

## Formal modelling and verification of a multi-agent negotiation approach for airline operations control

Bouarfa, Soufiane; Aydoğan, Reyhan; Sharpanskykh, Alexei

**DOI**

[10.1007/s40860-020-00123-0](https://doi.org/10.1007/s40860-020-00123-0)

**Publication date**

2021

**Document Version**

Final published version

**Published in**

Journal of Reliable Intelligent Environments

**Citation (APA)**

Bouarfa, S., Aydoğan, R., & Sharpanskykh, A. (2021). Formal modelling and verification of a multi-agent negotiation approach for airline operations control. *Journal of Reliable Intelligent Environments*, 7(4), 279-298. <https://doi.org/10.1007/s40860-020-00123-0>

**Important note**

To cite this publication, please use the final published version (if applicable).  
Please check the document version above.

**Copyright**

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

**Takedown policy**

Please contact us and provide details if you believe this document breaches copyrights.  
We will remove access to the work immediately and investigate your claim.



# Formal modelling and verification of a multi-agent negotiation approach for airline operations control

Soufiane Bouarfa<sup>1,2</sup> · Reyhan Aydoğan<sup>3,4</sup> · Alexei Sharpanskykh<sup>5</sup>

Received: 18 March 2020 / Accepted: 28 November 2020  
© The Author(s) 2021

## Abstract

This paper proposes and evaluates a new airline disruption management strategy using multi-agent system modelling, simulation, and verification. This new strategy is based on a multi-agent negotiation protocol and is compared with three airline strategies based on established industry practices. The application concerns Airline Operations Control whose core functionality is disruption management. To evaluate the new strategy, a rule-based multi-agent system model of the AOC and crew processes has been developed. This model is used to assess the effects of multi-agent negotiation on airline performance in the context of a challenging disruption scenario. For the specific scenario considered, the multi-agent negotiation strategy outperforms the established strategies when the agents involved in the negotiation are experts. Another important contribution is that the paper presents a logic-based ontology used for formal modelling and analysis of AOC workflows.

**Keywords** Workflow modelling · Rule-based modelling · Formal modelling · Multi-agent negotiation · Model checking · Airline operations control

## 1 Introduction

Airlines cope with many disruptions of different nature that implicitly or explicitly test their resilience on a regular basis. These disruptions may interact with each other, potentially creating a cascade of emerging disturbances that may span over different spatial as well as time scales, ranging from affecting only one aircraft or crew, up to a group of aircraft [1]. In current airline operations, disruptions are managed by Airline Operations Control (AOC), and may impact the economic performance of the airline and customer service. E.g., some flights are rerouted, some aircraft are leased, and some

flights are re-booked. Consideration of the aircraft routings, crew, maintenance, weather, customer needs, security and turnaround processes complicate AOC. Current AOC practice consists of a coordination process between many human operators, each of which plays an essential role in disruption management (see Fig. 1). With the ever-growing complexity and various types of interdependencies between airlines, airports, and ATC centres, maintaining airline resilience to expected and unexpected disruptions becomes a challenging task [2]. To manage disruptions in a resilient way, advanced forms of coordination between human operators and automation are required. This paper aims at evaluating a new coordination approach based on multi-agent negotiation and comparing it with existing strategies in the context of a realistic operational scenario.

Automated multi-agent negotiation in which smart software entities negotiate with each other or humans, has taken the attention of the AI research community in recent years. The research in this field varies from designing and developing negotiation protocols [3] to bidding strategies [4] and opponent modeling [5]. In the AOC domain, this has only been explored by one researcher [6]. However, automated negotiation has been applied in many fields especially in the supply chain domain. For instance, Chen et al. [7] implemented a dynamic supply chain simulation where the seller

---

✉ Soufiane Bouarfa  
soufiane@delftaviation.com

<sup>1</sup> Abu Dhabi Polytechnic, Al Ain Campus,  
United Arab Emirates

<sup>2</sup> Delft Aviation, Delft, The Netherlands

<sup>3</sup> Department of Computer Science, Özyeğin University,  
İstanbul, Turkey

<sup>4</sup> Interactive Intelligence, Delft University of Technology,  
Delft, The Netherlands

<sup>5</sup> Faculty of Aerospace Engineering, Delft University  
of Technology, Delft, The Netherlands

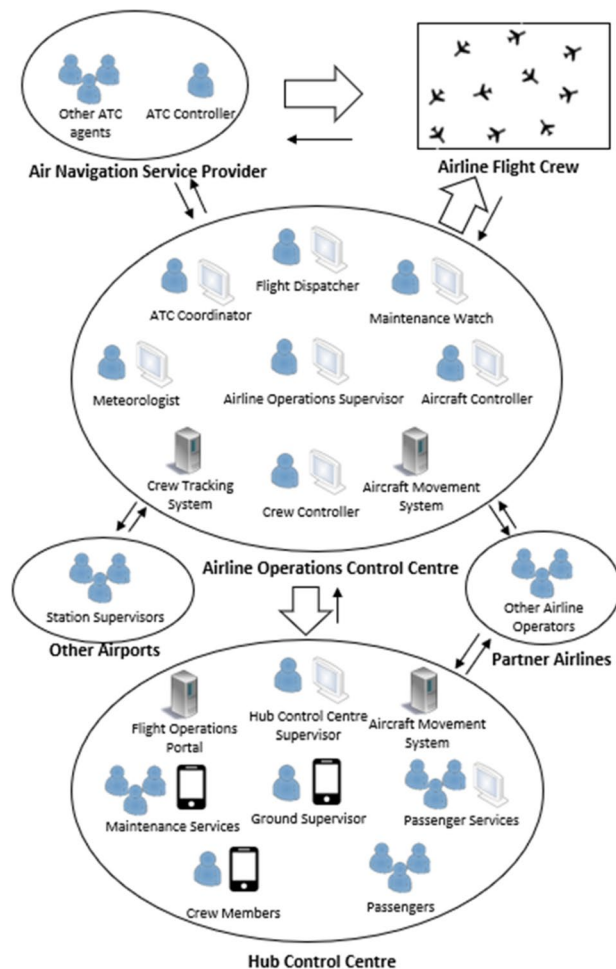


Fig. 1 Coordination in airline operations control

and buyer agents negotiate with each others. In this study, two negotiation protocols are used namely a pair-wise and auction protocol. In the pair-wise negotiation protocol, one of the agents in the supply chain sends an offer to another agent, and negotiation starts when the other agent replies to the offer. The response can be a counter-propose or accept. The negotiation ends when an agreement is reached. In the auction protocol, one of the seller agents informs an auctioneer about the good to be sold and the highest desired price. Then the auctioneer broadcasts the message to potential bidders and receives their bid. Finally, an auction is organized according to the seller's bid. The process is repeated until the seller reaches an agreement with one of the bidders and auctioneer announces the results to the agents involved in the negotiation. Nguyen and Jennings [8] introduced a flexible commitment model in which multiple agents negotiate in a bilateral fashion concurrently. There is a global deadline which imposes a time constraint on all negotiations. Before this deadline, the negotiator agents can reach non-binding

intermediate agreements, and all of those are finalized after the global deadline is expired.

Wang et al. [9] developed an ontology-based supply chain model to alleviate human interactions in the supply chain. Each agent has a private ontology that stores his negotiation strategy (e.g. concession, bidding, acceptance). The public ontology includes shared information such as negotiation deadlines, negotiation issues, and their domains. Cui-Hong [10] stated that the collaboration of supply chain agents is essential to achieve a dynamic, flexible, and agile supply chain. He introduced supply chain coordination models for communication between supply chain agents. Rady and El-Shorouk [11] proposed a multi-agent system application where the agents form a network to work collaboratively to manage procurement, selling, and scheduling. Each agent is assigned with a separate task and their collaboration forms the supply chain management strategy. Xue et al. [12] proposed an agent-based negotiation platform for construction supply chain environments, to improve the efficiency of decision-making while collaborating with other agents. In their platform, there exists a set of specialized agents which negotiate by adopting multi-attribute negotiation theory. In the system, a general contractor agent negotiates with multiple agents such as subcontractor agents, supplier agents, designer agents, and owner agents. During the negotiation, agents exchange offers until an agreement is reached. The bids are proposed according to feasible solutions for each agent, and if no solution is found for one agent, he may reject and inform the other agent to alter his offer. In case the solution space of the other agent is empty, the negotiation may end with a failure. Each agent evaluates the utility of the offers by his and the other party's utility into account to reach a pareto optimal solution.

Depending on the characteristics of the given problem, the interaction among agents may differ. There are a variety of negotiation protocols proposed in the literature. For instance, the Stacked Alternating Offers Protocol (SAOP) [13] governs the interaction among agents in a turn-taking fashion. One of the agents initiates the negotiation by making an offer. The next agent in line can accept this offer or make a counteroffer by overriding the previous offer or end the negotiation. This process continues until reaching a mutual consensus or reaching a deadline. In the Single Text Mediated Protocol [14], there is an unbiased mediator searching for an agreement without knowing each agent's preferences. The mediator initiates the negotiations by making a random bid and asks each agent to vote for or against the offer. When all agents accept the given offer, the mediator keeps this offer as the most recent accepted bid. In the next round, the mediator only changes the value of one of the issues and asks agents to vote for or against the modified offer. Other protocols include the Feedback based Protocol [15] where agents can give insightful feedback to mediator

rather than simply voting to accept and reject and the Intra-team negotiation protocol [16] for governing the interaction of a team with their opponent. While some of these protocols involve an unbiased mediator, which aims to help negotiating agents to find a consensus; others focus on the interaction among only negotiating agents. To model negotiation in AOC, the authors applied the single text mediated protocol. In this protocol, a team representative acts like a mediator to reach a unanimous agreement by making offers according to his/her preferences and asking other agents to vote for or against the given offers. This protocol is compatible with AOC in which the supervisor makes the final decision upon feedback from other experts.

