FAIRNESS ANALYSIS WITH FLIGHT COST IMPACT USING THE REINFORCEMENT LEARNING APPROACH

Antonio Carlos de Arruda Junior
Laboratory of Computational Model for Air Transportation (TransLab)
University of Brasilia, C.P. 4466, Brasilia, DF CEP: 70910-900, Brazil
(Phone: 55-61-3310-6558; Fax: 55-61-3107-6379; e-mail: jnarrd@yahoo.com.br)

Li Weigang
Laboratory of Computational Model for Air Transportation (TransLab)
University of Brasilia, C.P. 4466, Brasilia, DF CEP: 70910-900, Brazil (e-mail: weigang@unb.br)

Alexandre Gomes de Barros
University of Calgary
2500 University Drive NW, T2N 1N4, Calgary, Canada (e-mail: debarros@alumni.ucalgary.ca)

ABSTRACT

To avoid or to reduce air traffic congestion, air traffic controllers apply restrictive measures to aircraft both on the ground and airborne. However, the impacts caused by those actions do not affect airlines equally. This paper presents a study on air traffic flow management using a dynamic approach, with a methodology based on reinforcement learning. The model uses agents that recommend restrictive measures to air traffic controllers, based on ground holding. Aiming at ensuring safety and preventing congestion, the paper defines a reward structure based on evaluation functions and reinforcement learning procedures that evaluate factors such as congestion levels, delay imposed to aircraft and airlines and an impact function that includes factors such as financial cost and fairness. Experimental results from the application of the model to the Brasília Flight Information Region show that the restrictive measures generated with it causes delays up to 25% less than with the use of the graphs theory based on Edmonds-Karp algorithm. This study also proves the usefulness of including fairness and economic factors in the optimization problem, without compromising safety.

Keywords: Air Traffic Flow Management, Decision Support System, Fairness Analysis, Impact Analysis, Reinforcement Learning.

This work was partly supported by the Brazilian National Council for Scientific and Technological Development - CNPq under Grants of Proc. 306065/2004-5 and 4739462009-2.

1. INTRODUCTION

In January 2009, 180 thousand flights – 6,000 flights a day on average – carried 10.7 million passengers among the largest airports in Brazil (INFRAERO, 2009). Thus, it can be seen that in just one hour, 250 aircraft are in the air, impacting the airport resources and the air navigation infrastructure.

To co-ordinate this aircraft flow in a safe, efficient manner, air traffic control centers (ATCC) apply Air Traffic Flow Management (ATFM) principles all over the Brazilian airspace.

ATFM is a very complex methodology, highly specialized and strongly based on the air traffic manager's experience (Bianco et al., 2001). Tasks performed by ATFM must address critical issues such as efficiency, equality, adaptability, reliability and safety in airspace management. Its main goal is to ensure flight safety, efficacy and punctuality, seeking a balance between airspace users' needs and the capacity of the airspace and airport infrastructure, subject to weather conditions and aircraft operational limitations (Weigang et al., 2008; Agogino and Tumer, 2009; Bertsimas and Gupta, 2009; Timoszczuk et al., 2009).

To manage the system, predictions are made for demand, infrastructure allocation and points of congestion during the planning stages of ATFM. However, ATCCs are often faced with real-time, unpredicted occurrences such as changing weather conditions, equipment failures, flight delays, non-schedule flights and others that may lead to saturation at certain air traffic control (ATC) sectors.

In Brazil, an ATC sector is considered congested when it contains 12 or 13 aircraft, and saturated when that number reaches 14. To minimize such occurrences, restrictive measures are applied for flow control. Examples of common restrictive measures are ground holding, airborne holding, dynamic re-directing, speed control, en-route separation between aircraft and alternative routes (Dib et al., 2007). However, every action incurs a cost to the system, as it can potentially lead to flight delays and cancellations (Bertsimas and Gupta, 2009).

The Brazilian airspace is divided into five Flight Information Regions (FIRs). Currently, flow management decisions made by the controllers and managers are based on their own experience. Therefore, even with the support of highly advanced technologies, human experience is still an important factor in the efficacy of restrictive measures. Such measures are decided upon at a tactical level, with no attempt to quantify their impact on airports, airlines and the economy as a whole (Weigang et al., 2008).

The Air Transport Association estimates the cost of flight delays to U.S. airlines in 2005 at US$ 5.9 billion (Bertsimas and Gupta, 2009; USDT, 2008). The U.S. Federal Aviation Administration estimates that flight delays and cancellations caused over 1.6 million hours of delay, and their total cost was estimated by the Joint Economic Committee at more than US$ 41 billion (Agogino and Tumer, 2009; JEC, 2008). Moreover, despite the recent economic downturn in 2008-09, several studies show a continuous, long-term growth in air traffic that is expected to continue in the foreseeable future (Rolim et al., 2004; Crespo et al., 2008; Weigang et al., 2008; Agogino and Tumer, 2009; Bertsimas and Gupta, 2009; Timoszczuk et al., 2009), contributing to the increasing complexity in air traffic flow management decisions.

