characterized by only three behavior levels: skill, rule, and knowledge. Then the key tasks identified were allocated to the seven levels of automation (LOA) shown in Table 1, which is a subset of the automation levels identified by Sheridan (Sheridan 1992).

Table 1. Levels of Automation (LOA)

LOA	Automation Level Description
1	No automation assistance, human take all decisions
2	The automation presents few alternatives, the human decides which one to select
3	The automation presents one alternative, the human decides to select it or not
4	The automation allows the human a restricted time to veto, then it executes
5	The automation executes automatically, then necessarily informs the human
6	The automation executes automatically, informs the human only if it (the automation) decides to
7	The automation decides everything and ignores the human

3. Conflict Detection Scenario

A typical conflict detection (CD) scenario is shown in Fig. 3, involving an ownship aircraft (which is conducting the CD task) and an intruder aircraft representing surrounding traffic. The scenario highlights the multiple trajectories that the aircraft may follow and hence the complexity of the decision as to which trajectories should be probed for conflict. Each aircraft broadcasts its state vector and its intent specified as trajectory change reports (TCR) and a target state which identifies the end state if the aircraft is currently turning or changing altitude. The intent and state projection may coincide as for the ownship or may be different as for the intruder. This scenario shows a potential conflict between the intruder state projection and the ownship intent/state projection.

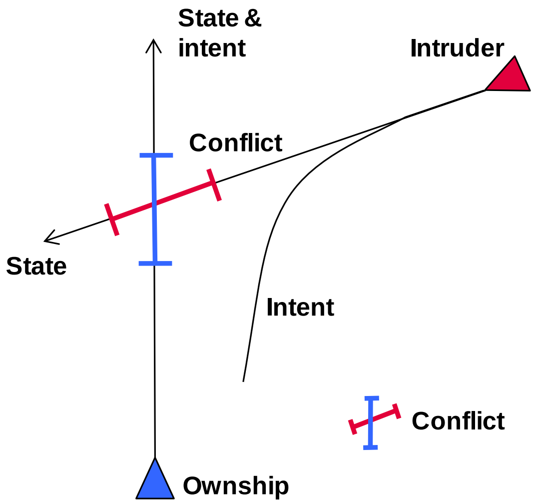

Fig. 3. Conflict detection scenario.

Fig. 4 shows a Petri net model of the AOP instantiation of this scenario. Each task (transition) is colored red if it is skill-based, yellow if rule-based, and blue if knowledge-based. Multiple color shading is used if the task is believed to have multiple behavior levels. The ownship has three alternative trajectories: the planned trajectory in the Flight Management System (FMS), the commanded trajectory based on the guidance settings currently engaged, and a trajectory projection of the current state. The transition "select primary/secondary trajectory ownship" selects from these trajectories one for a "primary CD" task and one for a "secondary CD" task, according to a set of rules: it uses the commanded trajectory for primary CD except when the aircraft is currently coupled (the guidance settings match the trajectory in the FMS) with a predicted decoupling (For example predicted command to start a descent). In this case it uses the planned trajectory for primary CD and the commanded for secondary CD. Therefore, the transition "select ownship trajectory for secondary CD" needs tokens in the predicted decoupling and coupled status to fire. The "Select Primary/Secondary Trajectory Traffic" selects one trajectory based on the following rule: use the traffic intent based on the TCR, if TCR is not available based on the target state, if not available based on a state projection. The primary and secondary CD tasks are rule-based: If uncertainty bounds around the trajectories overlap then a conflict is detected (Karr 2006). Primary and secondary conflicts can be present at the same time which is represented by separate "Display Primary" and "Display Secondary" transitions that are skill-based and easily automated.