This paper applies the Single Text Mediated Protocol to airline disruption management using mathematical logic. The motivation for choosing mathematical logic as a formal language for the specification of the system under consideration is twofold. On the one hand, it provides a natural, close to human, expressive specification language based on ontologies. Using this language, diverse quantitative and qualitative aspects of negotiation domains could be represented, including temporal and spatial dimensions. Logic-based languages are frequently used to describe coordination in multi-agent systems [17]. On the other hand, logic-based specifications can be analyzed using formal verification methods and tools such as TTL, model checking [18], often in an automated and systematic manner. This advantage is essential for building reliable intelligent (decision support) systems. The paper is organized as follows. Section 2 gives background about airline operations control. Section 3 presents the simulated strategies. Section 4 explains the research methodology used to develop the multi-agent system for the considered case study and presents the results. Finally, Sect. 5 provides key conclusions of this work.

## 2 Airline operations control

The idea of monitoring and controlling a transport network in real time is not new. The concept was first established in the nineteenth century in the railway industry when the development of the telegraph made it possible for the information to travel faster than physical transport [19]. This allowed for a central location in which real-time information about the current status of the network could be collected and acted upon. Today, the concept of monitoring operations in real-time is used across industries, with AOC as one example.

Airline disruption management starts when airline planning ends (Fig. 2). The scheduling process starts with publishing a preliminary timetable up to 1 year before the day of operations. The timetable provides the basis for the aircraft schedule, which assigns an aircraft type to each flight. With

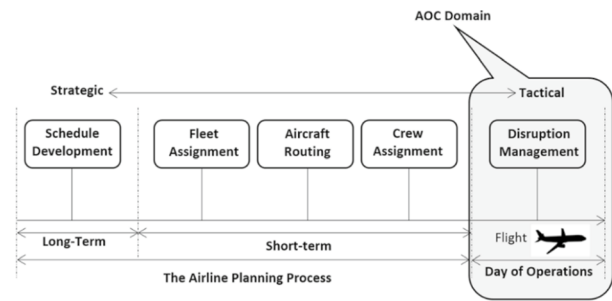


Fig. 2 Airline planning and airline disruption management

the flights and aircraft types known, crew pairing defines the amount and type of crew per flight. The next step is to assign specific aircraft and individual crewmembers to each flight in the tail assignment and crew rostering phase. After publishing the crew roster, crew members can request changes in their schedule in the roster maintenance phase. Disruption management starts after the airline planning process ends and is considered a tactical step during recovery [20, 21].

During the day of operations, the airline schedule is subject to many disruptions. The four main airline schedule disruptors are aircraft mechanical problems, severe weather, airport congestion, and industrial action (e.g. strikes). The goal of AOC is to deliver customer promise despite these disruptions. In doing so, it should minimize airline costs incurred during recovery, and return to the original schedule as soon as possible [22].

Disruptions affect the aircraft, crew, and passenger resources of an airline. Managing these resources is the duty of AOC operators. Each AOC operator has its own role. Such roles might vary per airline, but six are common to most airlines: flight dispatch, aircraft control, crew tracking, aircraft engineering, customer service, and Air Traffic Control (ATC) coordination [22]. Because the airline operations supervisor is ultimately responsible for AOC operations [23], he/she has the authority to make changes in the nominal schedule.

An airline controller can manage disruption in many different ways. To resolve a problem that affects the aircraft resource, a flight can be delayed, cancelled, rerouted, or the aircraft exchanged. Crew related problems can also be resolved by cancelling or delaying the flight, or by calling in new crew or reassigning existing crew. To resolve a passenger problem, an operations controller might change the passenger's flight or delay the passenger [6, 24].

How well disruptions are managed depends on how AOC is organized. For example in Europe, AOC often performs the task of flight following, while flight planning and dispatch are often performed outside AOC [22], whereas in North America, flight dispatchers and planners are assumed to make an integral part of AOC [23, 25, 26]. In the current

paper, we adopt the latter, which is also in line with [6, 27, 28].

According to Castro [29] and Machado [30], there are three types of AOC centers. A decision center, a hub control center, and an integrated control center. In a decision center, airline controllers are located in the same space while other functional groups such as maintenance services and crew control are located in a different physical space. A hub control center oversees the activities at the hub, which may include ground and passenger services, but other operations such as aircraft control are monitored from a different location. An integrated airline operational control center integrates all functional groups under the same physical location. The research presented in this paper considers an integrated control center.

Work practice differs from airline to airline and from individual to individual. Smaller airlines tend to use schedule visualization software to easily enable their controllers detect irregularities, while major airlines use software that is able to automatically detect these irregularities. Operators with similar roles sit next to each other to easily communicate and collaborate. Each desk keeps the necessary communication equipment such as phone and telex. Centrally placed screens show live news, as well as weather reports and performance indicators. Clocks indicate time in different time zones around the world [31].

### 3 AOC disruption management strategies

This paper evaluates four AOC strategies using multi-agent system modelling. Three strategies are based on current AOC practices S1–S3 and the fourth strategy is based on the Single Text Mediated Protocol [4]. The research goal is to understand the impact on airline performance if the agents strictly follow these strategies for the chosen scenario. This section gave details about the simulated strategies.

#### 3.1 Current AOC strategies S1–S3

To select representative AOC strategies and make a clear distinction between them, a critical element is the understanding of how AOC operators make their decisions in relation to various aspects during disruption management. Bruce [27, 32] has systematically studied the decision-making processes of 52 experts from six AOC centers. The 52 experts are airline controllers representative of AOC operators (e.g. in terms of gender, age, years of experience in the airline industry, years of experience in the AOC domain, and previous occupation). These controllers were presented with six different types of scenarios in real-time simulations and were asked how they would manage the disruption. There were two types of scenarios simulated: domestic scenarios, and international scenarios. The simulations were designed by two independent experts. One expert had extensive experience in international AOC operations and the other expert was experienced in domestic AOC operations. The value of expert input into the simulation design was to ensure the simulations were representative of realistic disruptions. Each simulation was conducted between the researcher and one participant in a separate room away from the operations room. Prior to the simulations, each participant listens to a short briefing before starting with managing the disruption. At each scenario, participants were asked to think aloud as they considered the operational disruptions. More details about the simulation design can be found in Bruce [27]. All comments of controllers were recorded and transcribed verbatim. These comments were analysed by Bruce [27, 32] who found out that airline controllers use strategies with three different levels of performance. These strategies are described below and summarized in Table 1.

*AOC strategy S1* Elementary level of performance: airline controllers identify various basic level considerations such as aircraft patterns and availability, crew commitments and maintenance limitations. For example, when a maintenance problem is reported, controllers at this level appear to acknowledge the information provided and begin considering the basic consequences of the scenario. They also

**Table 1** Overview of the three AOC strategies S1–S3 in relation to various disruption management aspects based on simulations with 52 experts

Aspect	AOC strategy S1	AOC strategy S2	AOC strategy S3
Maintenance Information	Accept information source and content, and act on information given about a maintenance situation	Challenge/query information about a maintenance situation	Seek alternative information and recheck source and reliability
Crewing	Await crew from inbound aircraft	Challenge crew limits/ Seek extensions to crew duty time	Seek alternative crew (e.g. from nearby base or other aircraft)
Curfews	Curfews are not taken into account	Identify curfews and work within them	Seek curfew dispensation
Aircraft	Seek first available aircraft	Request high-speed cruise	Combine flights to free up aircraft

See Appendix 2 for detailed rules corresponding to the three strategies



identify opportunities to replace the aircraft or rebook passengers on alternative flights (Table 2).

*AOC strategy S2* Core level of performance: airline controllers have a greater comprehension of the problem. They take into account the more complex consequences of the problem than those evident at the elementary level. Several constraints such as crew restrictions, slot times, and curfews are identified at this level. Controllers, would for instance negotiate maintenance requirements and crew limitations to overcome the risk of breaching the curfew.

*AOC strategy S3*– Advanced level of performance: airline controllers demonstrate thinking beyond the immediacy of the problem. They examine creative ways to manage the disruption. For instance, controllers at this level would consider more complex crewing alternatives such as positioning a crew from one airport to another airport where the flight crew is needed. Also, in the case of a maintenance problem, controllers at this level would seek alternative information and recheck the reliability of the information, e.g. through organizing a conference call with the maintenance watch people (Table 3).

### 3.2 Automated multi-agent negotiation strategy S4

This paper evaluates a fourth strategy that uses multi-agent negotiation following the Single Text Mediated Protocol [14]. In this context, the AOC supervisor has the power of making the final decision based on the feedback given by other agents and it also needs other agents' expertise to generate potential solutions for the underlying problem. For example, if the problem is related to aircraft, it is required that the aircraft controllers inform the AOC supervisor about all possible aircraft solutions. Since the given solutions may have an influence on other agents' inner processes, it is required to find a consensus among all agents. Accordingly, the proposed negotiation approach works as follows:

#### Pre-negotiation phase

- Upon identification of a problem, the AOC supervisor asks the specialist agents to provide all possible solutions corresponding to their problem dimension within a certain deadline.
- All specialist agents provide their potential solutions to the AOC supervisor. The specialist agents include the aircraft controller agent ACo, the crew controller agent CCo, and Passengers Services agent PS.

