A LOGICAL ARCHITECTURE FOR DECISION-MAKING SUPPORT IN AIR TRAFFIC MANAGEMENT

Ítalo Romani de Oliveira[1] (italo.oliveira@poli.usp.br)

Paulo Sérgio Cugnasca[1] (paulo.cugnasca@poli.usp.br)

Fábio Aparecido Gamarra Lubacheski[1] (fabio.Lubacheski@poli.usp.br)

Liria Matsumoto Sato[1] (liria.sato@poli.usp.br)

Mário Aparecido Correa[1] (mario.correa@poli.usp.br)

João Batista Camargo Júnior[1] (joao.camargo@poli.usp.br)

Eno Siewerdt[2] (eno@atech.br)

[1] Department of Computer Engineering and Digital Systems
Polytechnic School of University of São Paulo
[2] Atech Tecnologias Críticas

Abstract

In the current scenario of growing demand concerning Air Traffic Control Sectors and airports, decisions about air traffic management (ATM) and individual aircraft control are based on information provided by messages of several specific computer applications. A better general performance in exploiting the air transportation resources is achieved when information, from all applications, is available in order to take a single decision, in collaborative operations, and decisions are made public for all involved participants. This way of making decisions can be more efficient and stable when supported by safe, high performance and highly available computer networks, message handlers, and databases. This study proposes the multi-agent approach for decision-making, where agents use a proper message system and distributed database architecture. Finally, a case study is presented, in which we evaluate fuel savings that may be provided by a collaborative decision-making approach.

1. INTRODUCTION

Efficiency in air traffic operations requires quickness in making decisions, even when resources, such as runways, airspace and air traffic control working positions are close to their capacity limits. Once the traffic increases, following the world economic growth, its operation reaches increasingly its saturation. In this condition, air transportation system efficiency tends to decrease, mainly due to the occurrence of (AATT, 1997):

- Flight cancelling;
- Congestion-due delays;
- Infeasibility to obtain desired route;
- Infeasibility to obtain desired altitude;
- Infeasibility to obtain desired speed.

These items are consequences of typical airport and airspace capacity limitations, or of weather changes, and raise high costs. The air traffic flow planning and the weather forecast work to minimize these occurrences, but the presence of unforeseen disturbances may end up vanishing a considerable part of this work. When resources are saturated, the possibilities of compensating the disturbances will diminish and, therefore, the remaining solutions must be more clearly seen and more easily executable. Many decisions about waiting or beginning flights, altering routes, altitudes and speeds are safety critical, and therefore may be supported by safe and real time systems. Considering that, in more advanced systems, many of these decisions will be automatically proposed by computer systems to human operators, the need of applying safe and fair negotiation rules among the participants of the air traffic scenario is then clear. This study proposes the modeling of the interaction between air traffic control systems, using safe and reliable databases.

2. AIR TRAFFIC MANAGEMENT ORGANIZATION

Air traffic is promoted by air transporters, offering flights to link distinct places. Air transporters may be divided into several classes, according to their goals. However, as the greatest part of air traffic is generated by regular airlines, we will refer to an airline as an air transporter.

Several airlines share the same primary resources for air transportation, such as airspace and airports. The simultaneity, in the use of these resources by airlines, brings the need of a broad Air Traffic Management intelligent system.

Generally the Air Traffic Management is understood as a combination of strategic planning and tactical tasks. The flight strategic planning is attributed to the so-called Air Traffic Flow Management (ATFM), which has the task of distributing spatially

and chronologically the flights requested by the airlines. This task is conducted in a time period, varying from some months, days and hours before a flight operation, and must result in a safely manageable airspace occupation density. On the other hand, the tactical tasks are executed regarding the strategic planning adjustment and implementation, based on updated factual information. Given this distinction, the strategic ATFM is considered the complimentary task of the Tactical Air Traffic Control (ATC), which is provided between the hours or just moments before the flight, as well as during all flight long, until the aircraft is blocked in its destination airport.

Commonly, the ATFM tasks are executed for a great geographical area, which may be composed of a group of nations. The ATC is provided in the same geographical area, but hierarchically divided in several Control Centers, with sectors and sub-sectors. ATFM and ATC are integrated in a computer system, generically called ATMS, i.e. ATM System.

The ATMS is required to provide services for the groups of the so-called FIR (Flight Information Regions), and for the TMA (Terminal Maneuvering Areas), and each FIR and TMA is divided into sectors (see fig. 1). Each sector corresponds to a control working position, with at least one human operator. In many cases, the automation tools help to optimize the usage of air traffic resources. To understand the ATMS, it is necessary to explain the sequence of a flight.

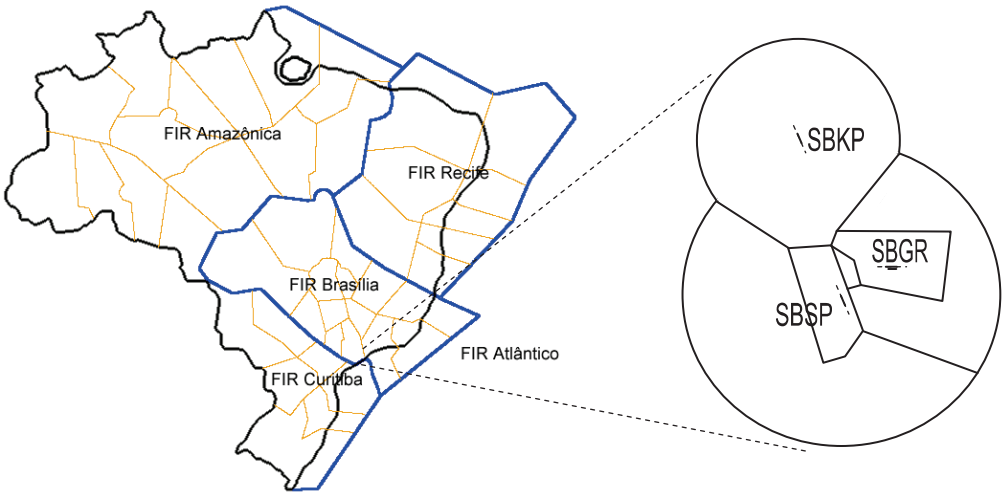


Fig. 1: The geographical division of the Brazilian ATMS (on the left), and the São Paulo TMA (on the right). Each FIR and TMA is divided into sectors.