To address these issues, this work aims at defining and evaluating a Decision Support System (DSS) comprising a computer model using multi-agents theory and reinforcement learning (RL). Recently, the research group of the laboratory of Computational Model for Air Transportation (TransLab) in the Dept. of Computer Science of the University of Brasilia (UnB) developed Air Traffic Flow Management and Decision Support System – SISCONFLUX, the architecture of which consists of scenario prediction and route choice modules using graph theory and flow maximization algorithms (Weigang et al., 2008; Crespo et al., 2008). SISCONFLUX is a tool that supports ATC decision making, suggesting measures for reducing air traffic

congestion and delays. This work uses the reinforcement learning technique of Artificial Intelligence (AI) to collaborate with SISCONFLUX, through the use of agents that use past experience to make recommendations about the ATFM actions/measures to be taken by traffic flow managers. The measures are determined based on a rewards structure with evaluation functions using real data from the Brasília FIR (FIR-BS), which is coordinated by the Brazilian Air Navigation Management Centre (Centro de Gerenciamento de Navegação Aérea – CGNA) and is responsible for about half of the national air traffic flow.

The results of the experiments run with the proposed model show that the total system delay can be reduced up to 25% as compared to the graph theory application, and factors such as financial costs can be taken into account in the decision process without compromising safety and mobility. This study also proves the usefulness of including fairness and economic factors in the optimization problem, without compromising safety

The remainder organization of this paper is as follows: Section 2 briefly describes ATFM concepts related to the proposed model. Section 3 describes the ATFM-related work using computational models and the proposed approach. Section 4 shows the modeling of the environment, the rewards structure, the agents' selection and its actions. In Section 5, the system architecture is described. Section 6 gives the experimental results of the prototype. Finally, in Section 7, the research conclusion is presented together with the relevant references.

2. AIR TRAFFIC FLOW MANAGEMENT

Air Traffic Flow Management (ATFM) aims to predict overloads in the aeronautical infrastructure, determine mitigating measures to reduce congestion and delay, ensuring an optimal air traffic flow in areas with excess traffic (Rolim et al., 2004; Dib et al., 2007; Crespo et al., 2008; Weigang et al., 2008; Arruda Junior et al., 2010). In Brazil, those measures are defined at three levels:

- *Strategic Planning* – undertaken at least 24 hours in advance, through the evaluation of the demand-to-capacity ratios expected throughout the planning horizon and the establishment of actions that may be necessary to avoid congestion;
- *Pre-Tactical Planning* – between 2 and 24 hours in advance, taking into account possible changes in the infrastructure, weather conditions and demand;
- *Tactical Planning* – up to two hours in advance, taking into account the up-to-the-minute state of the air traffic system (Rolim et al., 2004; Crespo et al., 2008).

The model proposed in this study focuses mainly on tactical planning. Actions taken in this phase have a direct impact on elements of the airspace system that are not included in the analysis.

To facilitate its management and operation, the airspace is divided into several areas called flight information regions (FIRs). Those FIRs are managed by CGNA. There are five FIRs in Brazil, as illustrated in Figure 1. Each FIR is further sub-divided into sectors.



Figure 1- Brazilian Flight Information Regions

Figure 2 shows the operational configuration of Brasília FIR (FIR-BS).

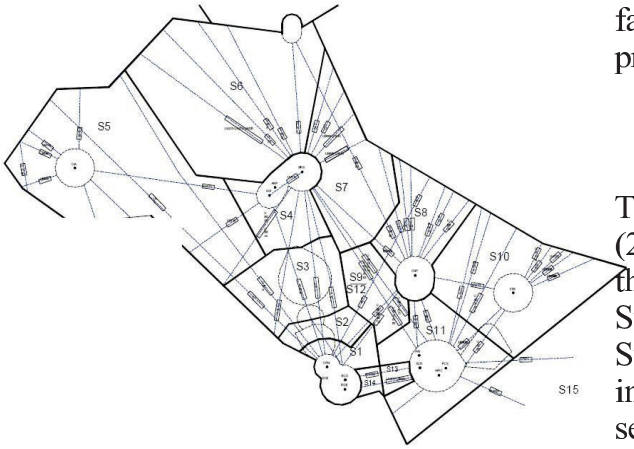


Figure 2- FIR-BS Operational Configuration

Within the sectors, aircraft fly on pre-established routes called airways. Terminal Maneuvering Areas (TMA) are areas that contain one or more airports and are located in the confluence of several airways. One TMA corresponds to a portion of the airspace surrounding one or more airports, with specific routes and procedures for aircraft on final approach to and aircraft departing from an airport.

