[EN-A-051] Design of a Multi-Agent System framework for Decentralized Decision Making in Air Traffic Management

(EIWAC 2017)

+Th. Koca*, M. A. Piera*, M. Radanovic*

*Telecommunications and System Engineering Department
Autonomous University of Barcelona
Barcelona, Spain
[thimjo.koca | miquelangel.piera | marko.radanovic] @ uab.cat

Abstract: Present tactical intervention methods from the air traffic control system, preserving the standard separation minima between aircraft, inherently do not scale well resulting in sector saturations. In this work, an automation-based framework is claimed within the air traffic management sector moving from a centralized system to a distributed system. A potential set of aircraft with self-governed capabilities searches for satisfactory conflict-free resolution trajectories. The key issues that such a framework should consider and propose a "skeleton" on which such a system can be based are described.

Keywords: Multi-agent system, MAS framework, agent negotiation, satisficing, search strategy, deadlock

1. INTRODUCTION

Current research on air traffic management (ATM) is striving to improve the airspace capacity, accessibility and efficiency of operations in highly dense areas, while preserving the safety performance indicators [1]. Tactical interventions from the air traffic control (ATC) system preserving standard separation minima between aircraft inherently do not scale well, leading to the well-known capacity saturation. An increased number of detected conflicts can affect not only the air navigation procedures, but also the full safety net, since the present Traffic alert and Collision Avoidance System has been designed to operate in the low dense areas only [2], [3].

AGENT (Adaptive self-Governed aerial Ecosystem by Negotiated Traffic) is a European research project presenting a possible future design of the automated ATM system, supporting a shift from a centralized control to a distributed one with the goal of improving the airspace capacity and efficiency while maintaining safety [4]. A move towards distributed systems results as well in systems with better scalability. Furthermore, it implies engagement of the airspace users in the decision making, which results in a higher satisfaction rate. The exploration of efficient solutions of emerging conflicts considering the surrounding traffic lies at the system's core.

This work introduces a Multi-agent system framework which tackles the problem specifications. Its goal is to identify the main issues that need to be taken into consideration and provide a scheme on which specific multi-agent systems (MAS) can be designed.

The rest of this paper is structured as follows. Section 2 describes T-CAS and its problematic behavior which leads to the necessity of a MAS, section 3 addresses issues of significant importance that need to be clarified from each MAS applied to our problem, section 4 gives the proposed "skeleton" while concluding remarks are given in section 5.

2. T-CAS, ITS LOGIC AND PROBLEMATIC BEHAVIOR

The Traffic Alert and Collision Avoidance System (TCAS) is a family of airborne systems designed to reduce the risk of mid-air collisions between aircraft [5]. The chosen system is activated in cases when the minimum separation distance between two aircraft is violated.

Its logic is straightforward, among the involved aircraft one must climb and the other one descent to regain the minimum separation distance.

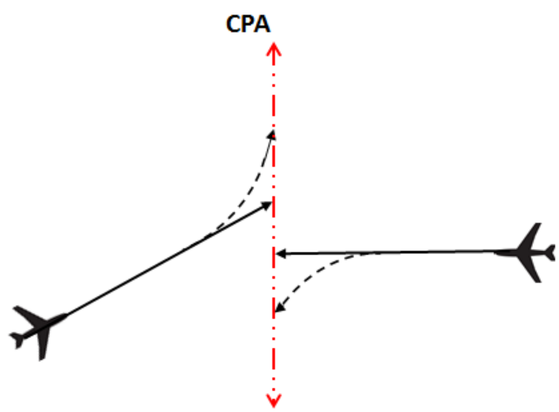

Figure 1 T-CAS logic illustration

This system is designed to deal with pairwise encounters and demonstrates a very good performance in densities up to 0.3 aircraft per square nautical mile [2]. In higher densities however, its limited logic can induce new conflicts with the surrounding traffic [3].

Moreover T-CAS is designed to operate only in vertical plane and therefore possible horizontal solutions are not considered.

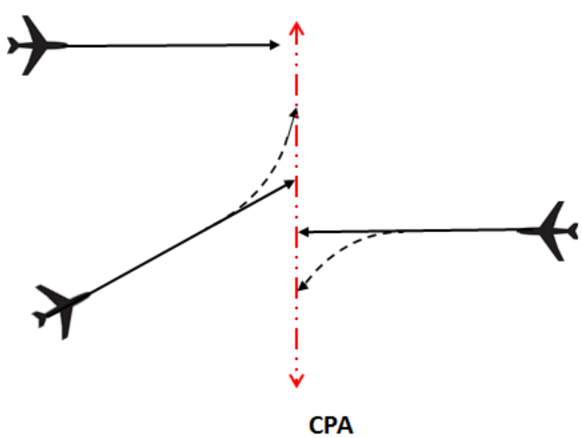

Figure 2 Illustration of a possible induced conflict

3. KEY ISSUES TO BE CONSIDERED

To deal with the two introduced issues a collaborative multiagent system (MAS) framework is proposed in this article. In the given approach, surrounding traffic will be always considered and horizontal possibilities can be as well explored.