2.1 Sequence of a flight

A flight begins at an airport. An airport is an entity that has a considerable complexity itself and, because of this, requires at least one dedicated control facility. This facility needs to monitor and assist the aircraft final approach, its landing, ground movements and take-off. When the aircraft is at the airport, ready to depart, it needs to receive information of clearance from the airport control. The airport may be subordinated directly to the FIR control facility, or in some cases, to a control facility called APP (Approach Control), when the airport's airspace imposes more workload to the Air Traffic Controllers (ATCo). In this case, the airport management task is extended to the airspace, of some tenths of miles around the airport, and divided between the APP and the airport control, more commonly known as tower. In addition, there may be several airports subordinated to only one APP, as indicated in the right side of Fig. 1. In this illustration, SBKP, SBSP and SBGR are airports. The airspace controlled by an APP is called TMA (Terminal Maneuvering Area).

Continuing the sequence of a flight, in the moment that an aircraft takes off, it will be under an APP control. More specifically, an APP divides its work, inside the TMA, into sectors, and each sector is under control though one control working position. Next to the moment of crossing the TMA border, the aircraft will be transferred to a control facility called ACC (Area Control Center). The ACC has one operator or one pair of operators for each sector, and the aircraft will pass successively from sector to sector, until it reaches a distance close to its destination airport. Then, the aircraft reaches the destination TMA and the control transference occurs in the opposite way of that happening on its departure transference.

Given this generic flight procedure, the air traffic control systems are typically hierarchically organized, which is represented in the diagram of Fig. 2. In this figure, only the ATC System (ATCS) is represented, since it serves as a structure for the ATM systems. Besides the organizations mentioned in Fig. 2, there are other participants in the system, such as the AOC (Airline Operations Center), which coordinates the operations of one or more airlines.

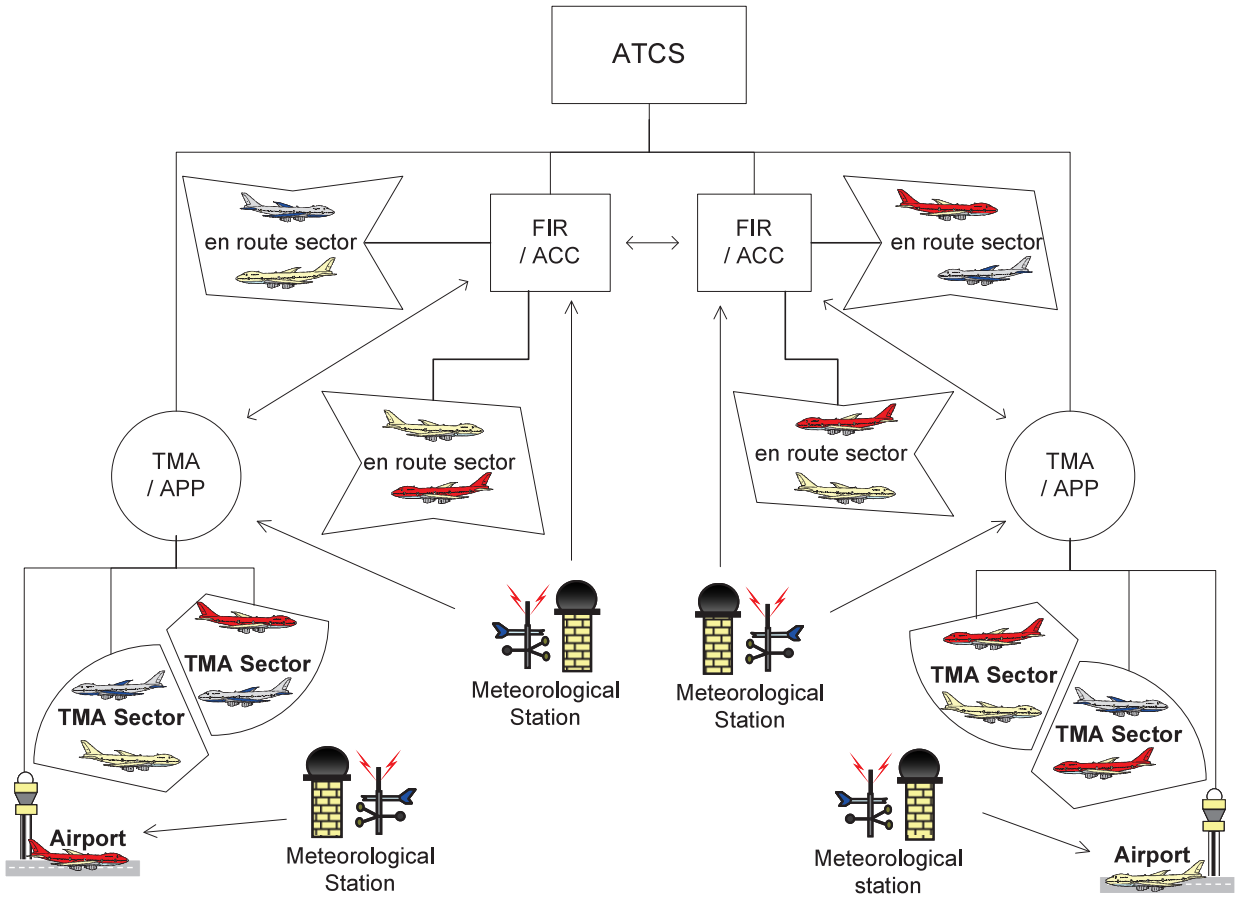


Fig. 2: An Air Traffic Control System Organization.

Observe in Fig. 2 the presence of meteorological stations. Airborne meteorological sensors, such as radars or even satellites, may also play this role. They provide valuable information concerning meteorological conditions, which may include alerts of dangerous zones or 3D atmosphere data.

2.2. Information exchange in the ATM System

The information, which is being exchanged between the sectors of Fig. 2, may be ranked in the following categories:

- Information about past, current and future aircraft state;
- Aircraft control actions;
- Information about the coordination and occupancy of the airport;
- Meteorological information.

This information may come from two types of sources: one from human operators, which manually input information in the system, and the other from sensors, of a great variety.

Some examples are:

- PSR – Primary Surveillance Radar;
- SSR – Secondary Surveillance Radar;
- ADS – Automatic Dependant Surveillance, and its broadcast version called ADS-B;
- SMR – Surface Movement Radar, integrated to the ASDE (Airport Surface Detection Equipment);
- MLT –Multilateration Equipment;
- Radars and meteorological satellites;
- Other meteorological devices.

All this sensor data is available to the ATC systems, by means of specific networks, and may travel as raw data or identified tracks. The tracking systems (BLOEM, BLOM, VAN SCHAIK, 2002) are needed when executing surveillance ATC tasks, and may use advanced algorithms in order to perform their functions. Nevertheless, whichever are the chosen algorithms, tracking is only safe and reliable if it uses safe and reliable databases.