Each sector is controlled by one air traffic controller, who limits the number of aircraft that can be present in a sector at any given time. Should the predicted number of aircraft in a sector be greater than its capacity, air traffic flow managers take restrictive measures to avoid congestion, most commonly by holding departing aircraft on the ground. Determining where aircraft departures must be delayed and by how long is a complex problem that has a significant effect on airline operations.

3. LITERATURE REVIEW

With the continuous growth observed in the world air traffic, ATFM has become the focus of research and studies worldwide. Two of the most common techniques used in those studies include dynamic programming and multi-agent systems (Dib et al., 2007; Crespo et al., 2008; Ribeiro et al., 2008; Souza et al., 2008; Weigang et al., 2008; Wolfe et al., 2008; Agogino and Tumer, 2009; Bertsimas and Gupta, 2009; Souza et al., 2009). A review of the works that use multi-agent learning and fairness factors to address ATFM issues is presented next.

3.1. Multi-Agent Systems in ATFM

The computer model proposed by Dib et al. (2008) can be defined as a distributed system that supports ATFM. The Air Traffic Flow and Synchronization Management – Multi-Agent System (ATFSM-MAS) uses artificial intelligence, multi-agent coordination, web service and computer grid technologies. The goal of the model was to adjust aircraft departure and arrival times seeking to optimize the air traffic flow between airports. Under congestion, the solution recommended by the agents can result in three types of action:

- Aircraft en route: adjusting the aircraft speed to meet the new arrival times;
- Aircraft on the ground: holding the aircraft on the ground and negotiating a new departure time with the airport; and
- Aircraft with scheduled departures: delaying the departure.

To establish the new aircraft flow for a given airport, the model uses an evaluation function that contains both a flow index and a congestion index, called Airport Balancing Standard.

This work introduced several innovations in agent modeling, evaluation function definition and modeling techniques, ensuring safety and maximizing aircraft flows.

Wolfe et al. (2008) proposed a multi-agent system for Collaborative Air Traffic Flow Management (CATFM). While ATFM uses a centralized approach, CATFM focuses on the exchange of information and a distribution of some elements of decision making. The proposed model used some simple approaches for route selection and traffic flow management. The authors modeled two types of players: the air traffic control entity, which seeks to maximize global parameters related to system performance; and the aircraft operators who seek to optimize their own objectives such as minimum cost and travel times. The evaluation function used included several

factors for each individual flight, such as terminating passengers, transfer passengers, crew on board scheduled for a subsequent flight and others.

Agogimo and Tumer (2009) proposed a prototype of a multi-agent system for use in a simulated environment using reinforcement learning. The model used software that propagates air traffic trajectories over time, called Future ATM Concepts Evaluation Tool (FACET). This software is a physical model of the U.S. airspace. The agents are modeled as air navigation fixes, such that all their actions seek to maximize system evaluation function, suggestion of restrictive measures based on aircraft in-trail separations. The authors presented several experiments based on actual traffic data combined with simulated traffic around New York City. The results show that the model agents achieved an improvement in the system performance of up to 20% as compared to human managers.

Impact factors such as cost and equity are introduced in the analysis of ATFM by Bertsimas and Gupta (2009). In their model, equity in ATFM is presented as an initial evaluation of the resulting order of aircraft departure, using integer programming. The authors state that, in general, ATFM models do not ensure that the initial order of departures is maintained in the resulting solution. Also, the airlines affected are not taken into account, which could lead to some airlines being more affected than others by the final solution. The authors then define equity as the preservation of the original order of arrivals at an airport. It is noted, however, that the exact order may not be achieved regardless of ATFM restrictive measures taken.

Crespo et al. (2012) presented a comparative study of ATFM measures generated by a computational agent based on reinforcement learning. The goal of the agent is to establish delays upon takeoff schedules of aircraft departing from certain terminal areas so as to avoid congestion or saturation in the air traffic control sectors due to a possible imbalance between demand and capacity.

3.2. Proposed approach

The review of the literature reveals many techniques that can be applied to ATFM. However, each work cited herein was associated to a specific environment and may not be applicable to other scenarios.

The model proposed in this paper builds on the several specific points aforementioned, such as the use of the number of aircraft delay to define a safety index, the total delay imposed to the system as a measure of system fluency, and the definition of a tolerance level for aircraft delays before their impact is considered. This work will also make new contributions by proposing a model based on multi-agent system that evaluates the impacts of the restrictive measures. A new architecture compatible with the system currently in use in Brazil and a new reward structure is developed with the evaluation functions that include factors such as cost and fairness. To achieve this objective, reinforcement learning is used as an approach to get the historical experience that suggests the new actions considering the available restrictive measures based on ground holding to help air traffic controllers.