**Table 2** Overview of strategy S1 rules that AOC agents should comply to in the context of the considered aircraft mechanical breakdown scenario

Agent	Strategy S1 in the context of the mechanical breakdown scenario
AE	The engineer should inform the station supervisor by radio as soon as he detects a mechanical problem The engineer should accept the information provided by the station supervisor, and act according to this information
SS	The station supervisor should accept the information provided about an aircraft mechanical problem by the engineer The station supervisor should report through the Aircraft Movement System in case he believes there is a problem affecting the flight The station supervisor should accept the information provided by the Airline Operations supervisor and act according to this information
AMS	Any information registered in the Aircraft Movement System must be synchronized with the Crew Tracking System
AOS	The Airline Operations Supervisor should continuously monitor the Aircraft Movement System The Airline Operations Supervisor must accept the information source In case of a disruption, the Airline Operations Supervisor should ask the Aircraft Controller and Crew Controller to provide solutions about their problem dimensions The Airline Operations Supervisor must accept the recommendations of both the aircraft controller and crew controller before making a final decision
ACo	The Aircraft Controller should continuously monitor the Aircraft Movement System. This enables him to monitor both disruptive events that might affect the flights, and ongoing disruption management processes The Aircraft controller must accept the information source about aircraft In case of an aircraft mechanical failure, the Aircraft Controller is responsible for solving the aircraft problem. He should always seek the first available aircraft and report his solution through the Aircraft Movement System
CTS	Any information registered in the Crew Tracking System must be synchronized with the Aircraft Movement System
CCo	The Crew Controller should continuously monitor the Crew Tracking System to identify crew related events that might affect the progress of flights (e.g. crew sign on and duty time) The crew controller should accept all information reported on his Crew Tracking System In the case of crew-related disruptions, the Crew Controller should provide a solution to the crew problem through the Crew Tracking System. To solve the crew problem, he should await crew from inbound aircraft

**Table 3** Overview of strategy S4 rules that AOC agents should comply to in the context of the considered aircraft mechanical breakdown scenario

Agent	Strategy S4 in the context of the mechanical breakdown scenario
AE	The engineer should inform the station supervisor by radio as soon as he detects a mechanical problem The engineer should accept the information provided by the station supervisor, and act according to this information
SS	The station supervisor should accept the information provided about an aircraft mechanical status by the engineer The station supervisor should report through the Aircraft Movement System in case he believes there is a problem affecting a specific flight The station supervisor should accept the information provided by the Airline Operations supervisor and act according to this information
AMS	Any information registered in the Aircraft Movement System must be synchronized with the Crew Tracking System
AOS	The Airline Operations Supervisor should continuously monitor the Aircraft Movement System The Airline Operations Supervisor must ask the specialist agents to provide a solution within 1.5 min in case a mechanical problem has been reported The Airline Operations Supervisor must announce a solution to the specialist agents after he receives input from them The Airline Operations Supervisor must accept the solution voted by the specialist agents The Airline Operations Supervisor must inform the Station Supervisor about the final solution
ACo	The Aircraft Controller should continuously monitor the Aircraft Movement System The Aircraft controller must provide a solution to the Airline Operations Supervisor upon his request within the specified deadline. He should combine flights to free up aircraft (level 3 decision making) and report his solution through the Aircraft Movement System The Aircraft Controller must vote for the solution announced by the Airline Operations Supervisor
CTS	Any information registered in the Crew Tracking System must be synchronized with the Aircraft Movement System
CCo	The Crew Controller should continuously monitor the Crew Tracking System to identify crew related events that might affect the progress of flights (e.g. crew sign on and duty time) The Aircraft controller must provide a solution to the Airline Operations Supervisor upon his request within the specified deadline. He should always seek alternative crew from another aircraft or a nearby base (Level 3 decision-making) The Aircraft Controller must vote for the solution announced by the Airline Operations Supervisor

- If the AOC supervisor does not receive solutions from all three specialist agents, the disruption cannot be managed.

### Negotiation phase

- The AOC supervisor evaluates all proposals received from the specialist agents and selects one of the solutions according to his bidding strategy. The AOC supervisor announces his chosen solution to the specialist agents.
- The specialist agents vote for or against the announced solution by the AOC supervisor. Note that the specialist agents may use different criteria to evaluate the offer (e.g., cost, safety, crew satisfaction, etc.)
- If all three agents agree about the solution, the negotiation ends with the current solution successfully.
- If no consensus is received, the AOC supervisor makes a new offer for the three agents to vote on. In the meantime, it keeps the offer which was accepted by the majority and updates this offer over time. Note that this process is repeated until reaching an agreement or the deadline.
- If the agreement is not reached before the deadline, the AOC supervisor ends the negotiation with the compatible offer that has the most favorable votes.

Preferences of agents can be modelled quantitatively or qualitatively [33]. In the quantitative approach, utility functions are mostly used to assess the value of each offer. A utility function maps each offer to a real value mostly between zero and one. The most desired outcome has a utility of 1.0. When specialist agents have many evaluation criteria, they mostly evaluate the individual satisfaction levels in terms of individual utility gained per each criterion and aggregate them to calculate an overall utility to make their decision. For this purpose, additive utility functions are used. When agents use such models, they have a reservation utility threshold which determines whether the outcome is acceptable or not. If the overall utility is below threshold value, the outcome is not acceptable. Furthermore, the satisfaction level for each criterion might be different which creates tradeoffs in negotiation. While agents may gain on one criterion, they may lose on other criteria. If all criteria must be satisfied, then these are not considered as preferences, but more hard constraints. If an outcome does not satisfy a hard constraint, then there is no need to calculate its utility because it will not be accepted by the agent at all.

## 4 Research methodology

To develop a multi-agent system model of the AOC strategies, the authors make use of the modelling methodology of Nikolic and Ghorbani [34] which consists of five steps: System analysis, model design, detailed design, software implementation.

### 4.1 System analysis

The main objective of this step is to analyze and identify the socio-technical system under investigation. For this purpose, the authors selected a challenging disruption scenario that is well described and evaluated in the literature. This scenario was presented by Bruce [27, 32] to a group of airline controllers to understand what they perceive in their environment, and how they reason and make decisions to manage the disruption. All participating controllers were representative of AOC operators in terms of gender, age, years of experience in the airline industry and AOC. The outcome of this step is the identification of agents, their behavior, the environment in which they operate.

#### 4.1.1 Scenario description

The scenario concerns a mechanical problem with an aircraft at Charles de Gaulle (CDG) airport, aiming for a long-haul flight to a fictitious airport in the Pacific, which is indicated by the code PCF. The overview of flights being monitored by the airline controller at the time of disruption is shown in Fig. 8 of Appendix 1. The time is 0655. Flight 705 is unserviceable in Paris (CDG). The engineers report that it has a hydraulic leak such that it may require a hydraulic pump change. If so, then they expect the pump change to take two hours. On this advice, the staff at CDG have stopped checking passengers in for Flight 705. After participants were given time to consider this situation, subsequent information was provided that confirmed the hydraulic pump change and advised that due to inclement weather, the maintenance work would be done in the hangar, delaying a possible departure considerably more than initial advice.

This scenario requires participants to consider strategies and consequences to resolve the delay caused by the unserviceable aircraft. The flight was progressively delayed at CDG for 3 h due to mechanical unserviceabilities, to the extent that the operating crew were eventually unable to complete the flight within their legal duty time.

In [27], this scenario was considered by a panel of AOC management experts. They developed several alternatives, and subsequently identified the best solution, which was to re-route the flight from CDG to PCF and to include a stop-over in Mumbai (BOM). In parallel, a replacement flight

crew was flown in as passengers on a scheduled flight from PCF to BOM to replace the delayed crew on the flight part from CDG to PCF (see Fig. 3). The question, therefore, is how well the outcome of the multi-agent system modelling and simulation of AOC strategies compare to the expert panel in finding a best solution?

#### 4.1.2 Identification of agents involved

Since the purpose of the simulation model is to evaluate different AOC strategies, the main agents are those human operators involved in managing the disruption and the decision support systems they use. Operational workflows from a European airline were used [30] to identify the different kinds of technical systems being used in case of a mechanical breakdown, the interactions of agents with these systems. Figures 4 and 5 show example workflows corresponding to both strategy S1 and strategy S4.

### 4.2 Model design

Once the key agents have been identified, their behavior in the context of the considered scenario is accurately specified according to the different rules they must adhere to (see Appendix 2). These rules are either based on established airline strategies S1–S3 [32] or the Single Text Mediated Protocol S4. Having identified the relevant agents in the previous step, this step aims at a high-level design of the interactions between different agents according to the prescribed strategy. The outcome of this step is the assignment of rules to agents.