Once available and used in the right manner, all information enumerated in this section is essential to the correct system management. Providing that high distribution level and real time requirements, over the information availability and processing, are given, it is essential to implement adequate decision-making techniques in order to obtain safe and efficient the operation performances.

2.3. ATM Systems Requirements

ATM systems need real-time-information flow and processing, since air navigation has its own synchronism and dynamics, on which human life and goods integrity will depend on. The real-time-information flow may be obtained by high safety and availability properties in telecommunication equipment, including data links and routers.

Besides telecommunications and data-processing infrastructure, user applications are needed to allow acting, aware of the proper information. User actions must be limited by some coordination rules and, once initiated, these actions must be immediately divulged among the other users under their influence or, at least, they should be accessible to all users, within the involved area. These working properties make the ATM System an integrated management system, which needs the Collaborative Decision Making (CDM, 2004).

The vision of integrated information management must guide the CDM (Collaborative Decision Making) application designers, considering essential the requirements of generation and distribution of ATM information. Such requirements include

(CORREA, CAMARGO, SIEWERDT, 2004):

- *Precision and Accuracy*: information must be precise and/or exact enough, and the user must be informed about the safety levels or constraints for validating the information.
- *Stability*: information must be stable enough and not be subject to spurious oscillations. In many cases, even valid information must be screened.
- *Coverage*: information must be complete enough to include all relevant elements to a given situation.
- *Opportunity*: information must be available at the right moment of needing it.
- *Coherence and Consistency*: there must be no conflict of information, and the information must make sense to the user. Since the information is the decision support, there must be information update synchronization, allowing coherence in the decision-making process.
- *Security and Reliability*: non-authorized interventions in the system have to be avoided. Classified information, mainly in special operations, must be protected.
- *Availability*: the system availability level must be compatible with the application environment, with respect to the existence of other complimentary systems, avoiding the risk of entering in contingent states, etc. Low availability may generate safety problems, because traffic inertia cannot be suppressed at any sudden need.

3. THE MULTI-AGENT APPROACH

Jennings (1996) showed that, due to the characteristics of Multi-Agent Systems (MAS), there are significant advantages over the monolithic problem solvers, in many situations. These characteristics are:

- Parallelism, which may provide a faster problem-solving speed;
- Less data traffic, making partial high-level solutions to circulate, instead of raw data;
- More flexibility, since in an MAS there are agents with different abilities interacting to solve the same problem;
- Higher availability, allowing correctly working agents to take over the tasks of faulty agents.

As defined by Wooldridge (1999), a computer system is an agent in this study, placed in a given environment, and able to execute autonomous actions in this environment, trying to reach the proposed goals. Wooldridge and Jennings (1995) presented an agent as being an entity with intrinsic

problem-solving abilities.

The features of Multi Agent Systems Multi Agent Systems make them adequate, at least from the theoretical point of view, to this application field do its nature of distributing air traffic, as well as for its complexity. The goals of this research study include tasks of defining basic agent properties, methods and communication schemes, in such a way that the introduction of a new automation tool, or recent modifications in existing tools, or even the absence of some of them, do not require structural modifications in the system, not even in its internal interaction rules (OLIVEIRA, 2004). The missing functions in malfunctioning tools may be performed, at least partially, by the interaction of non-specialized agents.

3.1. Agents Interaction

A group of specialized agents was defined in order to identify common characteristics of air traffic agents. This is not a closed group, but represents the most significant agents to the decision-making process in ATMs. Agents described in figures 3 and 4 compose this group.

Agents in Fig. 3 and Fig. 4 were explained in section 2, except for the Router and the Synchronizer. The Router is basically a server to find connections among agents, through the available networks. The synchronizer agent provides the universal time reference, and the synchronization in the data flow. Observe in Fig. 4 that all the agents, except for the Router and the Synchronizer, are part of the ATCS.

Besides the agents, Fig. 4 shows other elements as:

- Meteorological data sources (Meteo), which provides weather data;
- AMS (Agent Management System) – controls the entrance and the exit of agents in the system, and provides references to the reciprocal awareness of the agents.
- DDBN (Distributed Data Base Node) – it is one or more nodes of the distributed database system, which together, form a body with unified behavior, which is explained in section 4. They store the information described in section 2.2.

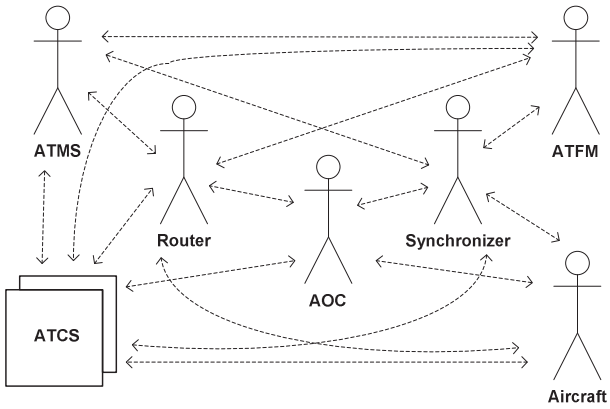


Fig. 3: ATM agents and its communication lines.

The communication between agents is performed by means of messages, from which message handling, described in the next subsection, were developed.

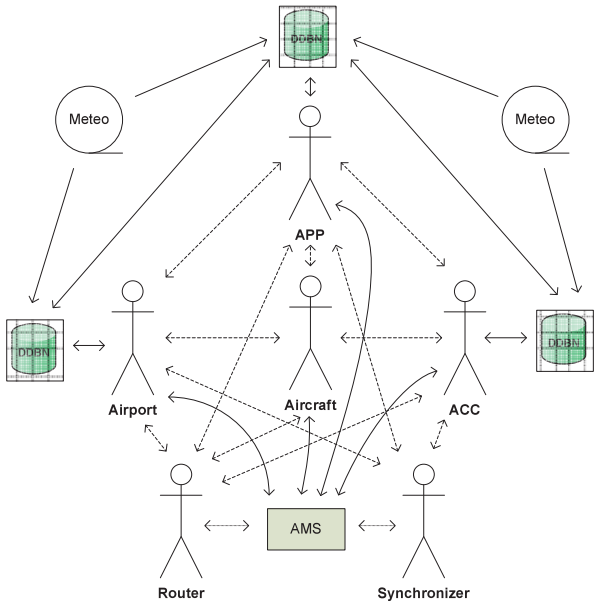


Fig. 4: Detailed interaction diagram of the ATCS agents, including data bases, data sources and communication elements.