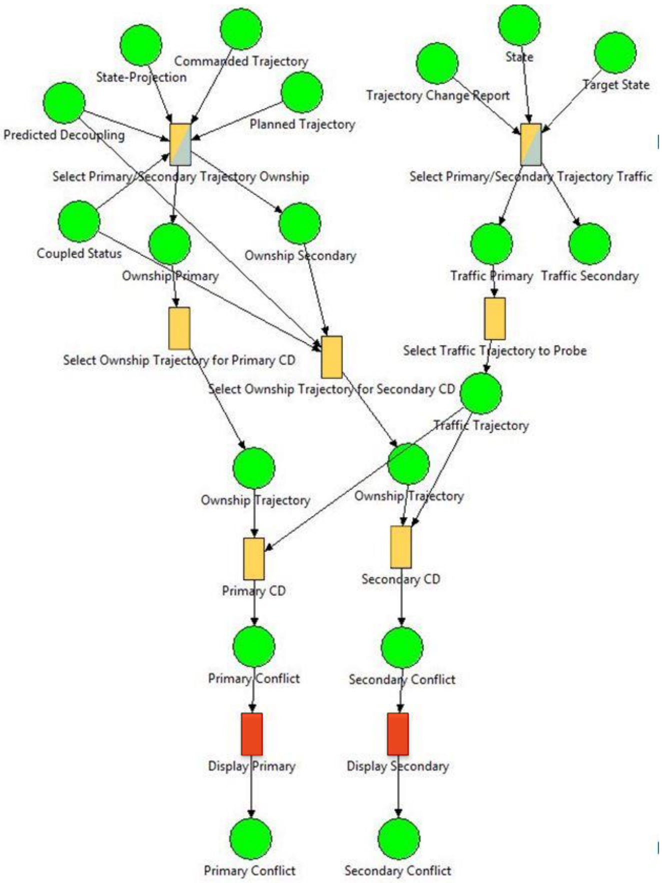

Fig. 4. Petri net model of conflict detection scenario.

The scenario in Figure 3 shows that there could be a benefit in including the pilot in this decision, hence the double color assigned to the tasks of trajectory selection in Fig. 4. In this scenario, according to its current implementation, AOP would probe the Ownship intent against the intruder (traffic) intent only and not detect a conflict. If the intruder did not follow its intent and traveled along the state projection, a conflict would happen, potentially with a short warning time for the pilot to react. This is a typical blunder situation that can be mitigated by the automation also probing the projection of the intruder current state as a secondary trajectory. The pilot may, if given the opportunity, decide based on experience to either trust the traffic to follow its intent or to avoid the potential blunder. The uncertainty of the situation motivates including the pilot in the decision making, however, at the expense of increased pilot workload. This scenario also highlights that airborne conflict detection is potentially more complex than ground-based conflict detection: In the airborne situation the ownship does not control the intruder and pilot-pilot coordination is not common. On the other hand, in the ground-based situation, the controller controls both aircraft and controller-pilot coordination is common practice.

4. Conflict Resolution Scenario

A resolution maneuver has to be selected for a conflict that is assessed to need resolution, for example if within a certain time horizon. AOP selects from two conflict resolution (CR) algorithms: strategic and tactical intent-based CR denominated SICR and TICR. As notionally presented in Fig. 5, SICR provides resolution maneuvers that follow particular patterns such as route offsets, return to the flight plan, and meet constraints (such as a required time of arrival (RTA)). SICR resolutions are complete route changes that can be implemented in the FMS. On the other hand, TICR uses tactical (heading or altitude) deviations that disregard constraints and are faster to compute. However, TICR tactical resolutions are open ended and hence require a recovery trajectory to return to the FMS route.