### 4.3 Detailed design

This step aims at developing the ontology of the socio-technical system. The ontology formally captures the information flow and interactions between agents during disruption management. In this research, the authors used the Temporal

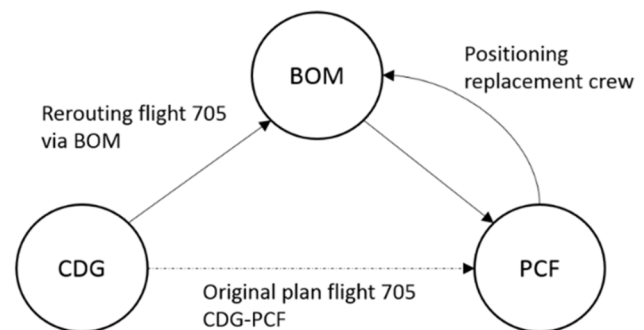
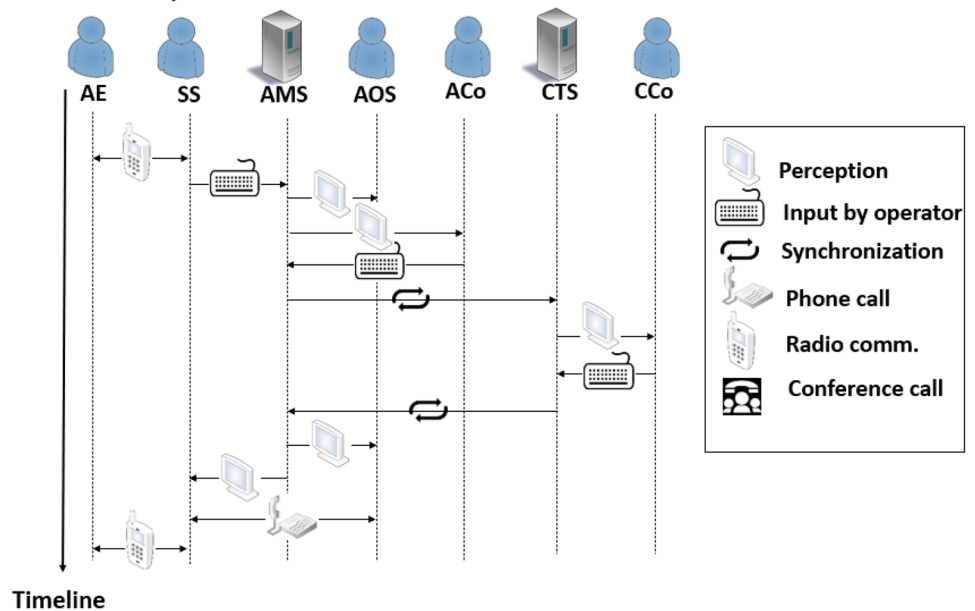


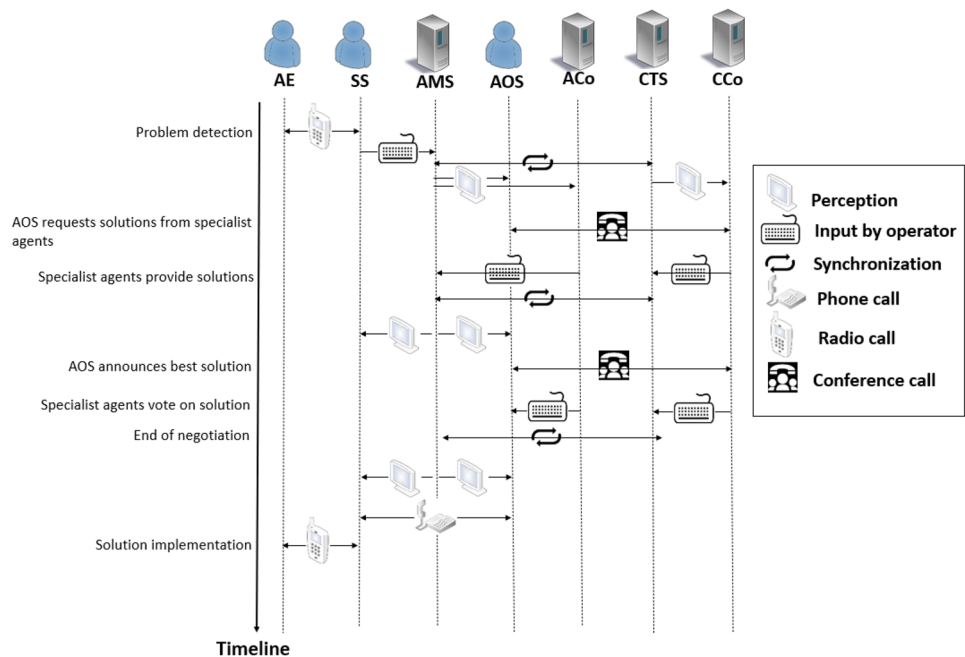
Fig. 3 Best solution identified by the expert panel



**Fig. 4** Operational Workflow corresponding to AOC strategy S1. *AE* aircraft engineer, *SS* station supervisor, *AMS* aircraft movement system, *AOS* airline operations supervisor, *ACo* aircraft controller, *CTS* crew tracking system, *CCo* crew controller



**Fig. 5** Operational Workflow corresponding to AOC strategy S4. *AE* aircraft engineer, *SS* station supervisor, *AMS* aircraft movement system, *AOS* airline operations supervisor, *ACo* aircraft controller, *CTS* crew tracking system, *CCo* crew controller



Trace Language (TTL) which has been used in different multi-agent case studies [35].

#### 4.3.1 Domain ontology: logical predicates

See Table 4.

#### 4.3.2 Domain ontology: sorts and elements

See Table 5.

## 4.4 Software implementation

The model has been implemented in LEADSTO, which is a software that simulates dynamic processes. The dynamic processes are modelled through specifying direct dependencies between state properties. The model is verified to check if the agents act according to the specific strategies through checking if certain properties hold. The simulation results are a specification of all the states and state properties referred to as a trace.

**Table 4** Logical predicates

Predicate	Description
Internal states and communication activities of the agents	
Observation (A,I)	Agent A observes information I from the world
Belief (A, I)	Agent A believes that information element I is true in the world
Incoming_communication(A, C, I)	Agent A receives message type C with content I
Communicate_from_to(A, B, C, I)	Agent A communicates to agent B message type C with content I
Other predicates used for this scenario	
Disruption(DT,AC,AP)	Describes a disruption of type DT, concerning aircraft with registration code AC, at airport AP
Query(A, B, I)	Query by agent A to agent B about Information I
Query_disruption(DT,AC,AP)	Query about disruption (DT,AC,AP)
Flight_crew(AC)	Flight crew of aircraft with registration code AC
Reserve_aircraft(amount)	To denote the number of reserve aircraft available
Aircraft_available_for_swap(amount)	Number of aircraft within the same type available for swap
Crew_inbound_aircraft(amount)	To denote the number of crew available from inbound flights
Aircraft_problem(AS)	Proposed solution to the aircraft problem
Crew_problem(CS)	Proposed solution to the crew problem
extend_crew_hours(y/n)	Possibility to extend crew hours (yes/no)
Check_disruption(DT,AC,AP)	Checking information reliability about a disruption of type DT, concerning aircraft with registration code AC, at airport AP
Disruption(t/f)	Confirmation whether there is a disruption or not by local agents
Conf_call(O,D,A,B,...,N)	Conference call organized by agent O about a certain disruption D with N + 1 participants in alphabetical order
Early_serviceability(AC,DT,AP)	Request for earlier serviceability for aircraft AC with problem DT at airport AP
early_serviceability(AC,DT,AP,y/n)	Possibility for earlier serviceability of aircraft AC with problem DT at airport AP (yes/no)
Start_conf_call(O,D,A,B,...,N)	Start of conference call
End_conf_call(O,D,A,B,...,N)	End of conference call
Transmit_construal(DT,AC,AP,RT,F)	Transmitting construal of the meaning of the signal back to the sender
Construal(DT,AC,AP,RT,F)	Content of a signal being sent
Exit_reporting(DT,AC,AP,RT,F)	Signal of exiting a coordination phase (reporting) about a certain type of disruption with various attributes
Start_aircraft_problem_solving(DT,AC,AP,RT,F)	Signal of starting a new coordination phase (solving crew problem) for a certain type of disruption with various attributes
Renew_compact(AS)	Renewing the basic compact about a particular information element (See Table 6.1-basic compact)
crew_day_off(AP,y/n)	Possibility to use crew with day off at airport AP
Verify_disruption(DT,AC,AP,RT,F)	Verifying a certain disruption with different attributes

#### 4.4.1 Software environment

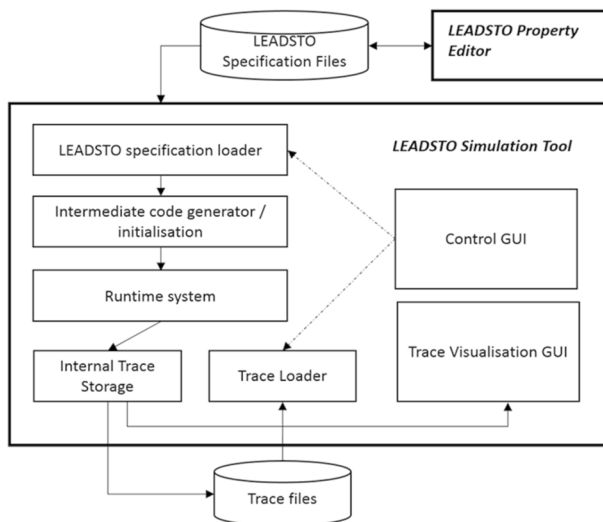
To implement agent interaction rules we made use of the LEADSTO simulation environment which uses a generic temporal-causal modelling approach [36, 37]. Use was made of the formal ontology presented in the previous section. LEADSTO proved its value in a number of projects in multi-agent systems research (e.g. in the areas of emergency response, organizational modelling, and behavioral dynamics [38, 39]. LEADSTO consists of two programs: a property editor and a simulation tool. The first is a graphical editor for constructing and editing LEADSTO specifications, and

the second is for performing simulations of the LEADSTO specifications; generating data-files containing traces for further analysis, and visualizing these traces. Figure 6 gives an overview of the simulation tool architecture and shows its interactions with the property editor. The bold rectangular borders define the two separate tools while the arrows represent the data flow, with the dashed arrows representing control.