In the design process of a MAS framework applied to our problem there are some key issues that need to be clarified.

Firstly, given two conflict aircraft, which other aircraft should be considered in the solution-seeking process. In other words, how can we quantitatively define "surrounding traffic".

Secondly, being in an environment where different entities measure "goodness" differently (each agent wants to get the best trajectory for itself and therefore a good solution for agent A is not necessarily a good one for agent B), the question of how to take satisfactory decisions for all agents naturally arises.

Thirdly, our problem can be formulated as a search problem. Searching strategies can be classified to exploitative and exploratory ones. After choosing in subsection 3.3 the way in which a good solution is defined, we can argue and a priory guarantee that exploitative strategies are generally a better choice than explorative ones, the opposite, or none of them.

Lastly, this system should take decisions in real time. Moreover the system is time-critical, there is a point after which no possible solutions exist anymore. The question of how to find the end of the negotiation time interval and what to do if the agents don't come with a compromise solution.

3.1. Members identification

Few works have been published so far regarding the topic. Among them, [6] introduces a MAS architecture in the context of free-flight. Dealing with free-flights the identification of the set of aircraft that participates in the decision making in this work is global. For the purpose of this work a more local approach proposed in [7], which considers the two conflict aircraft and the surrounding traffic, a set referred to as ecosystem, is adapted.

Surrounding traffic can be well specified. Formally defining it, an aircraft A is part of the surrounding traffic if there exist a feasible maneuver that performed during the existence time of the ecosystem by some member B of the ecosystem, or by aircraft A itself, will induce a new conflict in the ecosystem between aircraft A and B.

This definition does not specify what maneuvers are possible, giving it a high level of flexibility.

3.2. Identifying good solutions

Being in a system where each agent (in our case aircraft) seeks to maximize its own profit, makes it impossible to define a unique optimum for the whole system. This is a well-known restriction which lies in the foundations of Game Theory [8].

Several solution concepts are proposed in literature, each one having its pros and cons. Among them the two most well-known ones are Nash equilibrium and Pareto optimum.

Nash equilibrium defines a sense of rationality by having the property of being the best individual choice given the actual choices of the other agents [9]. Pareto optimum is a solution where no agent can do better without hurting some other agent [9].

The two solution concepts have a wide area of applications, however none of them is appropriate in our framework. The problem with Nash equilibrium is that quite often it is Pareto dominated, i.e. there exist another solution which is better for at least one of the aircraft and as good as the equilibrium for the rest. Being in a collaborative environment, makes this situation not desirable. The problem with Pareto optimum is that at times it can be quite bad individually. In a situation with 5 aircraft for example, a proposed solution which is as well Pareto efficient can be unacceptable for some of the aircraft.

In these conditions a hybrid solution concept, that of satisficing [10], is adopted. Under this concept each agent has to give its minimal requirements. The goodness and acceptability of a solutions is defined based on the aggregation of these requirements.

3.3. Explorative vs exploitative search strategies

The problem of solving a conflict between two aircraft can be formulated as a search problem. In doing so, the process of finding a solution would be equivalent to defining a search strategy. Search strategies can be classified to exploratory ones, in which the leading principle is to explore the search space as broadly as possible and exploitative ones, in which the leading principle is to find a solution as soon as possible [11].

It is clear that exploitative strategies converge faster. It can be also seen that one of the implications of choosing satisficing as our solution concept makes the system free of local optima. Considering these two observations, exploitative search strategies are the obvious choice.

3.4. Deadlock

The system being a real-time system and time-critical raises two questions. Firstly, how to calculate the deadlock, i.e. the time instance after which no possible solution exists. Secondly, if this point is reached, what decision should be made.

For the first point, there is some ongoing research combining simulations and analytical techniques. The results are yet to be published, but it can safely assumed that the deadlock can be calculated early enough. Moreover, there are constructive ways of calculating the deadlock, which means that by finding the deadlock we find at the same time the latest possible conflict-free configuration of the system. This provides an answer to the second raised question as well. If the deadlock is reached, the latest possible solution is imposed to the system without the necessity of approval of the involved agents.