4. MULTI-AGENT SYSTEMS MODELING

This research proposes a multi-agent system model that uses reinforcement learning for decision support during tactical planning, focusing on the short-term actions by air traffic managers. This section discusses the concepts of multi-agent systems and reinforcement learning, as well as the modeling of ATFM using those concepts.

4.1. Multi-Agent Systems

An autonomous agent can be defined as an intelligent system that interacts with other agents in a dynamic environment, seeking well-defined goals through individual efforts (Sutton and Barto, 1998). Weiss (1999) states that agents are intelligent when they operate in a flexible, rational way in a variety of circumstances, according to the information that is available to them and to their capacity

of perception. In this environment, there may be other agents with similar characteristics. The flexible rational behavior is achieved through the use of key processes such as problem resolution, planning, decision making and learning. It can be said that one of the agent's goals is to choose the best possible action in any situation. The theory of multi-agent systems is used in a wide range of applications and is currently the object of research in several fields of knowledge, such as problem resolution, distributed control systems, planning, resource management, collaborative decision support systems and decision making (Stone and Veloso, 2000).

The agents in a multi-agent system can be programmed with advanced behaviors, which are often necessary for online learning of new behaviors, such that the performance of an agent or a multi-agent system will gradually improve (Weiss, 1999). This technique was used in this work because complex problems such as the interactions between aircraft, airports and air traffic controllers are naturally distributed. Therefore, they require an adaptive multi-agent systems approach (Agogino and Tumer, 2009).

4.2. Reinforcement Learning

Reinforcement Learning (RL) is a technique of Artificial Intelligence (AI) in which an entity learns from its interaction with the environment, by recognizing the individual's state, the actions taken within that environment and the changes in state that occurred after the actions (Sutton and Barto, 1998). As opposed to supervised learning, in which a tutor tells the agent what the correct action for each state is, RL has only a critic who points in the right direction, but does not give the agent the exact answer. The agent then chooses its next actions based solely on its interaction with the dynamic environment. This characteristic gives the agent greater autonomy, allowing learning to happen through trial and error (Kaelbling et al., 1996). The agent must then interact directly with the environment to gather information that will be processed by an appropriate algorithm, in order to choose the actions that will optimize the environment states.

At every time step, the agent assesses the state of the environment and takes an action, causing the environment to change into a different state. After that change, the agent receives a scalar reward, which measures the quality of the state transition. This feedback is less informative than supervised learning, in which the agent is told the correct action, but has no knowledge of its effects (Cherkassky and Mulier, 1998).

In RL, the states, actions, rewards and transition probabilities are defined using Markov Decision Processes (Sutton and Barto, 1998). The environment meets the Markov property if its state summarizes the past in a compact way without losing the ability to predict the future. In other words, the transition probabilities from the current states to the next states depend solely on the current states.

Based on Markov property, RL algorithms are used in the following equations to calculate the actions, transitions, future states and rewards:

- Actions that can be taken in each state: $A(s)$, $s \in S$;
- Transition Probabilities: $P(s_{t+1} = s' \mid s_t = s, a_t = a)$; and
- Rewards: $R(r_{t+1} = r \mid s_{t+1} = s', s_t = s, a_t = a)$.

RL models define strategies for quick decision making, efficient learning and increasing the learning speed. These strategies are defined in the form of four basic elements:

- *Action policy* (*decision making or control*): responsible for the agent's behavior, comprised of states as a function of actions, $action = \pi(state)$;
- *Reinforcement or evaluation* (*reward*) *function*: defines the goal in the reinforcement learning problem. The function is associated to a state or to an environment action-state, and to a measure of gain. The agent's goal is to maximize the long-term cumulative gain;
- *Value function*: the value (or utility) function indicates the total gain that can be

accumulated in the future from the current state; and

- *Environment model* (*optional*): mimics the environment behavior in which, given state s and action a, the model predicts the next state and the associated gain.

Thanks to its flexibility, RL has been used as a solution to several types of problems. Intelligent agents have been used in games (Tesauro, 1995), elevator control (Crites and Barto, 1996) and road traffic control (Serra, 2004).

One of the most used algorithms in RL is Q-learning, proposed by Watkins (1989). This algorithm works with an evaluation function $Q(a, s)$, representing the Markov decision process. The value of this function is updated during the agent's learning process. Its values, based on states s and a, are stored in a simple table named *Q-Table*. In the Q-learning algorithm, the agent has a set of possible actions a for every state s. The choice of the action will be based on the historical rewards for that specific state/action combination. If an agent chooses a good action, its reward will be positive, otherwise it will be negative. The better the historical value of the reward for a given state/action combination, the greater the odds of the agent choosing that action again.