In LEADSTO, one can specify both qualitative and quantitative aspects of complex socio-technical systems using the Temporal Trace Language (TTL). TTL has the semantics of order-sorted predicate logic [40] that is defined

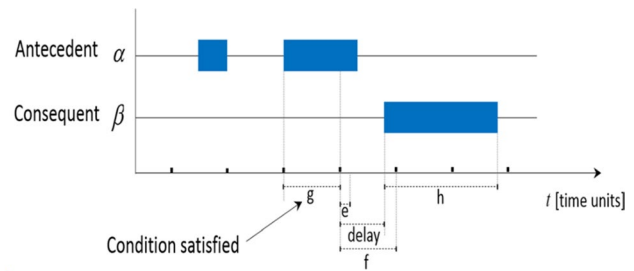
**Table 5** Sorts and elements

Sort	Elements
DISRUPTION_TYPE	{mechanical_failure}
AGENT	{AE,SS,AMS,AOS,ACo,CTS,CCo,FC,MWE}
MESSAGE	{fix_aircraft,hydraulic_leak,_no_reserve_aircraft,no_crew_available,delayed_crew,crew_hours,extend_crew_hours,no,yes,true,none,exit_AOC_disruption_management,start_crew_problem_solving}
MESSAGE_TYPE	{inform,request,permit,ask,declare,report,synchronize,confirm,answer,negotiate,check,consult,transmit,verify,announce}
AIRPORT	{CDG,BOM,PCF}
AIRCRAFT	{LHB}
AIRCRAFT_SOLUTION	{cancel_flight,fix_aircraft,no_reserve_aircraft,no_aircraft_available,pump_change}
CREW_SOLUTION	{no_crew_available,reroute_via_BOM,use_day_off_crew}
INTEGRATED_SOLUTION	{aircraft_solution,crew_solution,pax_solution}
VOTERESULT	{approval,rejection}
REPAIR_TIME	{three_hours}

**Fig. 6** LEADSTO architecture

by a rich ontological base including sorts, predicates, and variables. Relationships between system components can be expressed in a straightforward way. This provides wide means for the conceptualization of the airline disruption management domain. In addition, TTL is an extension of the standard multi-sorted predicate logic in the sense that it has explicit facilities to represent dynamic (temporal) properties of systems. Such a temporal expressivity is particularly important for the representation and analysis of processes over time.

The LEADSTO format is defined as follows: let  $\alpha$  and  $\beta$  be predicates, and  $e, f, g, h$  be non-negative real numbers. Then  $\alpha \rightarrow_{e,f,g,h} \beta$  means: If predicate  $\alpha$  holds for a certain time interval with duration  $g$ , then after some delay (between  $e$  and  $f$ ) predicate  $\beta$  will hold for a certain time

**Fig. 7** Visualizing traces in LEADSTO

interval of length  $h$ . An example of a dynamic property in the LEADSTO format is  $\alpha \rightarrow_{0.25,1,1,1.5} \beta$  where  $\alpha$  represents the predicate `Communication_from_to(external_world, AE, observe, leak)` and  $\beta$  represents the predicate `Communication_from_to(AE, SS, inform, pump_change_required)`. This property expresses the fact that, if the airport engineer AE observes that there is a hydraulic leak during 1 time unit, then after a delay between 0.25 and 1 time unit, AE will inform the station supervisor SS about the problem during 1.5 time units. By executing this rule, a trace of predicates holding true or false can be generated and visualized (see Fig. 7). The time units in this case study are in minutes. For the temporal parameters, the following assumptions were made: Solving an aircraft/crew problem takes 8 min; synchronization between IT systems takes 0.1 min following an update; an observation-belief-action cycle takes 1.5 min. These assumptions were based on observations made at two AOC centres in Europe.

#### 4.4.2 Model verification

For the identified strategies S1–S4, it is important to ensure that some required dynamic properties hold. Such properties

may for example represent system requirements, desired performance characteristics, absence of deadlocks and other forbidden states. To verify the identified strategies in the context of the case study, automated model verification tools can be used, such as TTL Checker [41]. The dynamic properties in TTL Checker need to be specified in Temporal Trace Language (TTL) [35]. LEADSTO is an executable

- *Strategy S2—property 1* If SS believes that there is a mechanical failure, then AOS should call the AE within 5 min the to verify the information. Formally:

$$\forall t \forall x:\text{AIRCRAFT} \forall y:\text{AIRPORT} \text{at}(\text{Belief}(\text{SS}, \text{Disruption}(\text{mechanical\_failure}, x, y)), t) \Rightarrow \exists t' t' > t \ \& \ t' \leq t+5 \\ \& \ \text{at}(\text{Communicate\_from\_to}(\text{AOS}, \text{AE}, \text{ask}, \text{Check\_disruption}(\text{mechanical\_failure}, x, y)), t')$$

sublanguage of TTL. TTL is also a variant of order-sorted predicate logic with the possibility to specify and reason about time. By using TTL Checker, dynamic properties in TTL could be checked automatically on simulation traces automatically generated by LEADSTO software based on multi-agent system model specifications.

- *Strategy S1—property 1* If the Station Supervisor (SS) believes that there is a mechanical failure, then within 5 min the Airline Operations Supervisor (AOS) also believes there is a mechanical failure. Formally:

$$\forall t \text{at}(\text{Belief}(\text{SS}, \text{Disruption}(\text{mechanical\_failure}, \text{LHB}, \text{CDG})), t) \ \& \\ \exists t' \exists c:\text{MSG\_TYPE} \ t' < t \ \& \\ \text{at}(\text{Communicate\_from\_to}(\text{AE}, \text{SS}, c, \text{Disruption}(\text{mechanical\_failure}, \text{LHB}, \text{CDG})), t') \\ \Rightarrow \exists t'' t'' > t \ \& \ t'' \leq t+10 \ \& \ \text{at}(\text{Observation}(\text{CCo}, \text{Disruption}(\text{mechanical\_failure}, \text{LHB}, \text{CDG})), t'')$$

It is important for this property to hold because under strategy S1, AOS must accept maintenance information content and act on it without challenging the information.

- *Strategy S1—property 2* If SS believes that maintenance information reported to him by the Airport Engineer (AE) is true, then this information should be noticed by the Crew Controller (CCo) within 10 min. Formally:

$$\forall t \forall x:\text{AIRCRAFT} \forall y:\text{AIRPORT} \text{at}(\text{Belief}(\text{SS}, \text{Disruption}(\text{mechanical\_failure}, x, y)), t) \Rightarrow \exists t' t' > t \ \& \ t' \leq t+5 \\ \& \ \text{at}(\text{Belief}(\text{AOS}, \text{Disruption}(\text{mechanical\_failure}, x, y)), t')$$

$$\forall t \forall x:\text{CREW\_PROBLEM} \text{at}(\text{Belief}(\text{CCo}, \text{Crew\_problem}(x)), t) \Rightarrow \exists t' t' > t \ \& \ t' \leq t+2 \ \& \\ \text{at}(\text{Communicate\_from\_to}(\text{CCo}, \text{FC}, \text{ask}, \text{extend\_crew\_hours}), t')$$

It is important that this property holds to ensure proper synchronization between the Aircraft Movement System (used by the SS) the Crew Tracking System (monitored by the CCo).

It is important for this property to hold because under strategy S2, the AOS must challenge information about a maintenance situation, and query the information source.

- *Strategy S2—property 2* if CCo believes there is a crew problem, then, within 2 min, CCo should ask the Flight Crew (FC) to extend their crew duty time. Formally:

It is important for this property to hold because under strategy S2, when the CCo is facing with a crew problem, he must challenge crew limits and seek extensions to crew

duty time, for instance through negotiating with the Flight Crew (FC).

- *Strategy S3—property 1* If SS believes that there is a mechanical failure, then within 5 min, AOS should organize a conference call with AE and Maintenance Watch Engineer to recheck information. Formally:

$$\forall t \forall x: \text{AIRCRAFT} \forall y: \text{AIRPORT} \text{ at(Belief(SS, Disruption(mechanical\_failure, x, y)), t) } \Rightarrow \exists t' t' > t \ \& \ t' \leq t+5 \\ \& \text{ at(Start\_conf\_call(AOS, Disruption(mechanical\_failure, x, y), AE, MWE), t')}$$

This property is checked to ensure that the specialist agents provide solutions to the AOS before he announces offers to solve the problem.