3.2. Message handling components

Aiming to introduce a whole set of automation algorithms in the proposed Multi Agent environment, the agents need to use components for safe and reliable message handling. The inter-agent communication occurs through messages, and the central idea in these message components is to provide common software objects. These are exhaustively tested and improved for the use of many agent classes, in several different situations, and will cover all the logic required to provide the agents, by:

- Establishing connection between other agents;
- Verifying the message origin and eventually its authenticity;
- Dispatching the actions required by each message;
- Encapsulating the participation of legacy systems;
- Precisely determining, along the time, the sequence of the received messages;
- Treating exceptions, catching the ones occurring in the message exchange process, and dispatching the corresponding handling method.

The messages, manipulated by these components, are compatible with the ICAO (International Civil Aviation Organization) standards, such as ADS, ADS-B and CPDLC (ICAO, 1999).

The scheme of message components is illustrated in the diagram of Fig. 5. The elements in this diagram are:

- Agent: the generic component that encapsulates the decision logic of each agent, presented in the subsection 3.1.
- *ConnectionServer*: the component that waits for connection requests and is able to verify, for each request, its security credentials. In case the connection is not allowed, it creates an *AgentConnection* in

the connection table. This task is executed with the help of the *ConnectionListener* thread. Another task of the connection server is to encapsulate agent requests to send messages, a task done through the *MessageManager*.

- *MessageManager*: dispatches the messages requested by one agent, to other agents, through the *AgentConnection* and its respective *OutputQueue*. Also process the messages in the *InputQueue* and call agent methods, when necessary, to each one of the messages.

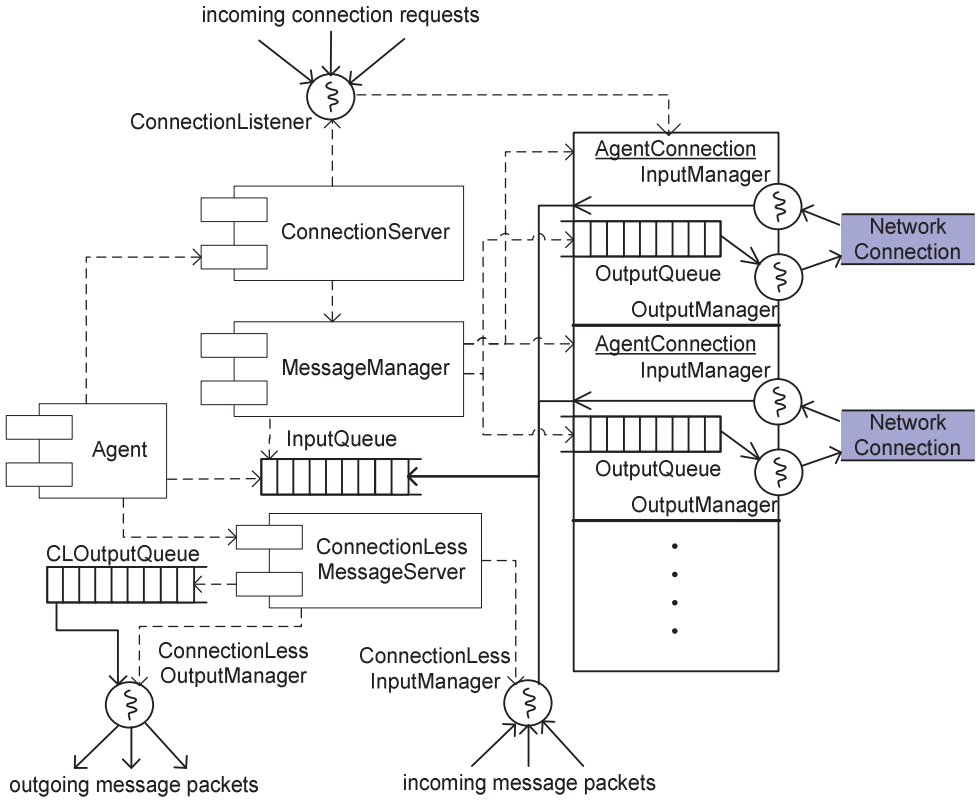


Fig. 5: Software components to exchange messages among agents.

- *AgentConnection*: object of *InputManager* and *OutputManager* threads, which control the message transmission and reception through a network connection with other agents. This task is executed using the *InputQueue* and *OutputQueue* objects. Each *AgentConnection* corresponds to a different agent, which is connected with the agent itself.
- *ConnectionLessMessageServer*: controls the message transmission and reception, where protocols do not require a network connection (unique route, packet sequence, etc.). These tasks are executed with the help of the *Connection-*

LessOutputManager and *ConnectionLess-InputManager* threads, and also of a *CLOutputQueue* object.

The next section presents the data base system proposed for the agents, which store and retrieve ATM data.

4. DISTRIBUTED DATABASE ARCHITECTURE

The distributed database here proposed, for storing and retrieving ATM data, is based on a distributed parallel file system called NPFS (Guardia, 1999). This file system allows breaking the processing of queries in parallel sub-queries, increasing performance. Queries can be decomposed and distributed among several processors. Therefore, this database system provides scalability, since the storage capacity follows, linearly, the increase of storage resources. Reliability is another quality of this system; its data distribution, among several nodes, makes database more robust avoiding data loss.

4.1. Network Parallel File System

Network Parallel File System – NPFS – is a file system for distributed memory architecture, composed by a set of nodes, interconnected in a high-speed network. Each node has its own disk and local memory, and works both for data storage and process execution. The processes are divided in three types: master, server, and client, which interact with each other.

The Master has a unique instance, and its location is determined in the system activation. It is responsible for generating centralized control operations, as opening and closing files. The Server process is placed in each node, and its function is to perform reading and writing operations in the parallel file segments. It is also responsible for the management of connection with application processes.

The Client is an application process that accesses parallel files, performing calls to the NPFS primitive functions. The reading requests are serially sent from the client to the implicated servers. They will simultaneously read the data and serially send data back to the clients. In writing operations, data is serially sent from the client to the appropriate servers.

This infrastructure high availability is obtained by means of an assurance mechanism for consistency, able to restore lost data in individual nodes. For this purpose, when a data stripe of n blocks is being written in the file system, this stripe will be composed by n – 1 data blocks, and one parity block. The parity block position is rotated through the nodes, for each data stripe and, therefore, parity blocks are circularly

distributed along the nodes, as illustrated in Fig. 6.

NPFS master process can detect the unavailability of some node, by getting the node list of some file, and comparing with the list of nodes with active servers. The lost or unavailable data can be re-calculated and restored by the parity data blocks.