The Q-learning algorithm was chosen for this work because it converges to the optimal value of $Q(s, a)$ regardless of the policy used, with the advantages that the agent is not told which actions to take, uses trial and error search, there is the possibility of reward with short-term and long-term gains, and explore/exploit trade-off can be used (Russel and Norvig, 1995). Therefore, it allows more space for testing and evaluating different evaluation functions.

4.3. Environment, states and actions

The proposed model is based on optimizing agents represented by modules MBF and MAAD (discussed in detail in Section 5) that recommend actions to air traffic managers, maximizing the system evaluation function. The data used by the agents consists of online information both on the aircraft currently in the system and the ones that are still to enter the system according to flight plans that have been filed with ATC.

Each agent will be responsible for recommending the best computed action to the traffic manager, as well as evaluating the scenario resulting from the restrictive measures and updating its experience history based on this information. The experience history is updated through the Q-learning algorithm proposed by Watkins (1989).

To model the present state of the environment without losing the evaluation of past states, the Q-Table is defined as an array of the historical experiences for agents using the Q-learning algorithm. Therefore, this table was created to present the states and actions.

The environment states are modeled using the same information used by human controllers: the amount of aircraft within each sector at any given time. It is this information that the controller uses to identify possible congestion points and to take action to avoid them.

Table 1- Sector load in number of aircraft by time

Date	*Sector 1*	*Sector 2*	*...*	*Sector 14*
11.06.2009 – 9:00 pm	7	3	...	7
11.06.2009 – 9:10 pm	9	4	...	6
11.06.2009 – 9:20 pm	11	3	...	6

Table 1 shows a fragment of the environment states modeled for each sector in FIR-BS at 10-minute intervals.

There are several different ways an agent can with the system, including re-routing, ground holding and en-route separation between aircraft. In this work, the restrictive measures recommended by the agents consist of ground holding only, applied to aircraft on the ground at airports in all terminal areas included in the analysis. Ground holding was chosen as the restrictive measure in this work for being the most frequently used by air traffic managers in Brazil (Crespo et al., 2008).

As explained in Section 2, a Terminal Area is comprised of one or more airports. The ground holding action applied limits the number of

aircraft that can depart a given Terminal Area in a given period of time, thereby reducing the number of aircraft that are airborne, thus reducing congestion. Table 2 shows an example of aircraft separation minima that can be applied to meter the aircraft flow from a Terminal Area. For example, the time interval between departures suggested for the São Paulo Terminal Area (TMA-SP) is between 10 and 15 minutes. For comparison, during peak hours TMA-SP can typically have one departure every minute.

Table 2- Suggested times between departures

Origin	*Minimum*	*Maximum*
TMA-BS	05 min	10 min
TMA-BH	05 min	05 min
TMA-SP	10 min	15 min

In Brazil, depending on meteorological conditions and airport infrastructure, normally a takeoff separation is 3 minutes in the best condition and without delay. In case saturation or congestion of aircraft are detected, this separation will be changed to 5-15 minutes, as the examples shown in Table 2. This separation is the suggestions of agents according to restrictive measures and may further be accepted by the traffic controllers as the restrictive actions.

For every action a, environment state s is evaluated and stored in the Q-Table in the form $Q(s_n, a_m)$, which represents the cumulative evaluations of the several environment states performed with the evaluation functions. Table 3 illustrates this concept.

Table 3- Q-Table: agent's rewards

Q	A_1	A_2	…	A_m
S_1	$Q(s_1, a_1)$	$Q(s_1, a_2)$	…	$Q(s_1, a_m)$
S_2	$Q(s_2, a_1)$	$Q(s_2, a_2)$	…	$Q(s_1, a_m)$
…	…	…	…	…
S_n	$Q(s_n, a_1)$	$Q(s_n, a_2)$	…	$Q(s_n, a_m)$

This table stores all the experience acquired by the agent during its learning cycles. This is table that will be used by the agent to determine the expected reward for every possible action in a given situation, enabling it to choose the best possible action to recommend to the air traffic manager.

4.4. Rewards structure

The proposed model uses agents that acquire experience on the environment through online information on the aircraft impacting it. As previously discussed, agents receive rewards for their actions, and these rewards are used to update the Q-Table. When an action is taken, the agent evaluates the effect it had on the environment using the evaluation function. The value of the reward will be added to $Q(s_n, a_m)$. The system environment is evaluated using one or more evaluation functions, which must reflect the ATFM objectives. The agents' goal is to optimize the evaluation of the system as a whole. A Q-Table with sufficient experience will be generated after a number of learning cycles, enabling the agent to determine which action is the best for each possible state (Watkins, 1989).