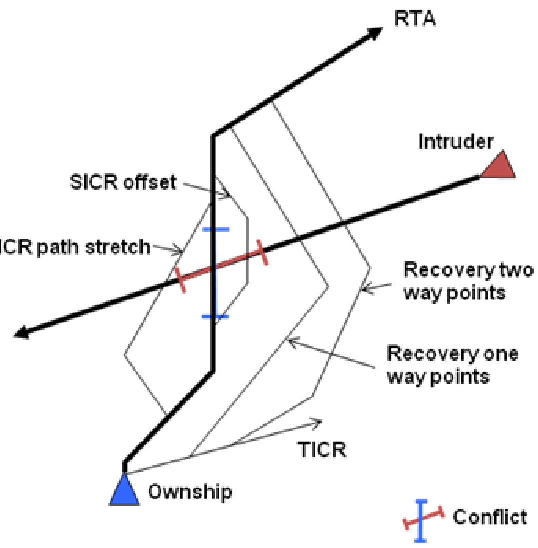

Fig. 5. Conflict resolution scenario.

The Petri net model in Fig. 6 represents the selection decision between the two resolution algorithms in AOP. The existence of a primary conflict token in the Primary Conflict to Solve place fires the task "check time to first loss of separation (LOS)", which is skill based automated task that produces a token to either the above SICR threshold (set currently at five minutes) or to the below SICR threshold places. The "use SICR" task is rule-based: If the aircraft is in coupled status and the time to first LOS is above the threshold then SICR is enabled. Similarly, the "use TICR" task is rule-based and represented by three transitions that are enabled if: either the time to the first LOS is below the threshold from the onset, or SICR did not find a solution before the time to first LOS slips below the threshold, or the aircraft is in decoupled status. At any point, the pilot can use a "manual override" knowledge-based task to disable SICR and enable TICR, representing the pilot overriding the automation by switching from SICR to TICR earlier than the automated switch threshold. For example, based on circumstances that are difficult to foresee by the automation, the pilot may decide to speed up the resolution since TICR has fewer constraints and therefore a bigger solution space, however, at the expense of a recovery after the resolution. Both "SICR" and "TICR" are rule-based tasks that are performed according to predesigned algorithms.

They both need criteria that are created according to either rule-based defaults (The most efficient resolutions in terms of fuel/time are ranked the highest) or knowledge-based human generated criteria. Finally, the pilot "selects resolution" from a list of proposed resolutions using additional knowledge and intuition.

In addition, AOP provides the pilot with the ability to "create manual resolution" which is a knowledge-based decision taken by the pilot based on the knowledge of the operational situation. For example, the current AFR rules dictate that if the automation of one aircraft in a conflict did not resolve it by a certain time then both aircraft automations attempt to resolve it. If not coordinated, this situation may result in cyclical instability leading to no convergence to a solution. The pilot is maintained as a safeguard to override the automation and resolve the conflict manually, preferably with coordination with the other pilot.

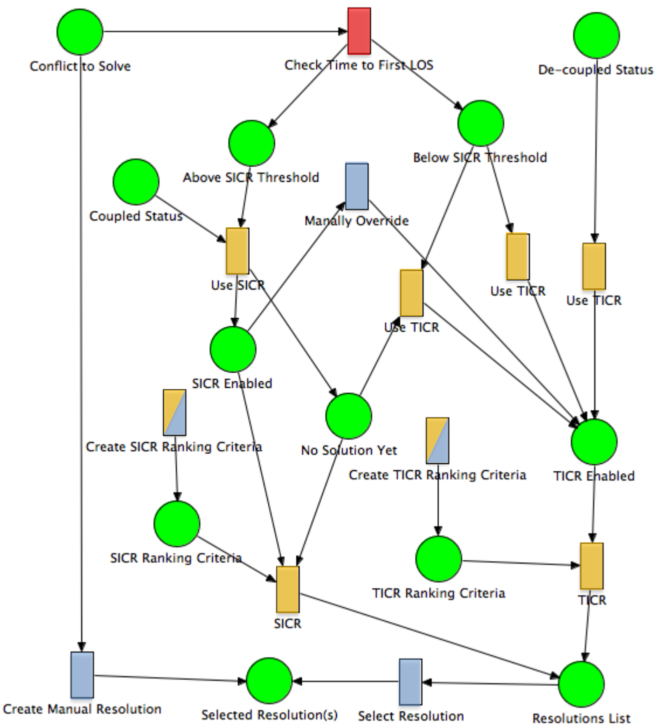

Fig. 6. Petri net model of *Conflict resolution scenario*.