The NPFS files are the most basic infrastructure to the Distributed Database work, described in the next subsection.

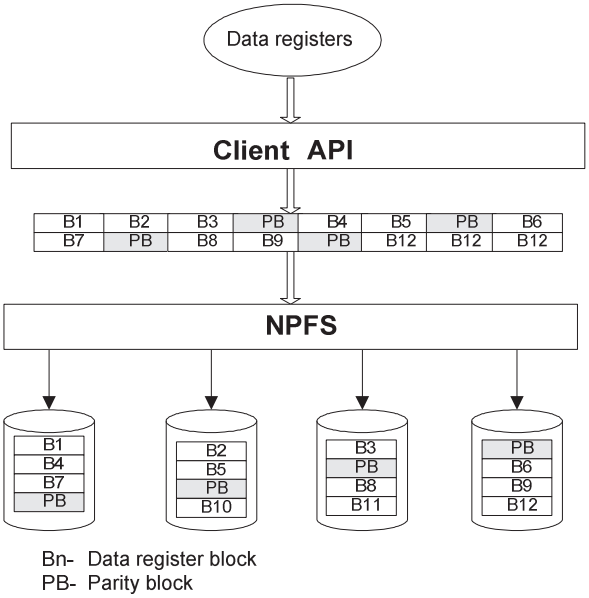


Fig. 6: NPFS data register distribution with parity blocks.

4.2. Distributed Data Base Management System

ATM agents are the users of the DDBMS. They access database by a proper API (Application Programming Interface) and by a high-level query language. Much of the query processing is performed in the client agent, counting with a local data catalogue that allows selecting the right remote servers to be queried.

As seen in Fig. 7, clients can access any of the server nodes. Client API has modules that establish and manage communication with servers, in accordance with the following procedure:

- A query is analyzed and evaluated by the client API and, depending on the kind of query, it chooses to access data, through some local DBMS or directly via NPFS. Query is divided into sub-queries;
- Local DBMS processes the sub query and stores the result in a temporary result set;
- Client API combines the partial result sets and locally stores the combined result set.

This scheme needs either a central data catalogue, or a replication system, so the clients can access it. The master block, shown in Fig. 7, is the NPFS master process.

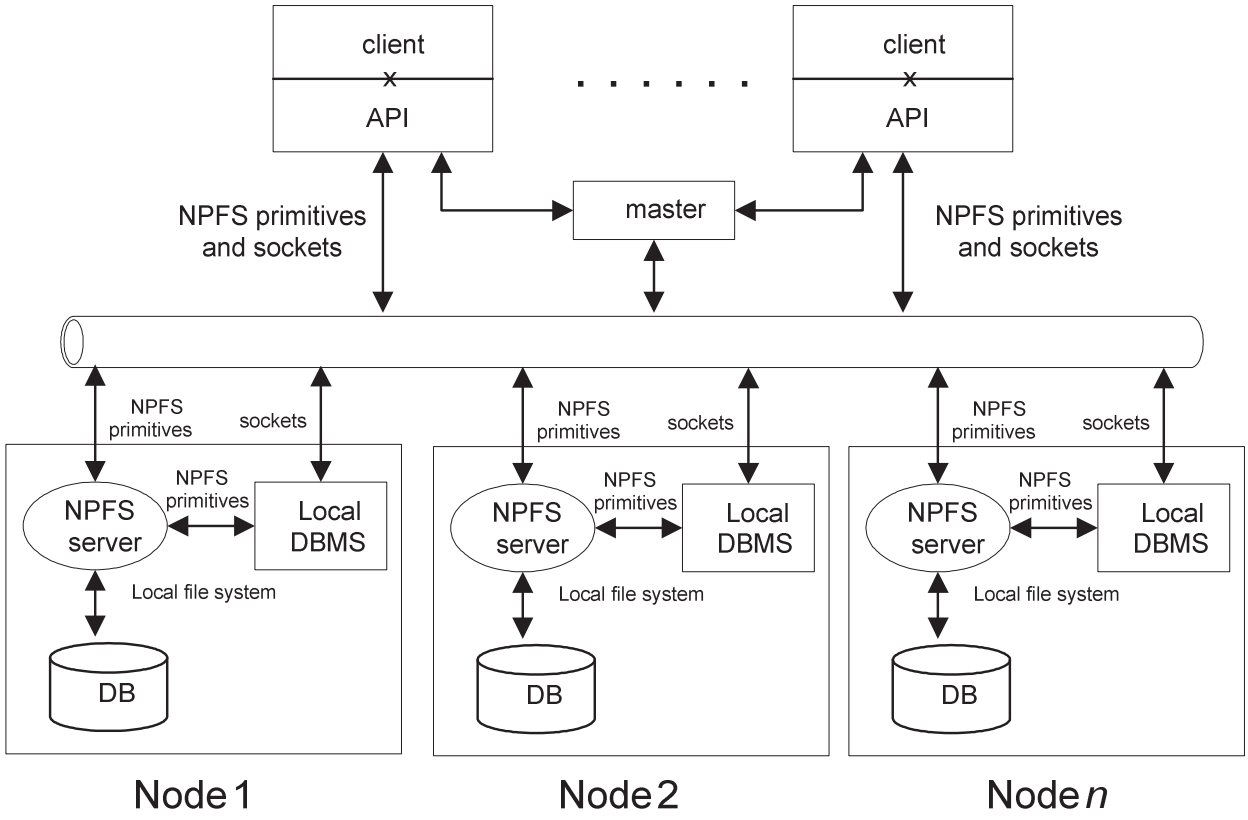


Fig. 7: Distributed Database Management System Architecture.

5. PRELIMINARY CASE STUDY

A preliminary specification of the application software was elaborated, to make a concept proof for the proposed architecture. The number of agents was reduced into three, and their behaviors were described in a use case specification. The basic descriptions of the use cases are following presented.

5.1. Use Case Specification

The diagram of Fig. 8 shows use cases. In short, the goal of these use cases is to make agents share information, and perform strategies to obtain higher efficiency in traffic operations.