In this work, the agents' objectives are:

- to minimize delays in the ATC sectors;
- to minimize delays in the terminal areas;
- to minimize the financial costs to airlines; and
- to minimize inequity among airlines.

Based on these objectives, the evaluation function $G(z)$ is defined as

$$G(z) = \beta C(z) + \delta T(z) + \gamma I(z) \qquad (1)$$

where z is the system under evaluation; $C(z)$ e $T(z)$ represent congestion and delay, respectively (Agogino and Tumer, 2009); $I(z)$ is a measure of financial cost and equity caused by the actions taken (Arruda Junior et al., 2010); β, δ and γ are the weights to match 100% of the income in the evaluation function.

A. Evaluation Function: to minimize delays in the sectors

To compute the congestion in airspace sectors, the agent must verify the number of aircraft at a certain time. It is important to make clear that the airspace can be used in its maximum capacity. In Brazil, congestion is characterized as dozen or more aircraft in a sector at the

same time. That is, the agent must verify that the number of aircraft in a sector is greater than the allowable limit, affecting the safety of the system as a whole.

The congestion function for each sector x is given by

$$Cs(x) = \Theta(V(x) - c)^{(V(x)-c)} \quad (2)$$

where $V(x)$ is the number of aircraft in sector x, c is the maximum number of aircraft a sector can have without being congested, x is the sector under analysis, and Θ is a function that returns zero if the input is less than zero and the value of the input otherwise.

When a sector is in the congestion condition, the output rate of any terminal area is high, and the delay time is proportional to the severity of congestion. This ratio is given by exponentiation of the value of the congestion by $(V(x) - c)$. This indicates that the greater the congestion, the larger the rate of Cs index.

The total system congestion $C(z)$ is given by the sum of the congestion functions for all sectors, in which x is a sector of the scenario analysis:

$$C(z) = \sum_{x \in z} Cs(x) \quad (3)$$

B. Evaluation Function: to minimize delays in the terminal areas

When ground holding is applied, en-route congestion is reduced by reducing the number of airborne aircraft. However, this restrictive measure will have the negative effect of imposing delays both to the aircraft held on the ground and to those in the air. This can also be seen as an impact on the infrastructure of airports, where the flights in the air need space to land. Hence the objective function contains a delay function that will compute the total delay imposed on the aircraft in the system.

The delay for one aircraft on the ground is

$$d_{Agh}(a) = \Theta(t - (\alpha(t_{As}(a), t_{Aes}(a)))) \quad (4)$$

and in the air it is

$$d_{Aah}(a) = \Theta(t - (\alpha(t_{Ac}(a), t_{Aec}(a)))) \quad (5)$$

where t is the current time, t_{As} is the actual departure time, t_{Aes} is the originally estimated departure time, t_{Ac} is the actual arrival time, t_{Aec} is the originally estimated arrival time and a is the aircraft.

The α function estimates the expected time (t_{Aes} or t_{Aec}) when the real time is not yet available; in other words, when the aircraft has not yet departed or arrived at the airport. Function Θ returns zero if the input value is negative, representing an aircraft that is not delayed.

The total delay for all aircraft in sector x is

$$d_{Ath}(x) = \sum_{a \in x} (d_{Agh}(a) + d_{Aah}(a)) \quad (6)$$

and the total delay for all sectors, including terminals and airways, is

$$T(z) = \sum_{x \in z} d_{Ath}(x) \quad (7)$$

C. Evaluation Function: to minimize the financial costs and to minimize inequity among airlines

For airlines, the notion of fairness is to have a program that preserves the order of arrival flights at an airport in accordance with the timetable published in the Online Airline Guide (OAG) (Bertsimas and Gupta, 2009). Thus, the restrictive measures may affect the airlines and passengers in different ways.

The impact of the restrictive measures on airlines is calculated as an index given by

$$I(z) = \sum_{x \in z} (J_{Ad}(x) + F_{Ath}(x)) \quad (8)$$

where $J_{Ad}(x)$ represents the impact on the distribution of delays and $F_{Ath}(x)$ represents the financial impact caused by delays. They represent an index computed with information of all the aircraft in the analyzed scene. The impact of delay, $J_{Ad}(x)$, is given by

$$J_{Ad}(x) = \sum_{a \in x} \left| 100 - \left(\frac{d_{Agh}(a) * 100}{d_{Td} / size(Term, t_{ini}, t_{end})} \right) \right| \quad (9)$$

where d_{Agh} is the ground delay imposed on aircraft a, d_{Td} is the total delay in a terminal area, and *size* is a function that returns the number of aircraft that were delayed in that terminal area during the period between t_{ini} and t_{end}. This function is affected by inequality distribution of delays between aircraft.