5. Overall Function Allocation Analysis

Three SMEs who are pilots and familiar with the AFR concept were presented with a list of abstracted tasks and asked to assign to each a behavior level among skill-, rule- and knowledge-based and a level of automation (LOA) that includes a sub-set of the Sheridan scale (Table 1). Table 2 shows the ranges of the behavior level characterization and of the automation level allocation in the answers for each task. The following observations are made under each of the major task areas:

Table 2. Task Behavior level (BL) (S=Skill, R=Rule, K=Knowledge) and Automation Level (LOA)

	Task	BL	LOA
CI	Select Own Trajectory for CD	R	7
	Select Traffic Trajectory for CD	R	7
	Select CD Time Horizon	R (K)	4 – 5
	Generate CD Separation Criteria	R (K)	4 – 7
CA	Determine Time Urgency of Potential LOS	R (K)	3 – 5
	Determine Cause of Potential LOS	R – K	4 – 5
	Determine Certainty of Potential LOS	R (K)	4 – 6
	Determine Complexity of Potential LOS	R – K	4 – 7
RS	Create Ranking Criteria for CR	R – K	1 – 2
	Select Manual or Automated CR	K	2
	Select Tactical versus Strategic CR	R – K	2 – 4
	Select the Resolution Maneuvers	R – K	2 – 4
	Select Time Horizon for CR	R – K	5 – 7
	Select Constraints to Relax for CR	R – K	3 – 5
RI	Implement Selected Resolution	S – R – K	1 – 4

5.1 Conflict Identification

In addition to the selection of the ownship and traffic trajectories, two additional subtasks were identified for conflict detection: selecting the time horizon and selecting the separation criteria. There was a strong resistance by the SME's to allocating a role to the pilot in conflict detection, particularly for the trajectory selection tasks. The argument was that it would distract the pilot from the main responsibility of safely flying the airplane. Hence the responsibility to detect conflicts should still be assigned to the ground-based controller or to the airborne automation but not to the human pilot. This result implies that the rules used by the automation should be sufficient without human input to ensure the safe operation of the flight. For example, the blunder situation presented in Section 3 should be avoided by expanding the automation rules to cover most possible trajectories. Limited roles for the pilot were considered in selecting a time horizon and the separation criteria for the detection. However, the functions were still characterized mostly as rule-based and recommended to be automated. The role of the pilot is to be informed by the automation or to be able to veto the automation to adjust the rules when desired. For example, the pilot may be risk averse and hence decide to add buffers to the automated separation minima or to increase the horizon of the detection to avoid, for example, frequent rolling detections with the same flight.

5.2 Conflict Assessment

Four criteria were hypothesized as potential factors that are needed to assess if a conflict should be resolved once identified or delayed until more information about it are available: (1) the time urgency of the conflict, for example, in terms of the time until LOS, (2) the cause of the conflict, for example, some conflicts may be caused by intent that is known to be wrongly transmitted, (3) the uncertainty of a conflict, for example, a conflict may be identified along a route that is unlikely to be followed by the traffic, and (4) the complexity of the conflict, for example, in terms of the number of flights involved. All the functions of determining these factors were believed to be mainly rule-based but to have knowledge-based elements. For example, the SME's felt that most of the possible causes of the conflict should be known and hence based on a set of rules. However, one cannot rule out causes outside such rules that may arise and require knowledge-based behavior to assess. Similarly, the urgency and complexity of a conflict may depend on how busy the pilot is at the time.

Despite the existence of the knowledge-based behavior, the SME inputs suggest that these functions should be mostly automated due to the high pilot workload needed for computation, interpretation and inference. For example, the human is known to perform poorly in assessing and interpreting probabilities. The role assigned to the human was higher than for conflict identification, but was limited to reacting to the automation assessment, through alternative selection or veto.