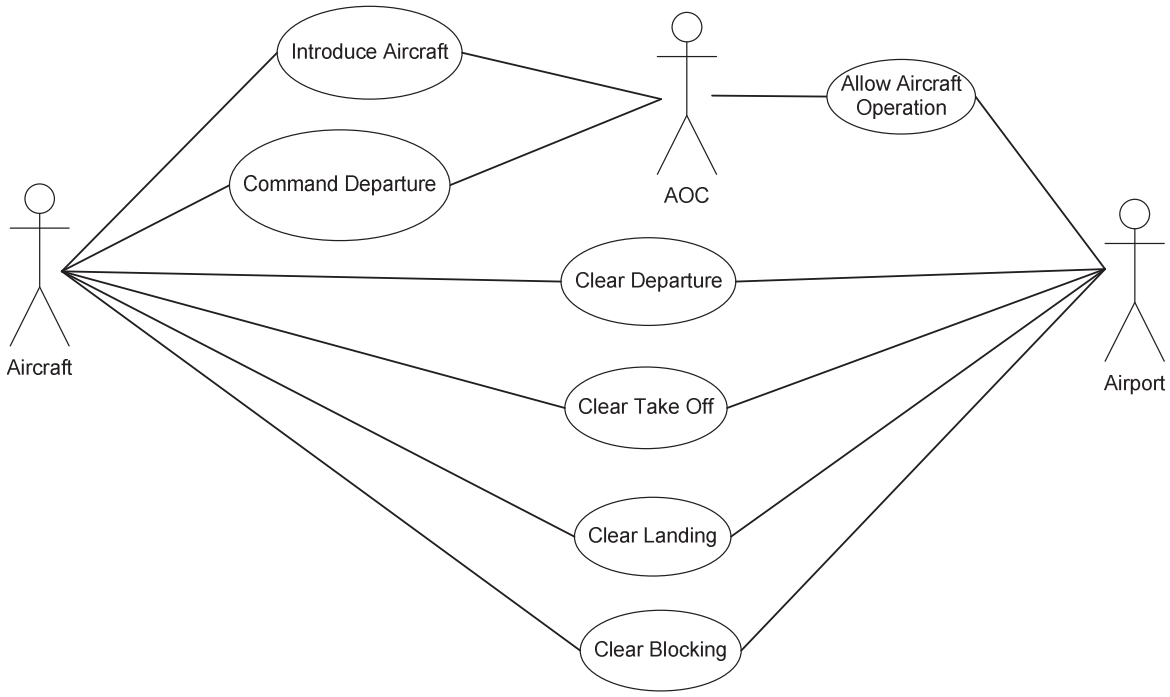


Fig. 8: Use Case Diagram.

5.1.1. Allow Aircraft Operation – Use Case Description

Before beginning its actions, an aircraft is at some airport, and its presence is registered in database, before the simulation begins. For the aircraft starting its operation (to participate in the simulation as an agent), the AOC (Airline Operations Center) needs to request the airport to position the aircraft, in order to allow its operation.

5.1.2. Introduce Aircraft – Use Case Description

Once the AOC is up to be operating the aircraft, it must introduce the aircraft into the scenario. This use case consists in introducing the aircraft agent into the simulation environment, and also in the execution of the specific initialization steps of the aircraft agent. These steps are: programming the first flight plan, establishing connection with the AOC, the Synchronizer and the current airport.

5.1.3. Command Departure – Use Case Description

The AOC commands the departure of an aircraft, blocked at an airport, in accordance with its flight-planning database. Toward that, the aircraft must have been introduced and blocked in that airport. Its flight plan, to be executed in that moment, should be known

and the same of that known by the AOC. After that, the departure command is given, and there are two possible responses given back by the aircraft. The first one reports that the aircraft is ready, and the second response warns that the aircraft is not ready for departure. With this response, the AOC estimates a time Dr to the aircraft to be ready, when then, the aircraft must give a new departure command.

5.1.4. Clear Departure – Use Case Description

An aircraft requests clearance to departure from an airport. The origin airport checks the estimated arrival time in its destination airport, according to its initially programmed departure time. With this information, it verifies whether there will be any delay in its arrival time. It also verifies if the departure queue is not too long and, depending on the conditions, provides a response to the aircraft. The response must be an immediate departure clearance, or an instruction for the aircraft to keep waiting and request its departure again after a while.

5.1.5. Clear Take Off – Use Case Description

The airport, managing the departure queue, determines which aircrafts are first in the queue and, in case the runway is not blocked by other take-off or departure, sends a take off clearance message to the aircraft.

5.1.6. Clear Landing – Use Case Description

An aircraft requests authorization to land at an airport. The airport sets the requesting aircraft at the end of a virtual landing queue, allocating a time slot for its runway, and returns the response to the aircraft. The response can be either an immediate landing clearance or an instruction to wait. In case there is an available time slot, the airport will contact again the aircraft to clear its landing procedure.

5.1.7. Clear Blocking – Use Case Description

From the moment the aircraft receives its landing clearance, up to when it leaves the landing runway, it must receive the clearance to stop and block in some boarding gate or position. In case there is saturation in the boarding positions of the airport, the aircraft may be instructed to wait in some other point of the airport, until some boarding position is available.

5.2. CDM Analysis Scenarios

To study the benefits of CDM for the Air Traffic, the “Departure Clearance” of the use case may be simulated in three different scenarios:

5.2.1. Scenario 1

The departure-clearance airport does not consult the schedule of the destination airport, and takes into account only its own runway availability, to instruct the departure clearance for the aircraft.

5.2.2. Scenario 2

a) The departure-clearance airport consults and takes into account the schedule of the destination airport, but DOES NOT consider the possibility of the aircraft using a speed HIGHER than standard cruising speed, NEITHER A LOWER ONE. With these decision elements, the departure airport may delay its departure clearance, taking into account the destination airport schedule, and considering only the standard performance of the aircraft.

b) The departure-clearance airport consults and takes into account the schedule of the destination airport, and the possibility of the aircraft to use a LOWER cruising speed, in order to MAINTAIN its departure time. However, it DOES NOT consider the possibility that the aircraft may perform a speed HIGHER than its standard cruising speed. With these decision elements, the departure airport may delay or maintain the departure time of the flight.

5.2.3. Scenario 3

The origin airport consults and takes into account the schedule of the destination airport, considering BOTH the possibilities of applying a LOWER cruising speed, and of applying a speed HIGHER than its standard cruising speed. With these decision elements, the departure airport may delay or maintain its departure time, taking into account the schedule of the destination airport.

These three scenarios are expected to be presenting different take off and landing queue behaviors and, therefore, different waiting times for aircrafts, both airborne and on the ground.

5.3. Initial Simulation

The software RAMS Plus 5.24 was used, in order to perform a concept proof to analyze the expected benefits of the presented system architecture. The simulation scenario includes only the six most important airports of the Brasília FIR, namely: Brasília (SBBR), Belo Horizonte (SBBH), Galeão (SBGL) and Santos Dumont (SBRJ), being all flights addressed to Congonhas (SBSP) or Guarulhos (SBGR) airports, both in São Paulo. Fig. 9 illustrates this scenario.