- *Strategy S4—property 2* If AOS announces an integrated disruption management solution, he should obtain within

5 min the vote results (approval/rejection) from both ACo and CCo on the AMS. Formally:

$$\forall t1, t2 \forall s: \text{INTEGRATED\_SOLUTION} \& \text{ at(Communicate\_from\_to(AOS, ACo, inform, integrated\_solution(s)), t1) } \& \text{ at(Communicate\_from\_to(AOS, CCo, inform, integrated\_solution(s)), t2) } \\ \Rightarrow \exists t', t'', t3, t4 t' < t1+5 \ \& \ t'' < t2+5 \ \& \ t3 < t1+5 \ \& \ t4 < t2+5 \ \exists z1, z2: \text{VOTE\_RESULT} \& \\ \text{ at(Communicate\_from\_to(AOS, ACo, reply, vote\_for(s, z1)), t') } \& \text{ at(Communicate\_from\_to(AOS, CCo, reply, vote\_for(s, z2)), t'') } \& \text{ at(Observation(AOS, vote\_for(s, z2)), t3) } \& \text{ at(Observation(AOS, vote\_for(s, z2)), t4)}$$

It is important for this property to hold because under strategy S3, the AOS must seek alternative information and recheck information source and reliability, e.g., through seeking a second opinion from the MWE.

- *Strategy S3—property 2* if MWE believes there is a mechanical failure, then within 5 min, the CCo should notice the aircraft solution on the CTS. Formally:

$$\forall t \forall x: \text{AIRCRAFT} \forall y: \text{AIRPORT} \text{ at(Belief(MWE, Disruption(mechanical\_failure, x, y)), t) } \Rightarrow \exists t' t' > t \ \& \ t' \leq t+5 \\ \& \exists s: \text{AIRCRAFT\_SOLUTION} \text{ at(Observation(CCo, Aircraft\_problem(s))), t')}$$

This property is checked to ensure that the AOS obtain the vote results after he announces a solution to solve the problem. All the identified properties were verified as true for the developed multi-agent system model.

This property is checked to verify a proper synchronization between the CTS (monitored by the CCo) and the AMS (used by the ACo). After hearing the confirmation from MWE in the conference call, the ACo directly reports the aircraft solution through AMS.

- *Strategy S4—property 1* Before announcing an integrated disruption management solution to ACo and CCo, the AOS should have noticed the solutions to the aircraft problem and crew problem reported on the AMS by the ACo and CCo respectively. Formally:

$$\forall t1, t2 \forall x: \text{AIRCRAFT\_PROBLEM} \forall y: \text{CREW\_PROBLEM} \text{ at(Observation(AOS, Aircraft\_problem(x))), t1) } \\ \& \text{ at(Observation(AOS, Crew\_problem(y))), t2) } \& \exists t3, t4 t3 < t1 \ \& \ t4 < t2 \ \& \\ \text{ at(Communicate\_from\_to(AMS, ACo, inform, Aircraft\_problem(x)), t3) } \& \\ \text{ at(Communicate\_from\_to(AMS, CCo, inform, Crew\_problem(y)), t4) } \\ \Rightarrow \exists t', t'' t' > t1 \ \& \ t' > t2 \ \& \ t'' > t1 \ \& \ t'' > t2 \ \exists s: \text{INTEGRATED\_SOLUTION} \& \\ \text{ at(Communicate\_from\_to(AOS, ACo, inform, integrated\_solution(s)), t') } \& \\ \text{ at(Communicate\_from\_to(AOS, CCo, inform, integrated\_solution(s)), t'') }$$

## 4.5 Model evaluation

In this step, each strategy is evaluated in relation to recovery solutions and airline performance. The simulation traces are analyzed, and cost is calculated separately into a spreadsheet.

The four AOC strategies introduced in Sect. 3 have been implemented and simulated in the presented multi-agent system model. For each of these four strategies, various results have been collected such as related to aircraft, crew, passengers, and the minimum time needed to manage the



**Table 6** Simulation results

P	FL	MP	CP	PAX	MDT (min)	OC	PAXC	TL
S1	Cancelled	Fixed	Not resolved	Distressed	26	326kEUR	168kEUR	24
S2	Cancelled	Fixed	Not resolved	Distressed	30	326kEUR	168kEUR	24
S3	Diverted	Fixed	Resolved	Delayed	33	360kEUR	126kEUR	8
S4	Delayed	Fixed	Resolved	Delayed	12	326kEUR	0kEUR	3

*P* Strategy, *FL* flight, *MP* mechanical problem, *CP* crew problem, *PAX* passengers problem, *MDT* minimum disruption management time, *OC* operation costs, *TL* time lost for passengers

disruption. Table 6 presents the simulation results obtained for the four AOC strategies. The table includes two types of costs: (1) the costs for the operator (the airline); and (2) the costs for the users (the passengers). For the airline costs, we used cost data from Air France KLM corresponding to FY 2013 to calculate the airline operating costs; and EU regulations to include passenger compensation rights. For costs incurred by the passengers, we have included the time lost by passengers which has an opportunity cost depending on many factors such as travel motive, passenger characteristics, etc. See [42].

The outcome of strategy S3 concurs with the best solution identified by the expert panel. However, the outcomes of S1 and S2 are significantly worse, and the outcome of S4 even outperforms the expert panel result. To understand the background of these differences, the simulation results have carefully been analyzed.

Under strategies S1 and S2, AOC operators make decisions based on limited coordination, as a result of which the disruption considered is not efficiently managed. The aircraft mechanical problem was eventually fixed, however, the flight was cancelled. As a result, the 420 passengers were accommodated in hotels (i.e. greatly inconvenienced). This unfavorable outcome can be explained as a result of the possible actions identified by the crew controller i.e. “await crew from inbound aircraft” and “seek extensions to crew duty time.” Crew controllers mainly considered crew sign-on time and duty time limitations and tried to work within these constraints. In this scenario, none of the possible actions solves the crew problem.

Under strategy S3, AOC controllers consider complex crewing alternatives such as flying a replacement crew from another airport. and can either choose to deadhead replacement crew from another airport or use crew from other aircraft. Therefore, under S3 the decision was made to reroute the flight via BOM and fly-in a replacement crew from PCF into BOM. Here, both the delayed crew and replacement crew were able to operate in one tour of crew duty time. This solution was chosen instead of using crew from other aircraft based on the transcript data from the expert panel simulations in [32]. In comparison to strategies S1 and S2, strategy S3 is much better from both the airline and the passenger’s perspectives. Regarding the minimum time required

for managing the disruption strategy, S3 takes more time than S1 and S2.

Under strategy S4, it was assumed that AOC agents make level 3 decisions similar to S3. Under S3, the crew controller agent can either consider various crew deadheading possibilities or user alternative crew from other aircraft. If the latter strategy is followed, strategy S4 is able to identify a possibility that had not been identified by the expert panel. The flight crew that had landed the aircraft at CDG had received sufficient rest to fly the delayed aircraft directly to PCF instead of enjoying their scheduled day-off in Paris. Passengers had a minimum delay compared to the previous strategies (S1–S3) as they only had to wait for the aircraft to be fixed. If the assumption regarding AOC agents under strategy S4 was changed to decision level 1 or 2 similar to S1 and S2, the crew problem would not have been resolved (Table 6).

In a previous work by the authors [43], the same scenario was evaluated for a different multi-agent system called MASDIMA [44]. The details of this evaluation are described in [45]. Both MASDIMA and Strategy S4 lead to a similar outcome for the aircraft, crew, and passenger problem, with differences in execution time ascribed to different temporal assumptions taken by Müller [45] and Bouarfa [46] e.g. regarding the time it takes an engineer to contact the station supervisor, and time required to input information in the aircraft movement system. Although both approaches use different negotiation protocols (e.g. The specialist agents use the Generic Q-negotiation protocol and can negotiate among each other unlike strategy S4), they both lead to the same solution. This can be explained by the small solution space for the considered scenario. MASDIMA’s crew controller agent solution space also includes exchange crew from another aircraft which is equivalent to level 3 decision-making. Similar to S4, The application of MASDIMA to the benchmark scenario yields as solution to delay flight 705 to fix the aircraft mechanical problem, to replace the crew from flight 705 by the crew from inbound flight 706, and to keep passengers on the same flight. It is assumed that the human supervisor accepts this solution proposal.