Many studies indicate that delays applied to aircraft generate financial costs for airlines; fuel costs, for example, and passengers, through delays and cancellation of flights (Bertsimas and Gupta, 2009; Souza et al., 2008; JEC, 2008). Therefore, the financial impact function is based on the number of delays and cancellations.

The financial impact index is given by

$$F_{Ath}(x) = \sum_{a \in x} \Theta(d_{Ath}(a) - D(a))^{1+\Theta(d_{Ath}(a) - C(a))} \quad (10)$$

where $C(a)$ is the maximum ground delay tolerated before a flight is cancelled, $D(a)$ is the maximum delay allowed for aircraft in the air without compromising safety (Bertsimas and Gupta, 2009), $d_{Ath}(a)$ is the total delay for aircraft a and Θ is a function that returns zero if the aircraft is not delayed.

The term $I(z)$ is used to calculate the impact index. The terms of safety rates and fluency, $C(z)$ and $T(z)$, represent the congestion and total time of delays attributable to the air scenario. It is justified by the relationship of both impacts and, when restrictive measures of delay are applied in the system, always creating some kind of cost to airlines and passengers, which involves additional costs of fuel, crew, taxiways, equipment, etc.

5. ARCHITECTURE

Based on SISCONFLUX developed by Crespo et al. (2008) and Weigang et al. (2008), the architecture of Evaluation and Decision Support – MAAD is proposed in this section. This model predicts congestion and makes recommendations for restrictive measures in the tactical planning phase of ATFM. The SISCONFLUX is composed of the following modules:

- MAPC (*Scenario Forecasting and Monitoring*): uses flight scheduling information to forecast the loads and occupancy of airspace sectors;
- MBF (*Flow Balancing*): determines restrictive measures to be applied in response to congestion scenarios, using flow maximization algorithms and graphs theory to solve Ground Holding Problem (GHP); and
- MAAD (*Evaluation and Decision Support*): coordinates the agents in the evaluation of restrictive measures through a reinforcement learning algorithm. These actions are suggested to air traffic managers in the form of delay times to be applied to airports and terminal areas. The module then evaluates the effects of the measures taken.

This section focuses on the development of MAAD and its integration with SISCONFLUX.

5.1. Evaluation and Decision Support Module (MAAD)

The basic objective of MAAD is to suggest the most appropriate restrictive measures in Air Traffic Flow Management to avoid the congestion or saturation in the FIR sectors (Arruda Junior et al., 2010).

The main differences between MBF and MAAD are the procedures, the features and technologies employed. To identify congestion, MBF uses maximum flow algorithms of the graph theory to calculate optimal solutions. MAAD is developed to evaluate the scenarios in the stores using the historical experience of the air controllers, by means of Multi-agent system associated with Reinforcement Learning. The recommended measurements of the MAAD in SISCONFLUX are offered to air controllers as suggestions.

Another function of MAAD is to evaluate the actions, and further suggest the restrictive measures to traffic controllers. This process is collaborated with Q-Learning to help the controllers to make an optimal decision.
To perform these functions, MAAD is implemented with the following main submodules:

- RL_Agent: this submodule accounts for creating the agents to be used in the model and for defining their characteristics. It also performs the tasks for upgrading the past experiences and prompts the system to search the best action in a congestion situation. This model also suggests actions to traffic managers;
- State_Action_Manager: this submodule manages the possible states and actions to the air scenario. It is developed with functions as inserts, selects and updates the states and actions, through the information originated from the environment. It also accounts for maintaining the experience in the Q-Table. The best action is calculated for a given scenario, according to the specific rules of the Q-Learning algorithm;
- Evaluation_Function: this submodule is implemented to evaluate the reward functions. It can also be used to analyze the suggested action, the restrictive measures applied by the air traffic manager and final scenario; and
- Agent_Result: this submodule manages the storage of all the historic states of the environment and the actions taken by the traffic manager. While the Q-Table stores the computation results regarding the evaluations computed in the scenario, the submodule stores a picture of the environment in situations of congestion and saturation of the sectors, without concern about the evaluations performed.

With these modules, MAAD is developed as an agent system with reinforcement learning algorithms to maximize a comprehensive reward system making evaluations made by the number of aircraft in airspace sectors, the time delay associated with each, and additional information that indicates the impact on some previously determined factors.