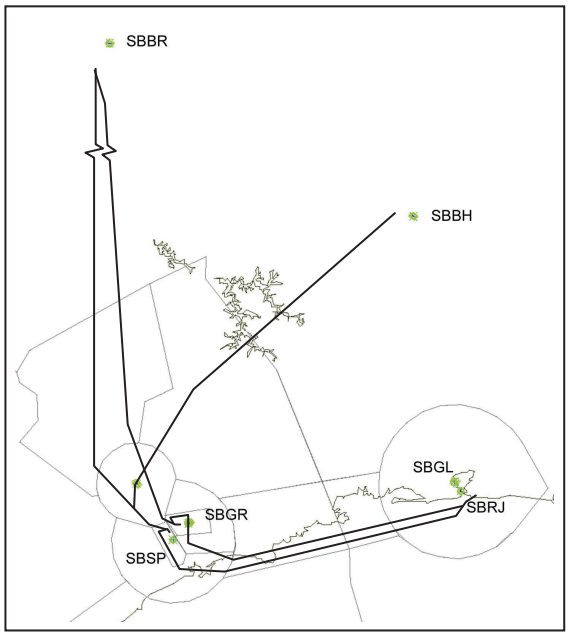


Fig. 9: The airports included in the scenario.

Regarding the time spent, in holdstack waiting orbits, as a crucial cost-increaser factor of a flight, the simulation environment was elaborated with the holdstacks around the destination airports SBSP and SBGR. They are located at points: COLT, DIRK, GURU, REDE, and SANTANA (as shown in Fig. 10).

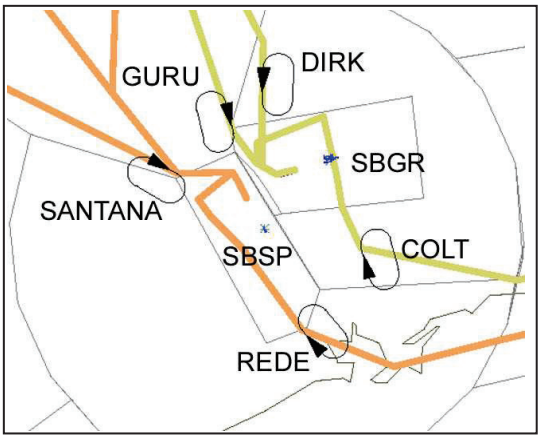


Fig. 10: Considered the waiting orbits.

5.3.1. The CDM impact on Flight Time

The results of Table 1 were obtained through data of the RAMS Plus software simulation, considering real schedule of November 24th 2004, a Wednesday. In total, 200 flights were simulated, arriving in SBSP and SBGR.

It was observed that no waiting orbits were performed to Guarulhos Airport. Possibly, this fact can be explained for the scenario excluding many important flight origin airports addressed to Guarulhos, diminishing considerably its traffic. To obtain Table 1, all waiting maneuvers of less than two minutes were discarded, because the simulator provides some samples which, in practical terms, are not possible. The generated simulation of Table 1 is in accordance with Scenario 1 (sub-section 2.1).

Table 1 - Simulation results of the waiting orbits, run on Scenario 1.

Holdstack	Flights that executed waiting orbits		Time in hours		
	Number of flights	Max. per instant	Total waiting	Mean waiting per aircraft	Maximum waiting per aircraft
REDE	6	2	00:27:07	00:04:31	00:07:44
SANTANA	34	2	03:20:10	00:05:53	00:12:03
TOTAL	**40**	**2**	**03:47:17**	**00:05:41**	**00:12:03**

For evaluating Scenario 3, here proposed (sub-section 5.2.1), we must assume that the destination airport consults flight schedules from the database, and performs a simulation, similar to the flights on Scenario 1, obtaining the waiting time to the flights, which have not departed from the origin airports in feasible time for delaying them. The time spent in the waiting orbits determines the estimated landing time, registered in the database. The origin airport then, consults the database, and verifies whether there will be delay in the arrival airport, according to the traffic data. In case there will be a delay, that flight will not obtain departing clearance, until the estimated delay interval is over. This collaboration way, based on message exchanges, is the working basis of Scenario 3.

Postponing flight departures, foreseen for the waiting orbits, will diminish airborne waiting queues and, therefore, reducing flight costs. In this case, it is extremely important that all information is exchanged in real time, among all of the airport agents, and between the aircraft and the airport.

The results of Table 2 were obtained performing the simulation, according to Scenario 3. A considerable reduction in total waiting time can be observed, if compared with Table 2. The waiting orbits still existing are explained by the rules of Scenario 3, which are not sufficient to completely eliminate the airborne waiting maneuvers.

Table 2 - Simulation result of waiting maneuvers on Scenario 3.

Holdstack	Flights executing waiting orbits		Time in hours		
	Number of flights	Max. per instant	Total waiting	Mean waiting per aircraft	Maximum waiting per aircraft
REDE	1	1	00:05:59	00:05:59	00:05:59
SANTANA	4	2	00:24:10	00:06:03	00:10:31
TOTAL	**5**	**2**	**00:30:09**	**00:06:02**	**00:10:31**

5.3.2. CDM Impact on Fuel Consumption

Scenario 1 was used as baseline to analyze the benefits obtained when operating under Scenario 3 rules. The cost of fuel, spent in waiting orbits used in the routes, considering an aircraft mix estimation for the real combination of flown hours per aircraft model, was illustrated in Fig. 8. This fleet combination was found in literature (SIQUEIRA, PORTO, 04), and was normalized to operations performed in Congonhas Airport (SBSP). For each aircraft model of the fleet, fuel consumption rate was calculated considering its waiting orbit navigation conditions, following the reference in literature (BADA, 2004). The results of these calculations are presented in Table 3.

In Table 3, the distribution of hours, flown per aircraft model, still needs updating. It can be observed that some newer aircraft models, such as Boeing 737-800, have not been included yet; besides, in the last two years there was a renewal on some airline fleets, and VASP operations came to an end, and they were not included either. However, we believe that this distribution is acceptable to estimate mean fuel consumption. In July 2005, R$ 1,50 was the minimum value for a liter of fuel in Brazil, and up to the conclusion of this study, one dollar (USD) may be exchanged up to around R$ 2,35 (BRL) in the Brazilian currency market.

Table 3 - Fuel consumption calculation per aircraft model, and mean consumption to the fleet considered in this study.