## 5 Conclusion

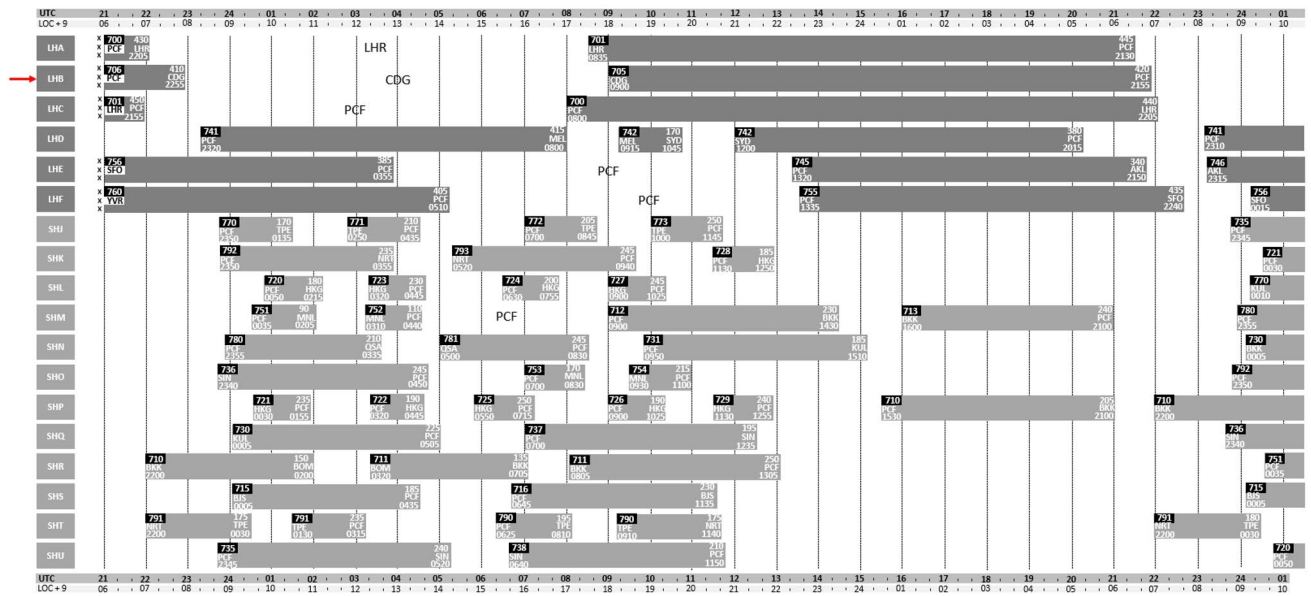
Efficient handling of disruptions by airlines requires advanced coordination and communication means employed by socio-technical teams, in which human operators are supported by intelligent technology. In this paper, we investigated four different airline disruption management strategies based on multi-agent coordination and negotiation models. The strategies varied in the level of performance in terms of the decision-making levels and coordination capabilities of the involved agents. The effects of the strategies were studied by simulation in the context of a realistic scenario involving a mechanical failure disruption. The performance metrics included the costs of each solution, associated delay, and execution time to manage the disruption. The results demonstrated that the effectiveness and efficiency of the strategies were in direct relation to the capabilities of the agents. For instance, under strategy S4, when the specialist agents make level 3 decisions (see Appendix 2), they are able to identify possibilities to effectively manage the disruption. Another important contribution of the paper is the formal specification of the strategies in a multi-agent system model using LEADSTO and TTL languages, which enabled simulation and automated verification. Using TTL Checker, a set of formalized TTL properties was verified on the model simulation traces, which were required to hold for the operational scenario under consideration. Based on the obtained results, we can conclude that the proposed approach could be a promising way forward for modelling, designing, and evaluating multi-agent systems for handling disruptions by socio-technical teams in the air transportation system.

In future work, we will explore probabilistic human models to represent uncertainties in strategic and tactical

decision-making of AOC operators. One promising approach is Bayesian Belief Networks which could be well used to model human reasoning and theory of mind models [47]. Investigating how to incorporate such models in AOC applications could lead to promising results. Furthermore, to formally verify probabilistic models, advanced probabilistic model checking tools such as PRISM [48] needs to be explored. Another possible extension is to consider more sophisticated decision models such as “Markov decision processes”. Agents can learn what the best vote is based on their previous experience and feedback received. In the proposed approach, all specialist agents provide their potential solutions to the AOC supervisor in advance. As human decision makers can generate new solutions on the fly based on the discussions with other specialists, it might be an interesting challenge to design such agents actively generating new solutions during their negotiation. In such a case, the proposed negotiation protocol should be adapted accordingly. Finally, the proposed coordination and negotiation strategies can be applied and evaluated in other operational scenarios, including ones with cascading disturbances. The properties of the strategies and the associated coordination protocols will need to be analyzed more extensively for their efficiency and robustness.

## Appendix 1: Printout of the screen image of airline controller at the time of disruption

See Fig. 8.



**Fig. 8** Printout of the screen image at the time of disruption 06:55 Coordinated Universal Time (see top horizontal UTC time-scale). A secondary horizontal time-scale showed local time (UTC+9 h). The horizontal blocks (called pukus) represent the flights and include relevant information such as the flight number, actual passenger loading, departure and arrival airport, and departure and arrival time. The background shading of each flight block was designed to represent a

type of aircraft (a darker block represents a large aircraft and a light block represents a medium sized aircraft). The longer the flight duration, the larger the size of the block. The vertical axis on the left side shows the aircraft registrations that identify each aircraft in the fleet. In this scenario the aircraft with the mechanical problem is designated by registration code LHB ‘Lima Hotel Bravo’ to the left of the second row highlighted by the arrow

## Appendix 2: Differences between strategies S1–S3 in relation to various aspects

See Table 7.

**Table 7** Strategies S1–S3

Strategy	Rules corresponding to strategies S1–S3	ID
<b>Information</b>		
Strategy S1	Accept information source and content and act on information given	P1a
Strategy S2	Challenge/ query information	P2a
Strategy S3	Seek alternative information and recheck source and reliability	P3a
<b>Schedule</b>		
Strategy S1	Identify late aircraft pattern	P1b
	Seek first available Aircraft	P1c
	Change aircraft within type to limit delay	P2b
Strategy S2	Request high-speed cruise	P2c
	Identify curfews and work within them	P2d
	Identify curfews and work within them	P2d
Strategy S3	Reroute flights to avoid problems (e.g. curfews)	P3b
	Combine flights to free up aircraft	P3c
	Seek curfew dispensation	P3d
<b>Weather</b>		
Strategy S1	Accept basic weather information	P1d
	Consider impact of weather at project times	P1e

**Table 7** continued

Strategy	Rules corresponding to strategies S1–S3	ID
Strategy S2	Assess maximum holding times	P2e
	Seek additional fuel	P2f
	Consider consequences of diversions, cancellations, etc	P2g
Strategy S3	Look for weather trends	P3e
	Consider handling/congestion at diversion ports	P3f
	Hold on ground at the departure port	P3g
Maintenance		
Strategy S1	Identify maintenance requirements	P1f
	Accept information about a maintenance situation	P1g
Strategy S2	Change utilisation to account for maintenance limitations	P2h
	Challenge/Check information regarding maintenance situation	P2i
Strategy S3	Negotiate alternative maintenance requirements and limitations	P3h
	Identify possible maintenance outcomes (e.g. earlier serviceability than advised)	P3i
Crewing		
Strategy S1	Await crew from inbound aircraft	P1h
	Check crew sign on and duty time	P1i
Strategy S2	Challenge crew limits	P2j
	Seek extensions to crew duty time	P2k
Strategy S3	Seek alternative crew (e.g. from the nearby base or another aircraft)	P3j
Tranships		
Strategy S1	Delay subsequent flights to wait for tranships from later aircraft	P1j
Strategy S2	Challenge tranship times at the airport	P2l
Strategy S3	Seek alternative uplifts to prevent tranship problem	P3k

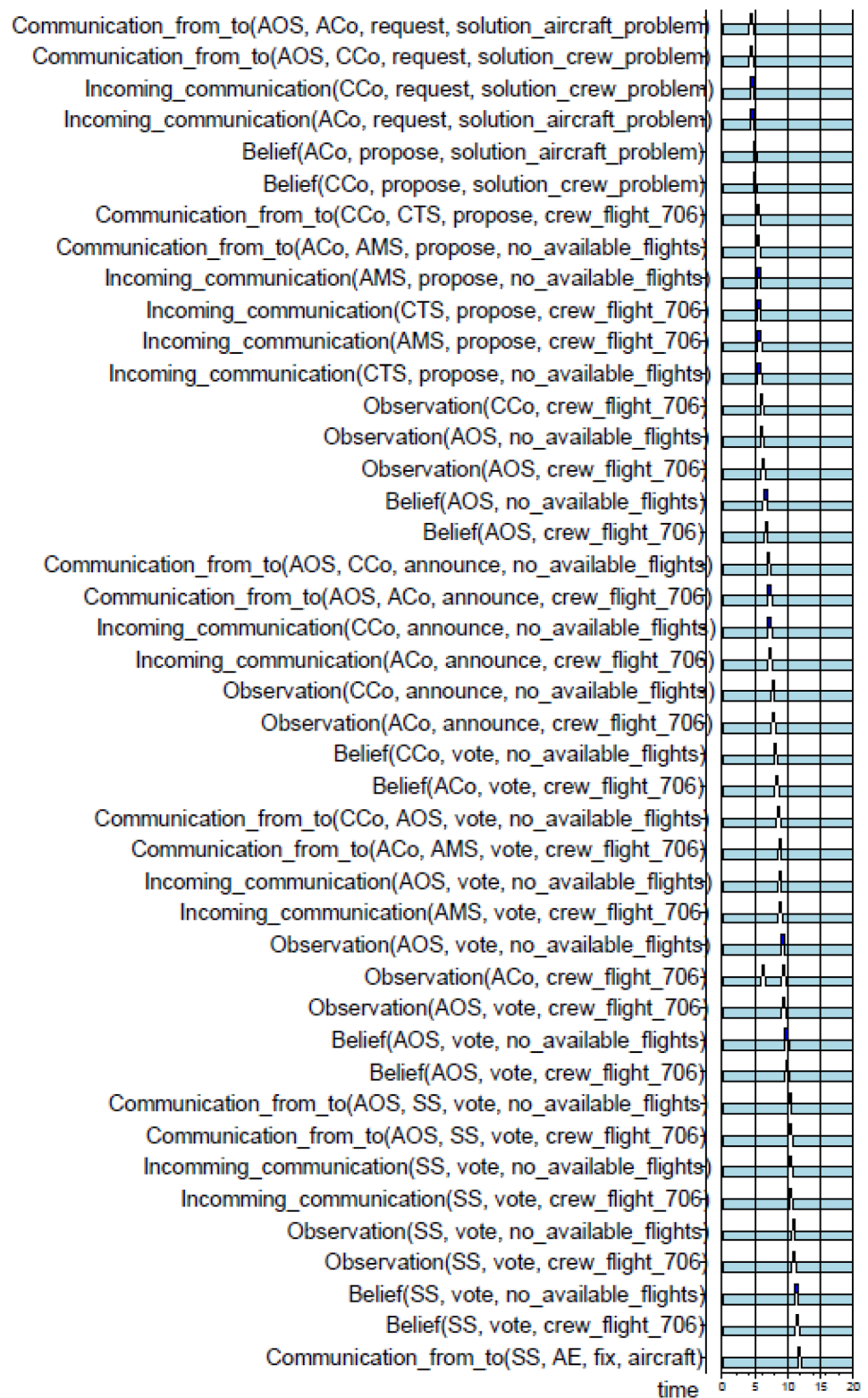
### Appendix 3: Simulation traces corresponding to strategy S4

See Fig. 9.