5.3 Resolution Selection

Six subtasks were identified for resolution selection: (1) creating ranking criteria for the selection, such as based on fuel efficiency or maneuver complexity, (2) selecting manual or automated resolution where the pilot may decide to resolve a conflict manually as explained in Fig. 6, (3) selecting strategic or tactical resolutions as explained in Fig. 5, (4) selecting a time horizon over which the trajectory should be free of conflict, for example ten or twenty minutes, (5) selecting the specific resolution trajectory/maneuvers such as using altitude or path stretching, and (6) selecting constraints to relax such as allowing the violation of time schedule constraints.

Most functions were characterized to have rule-based and knowledge-based elements, except the selection of the manual or automated resolution which was characterized as knowledge-based and assigned to the pilot with limited automation role to present alternatives. The SME's commented that one benefit of this manual task is to build the knowledge of the pilot in this new responsibility through involvement.

Similarly, the pilot was given a significant role in the tasks of selecting ranking criteria, tactical or strategic resolution, the resolution maneuvers, and the constraints to relax. The pilot is recommended to maintain the ability to select from alternatives suggested by the automation, or at least to be informed and be able to veto the automation selection. For example, the ranking criteria may be based on operational goals that are impacted by the knowledge and expertise of the pilot. While automation rules can be designed to minimize the use of tactical resolutions in favor of strategic more efficient resolutions, the pilot may decide to override these rules and switch to tactical resolution to resolve a conflict faster as explained in the scenario in Fig. 6. Similarly, default rules may be automated for the prioritization of the maneuvers to select or the constraints to relax, which help resolve conflicts without the pilot when under time pressure and help increase the trust of the pilot in the automation. However, providing alternatives to the pilot to choose from allows the pilot to bring information based on experience and expertise possibly not easily known by the automation. The automated rules can still prevent the pilot from selecting constraints that should not be violated such as active special use airspace. Finally, the selection of the resolution horizon is recommended to be automated, with potentially considering different horizons dynamically and a limited human role.

5.4 Resolution Implementation

The LOA ranged between 1, human take all decisions with no automation assistance, and 4, representing that the automation allows the human a restricted time to veto, then it executes. Currently the pilot is in charge of performing this task. However, the SME answers indicated that "executing a maneuver should be second nature to pilots", which is the definition of skill-based behavior. It was mentioned that in the near future the automation could be in complete control of the execution of the trajectory/maneuver of the aircraft.

REFERENCES

Cummings, M. (2014). Man versus machine or man + machine?, *IEEE Intelligent Systems* Vol.29, No. 5.

Fitts, P.M. (1951). Human engineering for an effective air navigation and traffic control system. *Washington, DC: National Research Council.*

Landry, S. (2011). The next generation air transportation system: an approach to function allocation. *Human Factors and Ergonomics in Manufacturing and Service Industries.* Pages 1-11. Wiley Periodical Inc.

Karr, D. and Vivona, R. (2006). Conflict detection using variable four-dimensional uncertainty bounds to control missed alerts, *AIAA Guidance, Navigation, and Control Conference*, Keystone, CO.

Rasmussen, J. (1983). Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models, *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-13, No. 3.

Sheridan, T.B. (1992). *Telerobotics, Automation, and Human Supervisory Control*, Chapter 1, MIT Press, Cambridge, MA.

Wing, D.J., Cotton, W.B. (2011). For spacious skies: self-separation with "autonomous flight rules" in US domestic airspace, *11th AIAA Aviation, Technology, Integration, and Operations (ATIO) Conference*, Virginia Beach, VA.

Wing, D.J., Prevot, T., et al. (2010). Comparison of ground-based and airborne function allocation concepts for NextGen using human-in-the-loop simulations, *10th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference,* Forth Worth, TX.