4. THE MAS FRAMEWORK

After clarifying the principle of how surrounding traffic identification can be performed, choosing a way of identifying good solutions, arguing that exploitative search strategies are the natural choice for our problem and accepting that the deadlock together with the last possible solution can be found in time, we are ready to provide a concrete MAS framework.

In this framework, each aircraft will act as an independent agent who takes part in the negotiation. An extra agent[1] with monitoring purposes and the power and ability to stop the negotiation process at each desired moment is included in the system.

4.1. Agents' interaction

One of the implications of having an exploitative search strategy is that we can split the decision-making process into steps and use a "divide and conquer" approach. More explicitly, at the dawn of the negotiation process the conflict aircraft can start searching for an agreement between them and only later, if necessary, involve the surrounding traffic aircraft.

[1] This agent could be an artificial one i.e. a computer agent, or a human one i.e. an ATC controller.

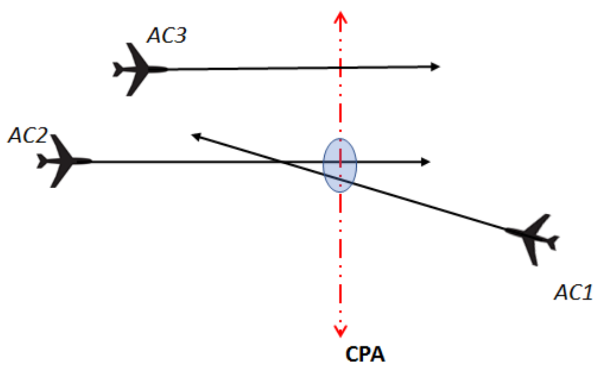

Figure 3 Three aircraft example. Black arrows are the trajectories of the three aircraft, while the red one shows the position of the conflict aircraft when they are at the closest point of approach (CPA).

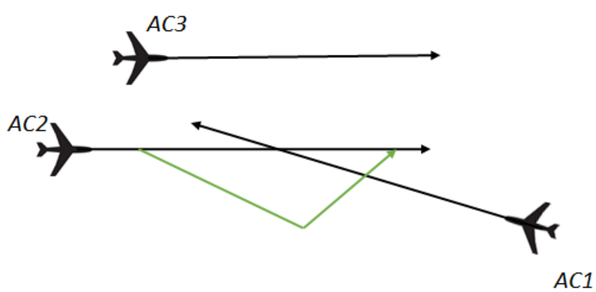

Figure 4 Possible agreement between AC1 and AC2 that will not cause a new conflict and therefore AC3 does not get involved in negotiation. The green arrow represents the proposed deviation for AC2.

In Figure 3 an example with three aircraft involved is taken to illustrate the principle. If AC1 and AC2 would start the negotiation between them and come up with a common proposal as in Figure 4. In this case, no induced conflict appears and therefore there is no need for AC3 to get involved in the negotiation process. This is not the case in the scenario depicted in Figure 5. The proposal here induces a new conflict between AC2 and AC3 and so AC3 should be called to take part in the negotiation process. In this case AC2 and AC3 will start a new negotiation process to see if they can come up with an agreement. If yes, a solution is found. On the other hand, if there is no agreement between the two, the initial proposal is refused and AC1 with AC2 restart their negotiation to find a different proposal.

This approach comes at the cost of possible domino effect during which the pairwise negotiations can increase exponentially with the total number of aircraft. To take care of this situation, the monitoring agent is introduced. Based on its criteria, this agent can decide to stop a negotiation branch[2] and lead the agents towards a discussion of other possibilities, or force them to accept the compulsory resolution.

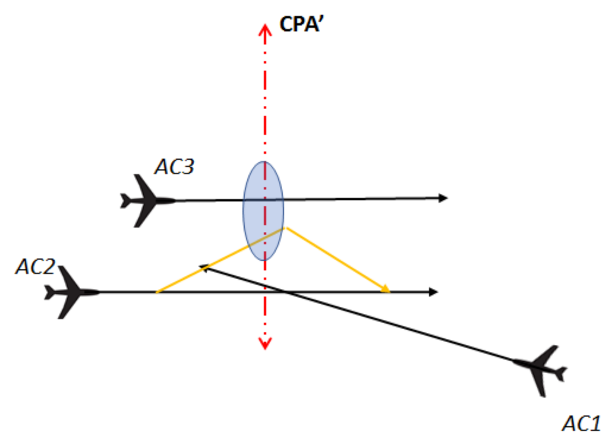

Figure 5 Possible agreement between AC1 and AC2 that will induce a new conflict between AC2 and AC3. In this case AC3 should be called to the negotiation process.