**Fig. 9** Simulation traces strategy S4



Fig. 9 (continued)



## References

- Bouarfa S, Blom HAP, Curran R, Everdij MHC (2015) Agent-based modelling and simulation of emergent behavior in air transportation. Springer, Berlin
- Blom, H.A.P., Bouarfa, S. (2016). Resilience. In Complexity Science in Air Traffic Management. A. Cook and D. Rivas (Eds.), Routledge, England.
- Marsa-Maestrea I, Klein M, Jonker CM, Aydoğan R (2014) From problems to protocols: towards a negotiation handbook. *Decis Support Syst Electron Commer* 60:39–54
- Baarslag T, Gerding E, Aydoğan R, Schraefel MC (2015) Optimal negotiation decision functions in time-sensitive domains. In: IEEE/WIC/ACM International joint conferences on web intelligence (WI) and intelligent agent technologies (IAT), pp 190–197.
- Baarslag T, Hendriks MJC, Hindriks KV, Jonker CM (2016) A survey of opponent modeling techniques in automated negotiation. In: AAMAS2016: 2016 international conference on autonomous agents and multi-agent systems, Singapore, pp 575–576
- Castro AJM (2013) A distributed approach to integrated and dynamic disruption management in airline operations control. PhD thesis, University of Porto.
- Chen Y, Peng Y, Finin T, Labrou Y, Cost R, Chu B, Yao J, Sun R (1999) A negotiation-based multi-agent system for supply chain management. In: Proceedings of Agents'99 workshop agent-based decision-support for managing internet-enabled supply chain, pp 15–23.
- Nguyen D, Jennings N (2005) Managing commitments in multiple concurrent negotiations. *Electron Commer Res Appl* 4:362–376
- Wang G, Wong T, Wang X (2013) An ontology-based approach to organize multi-agent assisted supply chain negotiations. *Comput Ind Eng* 65:2–15
- Cui-hong Hu (2011) Automated negotiation model of supply chain management based on multi-agent. *MSIE* 2011:178–180
- Rady HA, El-Shorouk (2011) Multi-agent system for negotiation in a collaborative supply chain management. *Int J Video Image Process Netw Secur IJVIPNS-IJENS* 11(5)
- Xue X, Lu J, Wang Y, Shen Q (2007) Towards an agent-based negotiation platform for cooperative decision-making in construction supply chain. *KES international symposium on agent and multi-agent systems: technologies and applications*, 416–425
- Aydoğan R, Festen D, Hindriks KV, Jonker CM (2017) Alternating offers protocols for multilateral negotiation. *Modern approaches to agent-based complex automated negotiation*. Springer, Cham, pp 153–167
- Klein M, Faratin P, Sayama H, Bar-Yam Y (2003) Protocols for negotiating complex contracts. *IEEE Intell Syst* 18(6):32–38
- Aydoğan R, Hindriks KV, Jonker CM (2014) Multilateral mediated negotiation protocols with feedback. *Novel insights in agent-based complex automated negotiation*. Springer, Tokyo, pp 43–59
- Sanchez-Anguix V, Aydoğan R, Julian V, Jonker CM (2014) Unanimously acceptable agreements for negotiation teams in unpredictable domains. *Electron Commer Res Appl* 13(4):243–265
- Ágotnes T, Bulling N (2014) Formal methods for coordinating multi-agent systems (Dagstuhl Seminar 14332). *Dagstuhl Rep* 4(8):21–44
- Bordini RH, Fisher M, Visser W, Wooldridge M (2006) Verifying multi-agent programs by model checking. *Auton Agent Multi Agent Syst* 12(2):239–256
- Peters PF (2009) Airborne on time. In: Cwerner S, Kesselring S, Urry J (eds) *Aeromobilities*. Routledge, London, pp 159–176
- Grandeau S (1995) The processes of airline operational control. Msc. thesis, Massachusetts Institute of Technology, Boston.
- Clausen J, Larsen A, Larsen J, Rezanova NJ (2010) Disruption management in the airline industry—concepts, models and methods. *Comput Oper Res* 37(5):809–821
- Kohl N, Larsen A, Larsen J, Ross A, Tiourine S (2007) Airline disruption management—perspectives, experiences, and outlook. *J Air Transp Manag* 13(3):149–162
- Clarke MDD (1998) Irregular airline operations: a review of the state-of-the-practice in airline operations control centers. *J Air Transp Manag* 4:67–76
- Barnhart C (2009) The global airline industry, chapter irregular operations. Wiley, West Sussex, pp 253–274
- Pujet N, Feron E (1998) Modelling an airline operations control. In: Presented at the 2nd USA/Europe Air Traffic Management R&D Seminar. [http://atmseminar.org/seminarContent/seminar2/papers/p\\_034\\_APMMA.pdf](http://atmseminar.org/seminarContent/seminar2/papers/p_034_APMMA.pdf). Accessed 25 Jan 2021
- Castro AJM, Oliveira E (2011) A new concept for disruption management in air-line operations control. *Proc Instit Mech Eng Part G J Aerosp Eng* 225(3):269–290
- Bruce PJ (2011) Understanding decision-making processes in airline operations control. Ashgate Publishing Company, Farnham
- Bouarfa S, Blom HAP, Curran R (2016) Agent-based modelling and simulation of coordination by airline operations control. *IEEE Trans Emerg Top Comput*. <https://doi.org/10.1109/TETC.2015.2439633>
- Castro AJM (2008) Centros de controlo operacional: Organizacao e ferramentas: monograph for post-graduation. Instituto Superior de Educao e Ciências, Lisbon
- Machado N (2010) Impact of the organizational structure on operations management. Msc thesis, University of Porto.
- Feigh KM (2008) Design of cognitive work support systems for airline operations. PhD thesis, Georgia Institute of Technology.
- Bruce PJ (2011) Decision-making in airline operations: the importance of identifying decision considerations. *Intern J Aviat Manag* 1:89–104
- Visser W, Aydoğan R, Hindriks K, Jonker C (2012) A framework for qualitative multi-criteria preferences. In: Proceedings of the forth international conference on agents and artificial intelligence (ICAART 2012), Portugal, pp 243–248
- Nikolic I, Ghorbani A (2011) A method for developing agent-based models of socio-technical systems. 2011 international conference on networking, sensing and control, 44–49
- Sharpanskykh A, Treur J (2010) A temporal trace language for formal modelling and analysis of agent systems. *Specification and verification of multi-agent systems*. Springer, Boston, pp 317–352
- LEADSTO software (2019). <http://www.cs.vu.nl/~wai/TTL/>. Accessed 25 Jan 2021
- Bosse T, van der Jonker CM, Meij L, Treur J (2007) A language and environment for analysis of dynamics by simulation. *Int J Artif Intell Tools* 16(03):435–464
- van den Broek EL, Jonker CM, Sharpanskykh A, Treur J, Yolum P (2006) Formal modelling and analysis of organizations. In: Boissier O, Padget J, Dignum V, Lindermann G, Matson E, Ossowski S, Sichman JS, Salceda JV (eds) *Coordination, organizations, institutions, and norms in multi-agent systems*, vol 3913. Springer, Heidelberg, pp 18–34
- Sharpanskykh A, Treur J (2006) Modeling of agent behavior using behavioral specifications. Vrije Universiteit Amsterdam. The Netherlands. Technical Report 06-02ASRAI
- Manzano M (1996) Extensions of first order logic, vol 19. Cambridge University Press
- Bosse T, Jonker CM, Van der Meij L, Sharpanskykh A, Treur J (2009) Specification and verification of dynamics in agent models. *Int J Coop Inf Syst* 18(01):167–193

42. Institut du transport aerien (2000) Costs of air transport delay in Europe. <https://www.eurocontrol.int/sites/default/files/content/documents/single-sky/pru/publications/other/cost-of-air-transport-delay-in-eu-ita.pdf>. Accessed 25 Jan 2021
43. Bouarfa S, Muller J, Blom H (2018) Evaluation of a multi-agent system approach to airline disruption management. *J Air Transp Manag* 71:108–118
44. Castro AJM (2013) A distributed approach to integrated and dynamic disruption management in airline operations control. PhD thesis. University of Porto.
45. Müller J (2015) Evaluating agent-based automation in airline operations control. MSc Thesis. Delft University of Technology.
46. Bouarfa S (2015) Agent-based modelling and simulation of safety and resilience in air transportation. PhD thesis, Delft University of Technology, ISBN/EAN 978-94-6259-924-6, Ipskamp Drukers Publisher.
47. Neapolitan RE (2004) Learning bayesian networks, vol 38. Pearson Prentice Hall, Upper Saddle River
48. Kwiatkowska M, Norman G, Parker D (2017) Probabilistic model checking: advances and applications. In: Drechsler R (ed) Formal system verification. Springer, Berlin, pp 73–121