Aircraft model	Flight hours	Fuel consumption liters/min	Fuel consumption R$/minute	Weighted consumption R$/minute	**Calculation parameters**
Boeing 737-300	25,85%	43,40	65,10	16,83	Waiting orbit conditions:
Boeing 737-700	22,48%	39,07	58,60	13,17	Altitude: 6000 feet Speed: 200 kt Temperature: 13°C
Airbus A320	22,36%	38,05	57,08	12,76	
Fokker 100	15,58%	26,83	40,25	6,27	
Boeing 737-500	9,96%	24,92	37,37	3,72	Fuel price: R$ 1,50 / liter
Airbus A319	3,77%	35,56	53,34	2,01	
		Mean R$/minute		**54,77**	

This way, it is possible to make savings estimation, as a result of working on Scenario 3, in comparison to Scenario 1. This estimation is shown in Table 4.

To estimate the yearly saved amount, weekend traffic was considered to be the equivalent of 1/3 of the traffic seen in working weekdays.

Table 4 - Saving estimation as a result of working on Scenario 3, in comparison to Scenario 1.

Daily airborne waiting time			Daily fuel saving	Yearly fuel saving
Scenario 1	Scenario 3	Difference		
03:47:17	00:30:09	03:17:08	R$ 16.217,11	R$ 4.735.396,27

FINAL REMARKS

A combination of solutions was here proposed, aiming to increase the efficiency of Air Traffic Management. The multi-agent approach enables the adaptability to new situations and fault tolerance. A preliminary case study can demonstrate that the Collaborative Decision Making (CDM) offers a great potential to increase ATM efficiency. All decisions, considered in CDM, depend directly or indirectly on the message handling components, which provide means for the agent communication, and also assure efficiency and reliability levels for the distributed database.

Further development of this study shall evaluate the impact on traffic safety, concerning human controller workload, which could be done through the extension of the initially performed simulation.

This is also a target of this study - to evaluate the behavior of the proposed architecture, in critical situations, such as the overload of message traffic and database queries.

CREDITS

This work is one of the fronts of the Giga project of RNP (Rede Nacional de Pesquisa / National Research Network – Brazil), supported by FINEP. The authors thank the support of these agencies, and the a precious collaboration of: ICEA (Instituto de Controle do Espaço Aéreo, São José dos Campos); SRPV-SP (Serviço Regional de Proteção ao Vôo – São Paulo); Infraero Aeroportos Brasileiros; the students Miriam Sayuri Yokoyama, Gabriel Iseppe Porto, Pedro Henriques and Rafael Anicet Zanini, who intensively collaborated in the development of simulation experiments and in the project documentation; to FDTE (Fundação para o Desenvolvimento Tecnológico da Engenharia) and FUSP (Fundação da Universidade de São Paulo), for the scholarships support.

REFERENCES

AATT Markets Report. *AATT Overall Research Study* NAS2-14279, 1997.

Bada. *User Manual for the Base of Aircraft Data (BADA)*, Revision 3.6. Eurocontrol Experimental Centre, Brétigny-sur-Orge, France, 2004.

Bloem E. A.; Blom, H. A. P.; Van Schaik, F. *Advanced Data Fusion for Airport Surveillance.* NLR-TP-2002-052, 2002.

CDM. *Collaborative Decision Making Scenarios* – European CDM. Site from Web: http://www.euro-cdm.org/scenarios.html. Accessed in july 2004.

Correa, M.; Camargo Jr., J. B.; Siewerdt, E. Collaborative Decision Making - *Uma Abordagem Multi-Agentes para Tomada de Decisão em Scenarios Complexos.* In Proceedings of III Simpósio Brasileiro de Pesquisa em Transporte Aéreo. São José dos Campos, December, 2004.

Guardia, H. C. C*onsiderações Sobre as Estratégias de um Sistema de Arquivos Paralelos Integrado ao Processamento Distribuído*. Doctoral Thesis - Escola Politécnica da Universidade de São Paulo, São Paulo, 1999.

ICAO. *Manual of Air Traffic Services Data Link Applications*.First Edition, ICAO Doc 9694-AN/955, 1999.

IMA-100-12. *Regras do Ar e Serviço de Tráfego Aéreo. Ministério da Aeronáutica* - Departamento de Eletrônica e Proteção ao Vôo, Brazil, 1999.

Jennings, N. R. *Coordination techniques for distributed artificial intelligence*. In: O'Hare, G. M. P.; Jennings, N. R. (Eds.). Foundations of distributed artificial intelligence. New York: John Wiley & Sons, 1996, p.187- 210.

Ncag 2. *Nova Circulação Aérea Geral, Fase 2*. Comando da Aeronáutica, Depto. de Controle do Espaço Aéreo - DECEA. July 10, 2004. Available in the website: http://www.decea.gov.br (accessed in October 2004).

Oliveira, I. R. et al. *An Evolutive Environment for the Development of Free-Flight*. In: Spitzer, C., Schmocker; U., and Dang, V. N.: Proceedings of PSAM - International Conference on Probabilistic Safety Assessment and Management, 2004, Volume 1, pp. 513-518. Springer-Verlag, Berlin, 2004.

RTCA. *Government/Industry Operational Concept for the Evolution of Free Flight*. Edition 2. Issued 8-16-00, Free Flight Steering Committee. Radio Technical Commission for Aeronautics, 2000.

Siqueira, C. A., Porto, P. P. *Implementação de RVSM no Espaço Aéreo Brasileiro: Estudo dos Custos e Benefícios*. In proceedings of III Simpósio de Transporte Aéreo, São José dos Campos, December, 2004.

Wooldridge, M.; Jennings, N.R. *Intelligent agents: Theory and practice*. The Knowledge Engineering Review, 10 (2):115-152, 1995.

Wooldridge, *M. Intelligent Agents*. In: Weiss, G. (Ed.) Multiagent Systems - A Modern Approach to Distributed Artificial Intelligence. MIT Press, 1999.

APPENDIX A: GLOSSARY

4-D	Four-Dimension
ACC	Area Control Center
ADS	Automatic Dependent Surveillance
ADS-B	ADS Broadcast
AOC	Airline Operations Center
AMS	Agent Management System
API	Application Programming Interface
APP	Approach Control
ATC	Air Traffic Control
ATCS	Air Traffic Control System
ATFM	Air Traffic Flow Management
ATM	Air Traffic Management
ATMS	Air Traffic Management System
CPDLC	Cockpit-Pilot Data Link Communication
DB	Database
DBMS	Data Base Management System
DDBN	Distributed Database Node
DDBMS	Distributed Database Management System
FIR	Flight Information Region
ICAO	International Civil Aviation Organization.
NPFS	Network Parallel File System
RAMS	Reorganized ATC Mathematical Simulator
TMA	Terminal Maneuvering Area