4.2. Pairwise negotiation

The “divide and conquer” approach introduced on the previous subsection left undescribed the pairwise negotiation process. The structure of this process depends heavily on the nature of the search space, i.e. pairwise interaction in a discretized space, as in [12], can be drastically different from an interaction in a continuous space. This fact, combined with the purpose of this material which is to give a general MAS framework that can be applied to airborne conflict problems, makes a further specification of the pairwise negotiation be beyond the scope of work.

5. CONCLUSSIONS

[2] A negotiation brunch is made up of pairwise negotiations that were produced by related induced conflicts

This article introduces a MAS framework that can be used in solving en route conflicts. Critical issues where identified and based on them, a framework was proposed.

The concept of the ecosystem is presented as a way of defining and finding the set of aircraft that need to take part in the solution process. The given definition is given using the allowed maneuvers, which are not explicitly specified in order to give the framework some generality and flexibility.

The solution concept of satisficing is proposed as a way of finding solutions which are individually efficient and in a sense rational.

Given satisficing concept, it is argued that exploitative search strategies suit better than exploratory ones. This statement relies on two facts. Firstly, exploitative strategies converge faster. Secondly the nature of satisficing evicts the problem of having a solution which is individually not good enough.

It is argued further that for such a system to be realizable, the deadlock of the system should be identified in advance. The last possible solution should be known as well and in case the agents cannot come to a compromised they should be forced to accept it.

Using the above principles, a framework with two kinds of agents, negotiating and monitoring ones, can be defined. Negotiating agents represent the involved aircraft and they are responsible to perform the collective search for a solution. The monitoring agent is responsible to give a guidance during certain situations in the search process, as well as firing the compulsory solution when necessary.

6. ACKNOWLEDGEMENT

This research is partially supported by the H2020 Research and Innovation Programme, Project: Adaptive self-Governed aerial Ecosystem by Negotiated Traffic (Grant Agreement No. 699313) and by the H2020 Research and Innovation Programme, Project: cooPerative depArtuRes for a compeTitive ATM networK sErvice (Grant Agreement No. 699307). Opinions expressed in this article reflect the authors' views only.

7. REFERENCES

[1] O. Gluchshenko and P. Foerster, "Performance based approach to investigate resilience and robustness of an ATM System," *Tenth USA/Europe Air Traffic Manag. Res. Dev. Semin.*, p. 7, 2013.

[2] J. Tang, M. A. Piera, and O. T. Baruwa, "A discrete-event modeling approach for the analysis of TCAS-induced collisions with different pilot response times," *Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng.*, 2015.

[3] Murugan S. and Oblah A. A., "TCAS Functioning and Enhancements," *Int. J. Comput. Appl.*, vol. 1, no. 8, pp. 46–50, 2010.

[4] AGENT team, "Report on AGENT functional and non-functional requirements," 2016.

[5] C. Munoz, A. Narkawicz, and J. Chamberlain, "A TCAS-II Resolution Advisory Detection Algorithm," *AIAA Guid. Navig. Control Conf.*, 2013.

[6] J. C. Hill, J. K. Archibald, W. C. Stirling, and R. L. Frost, "A multi-agent system architecture for distributed air traffic control," *Network*, 2005.

[7] J. Homdedeu, M. del M. Tous, M. A. Piera Eroles, T. Koca, and M. Radanovic, "A comparative analysis of different methods for identification of the evolution of the number of possible conflict-free airspace configurations including multiple aircraft and a single conflict," in *EMSS*, 2017, p. 7.

[8] Y. S. and K. Leyton-Brown, Y. Shoham, and K. Leyton-brown, "Multiagent Systems: Algorithmic, Game-theoretic, and Logical Foundations by Y. Shoham and K. Leyton-Brown Cambridge University Press, 2008," *SIGACT News*, 2010.

[9] K. Leyton-Brown and Y. Shoham, *Essentials of game theory*. 2008.

[10] H. Simon, "Rational choice and the structure of the environment.," *Psychol. Rev.*, 1956.

[11] M. Lones, "Sean Luke: essentials of metaheuristics," *Genet. Program. Evolvable Mach.*, 2011.

[12] M. Radanovic, M. A. Piera, T. Koca, and F. J. Saez, "Airborne ecosystem complexity analysis for tactical conflict management," in *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS*, 2017, p. 10.

8. COPYRIGHT