5.2. Proposed model architecture

The proposed model is based on the concept of Decision Support Systems (DSS) such that, with the integration of all the modules, the SISCONFLUX architecture will have new functionalities, such as a new graphical interface and reinforced learning module. The new architecture is illustrated in Figure 3.

evaluates the resulting scenario through the

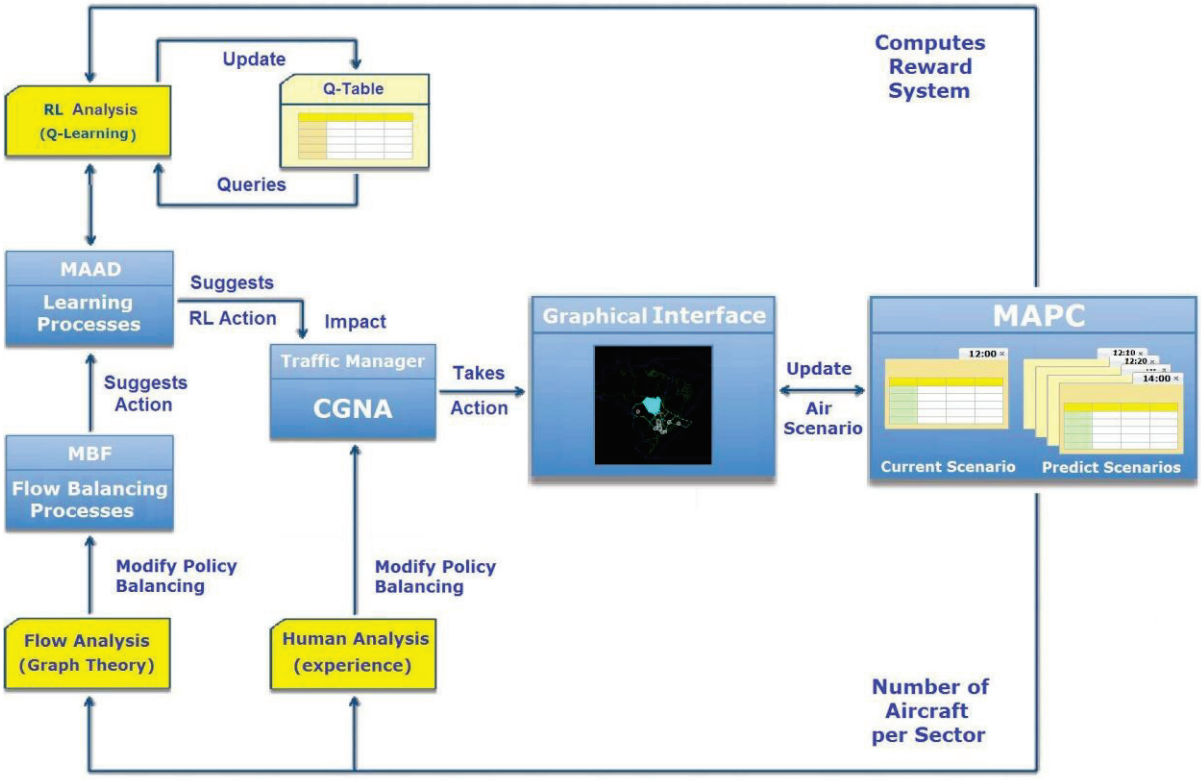


Figure 3- Proposed Architecture of MAAD to Collaborate SISCONFLUX

With the new architecture, the model will work in the following phases:

1. MAPC forecasts the loading levels at the airspace sectors, identifies where congestion is expected and activates MBF;
2. MBF balances the aircraft flows in all the sectors, suggesting ground holding measures based on flow maximization algorithms and graphs theory, using Edmonds-Karp algorithm (Souza et al., 2008);
3. MAAD agents evaluate the actions recommended by MBF and adjust them based on the experience accumulated in the Q-Table, providing revised recommendations;
4. The air traffic manager analyses the actions recommended by the system and implements the actions, which may or may not be the ones suggested by the system;
5. The agents are informed of the restrictive actions applied by the manager and evaluation functions, using the calculated rewards to update the experience database;
6. The system moves to the next evaluation period, starting over with phase 1.

The proposed architecture constitutes a semi-automated flow action control, in which actions are decided and implemented by the human manager based on the recommendations made by the agents. In this configuration, the final decision is always made by a human manager, who may take into account factors that could not be included in the model.

It is important to note that the restrictive measures applied to aircraft affect various factors and entities in the air scenario, such as passengers, aircraft, crew, airlines and airports, among others. However, in this study, we evaluated the following indexes: 1) the number of aircraft in a sector that characterize a situation of congestion or saturation; 2) the total delay time of the alleged aircraft in air

scenario; 3) the financial cost based on delay times and the equitable distribution restrictive measures between aircraft, based on the problem of waiting on hold on the ground. The choice of these factors is based on researches and several studies in the area of ATFM, as presented in Section 3.

6. ANALYSIS AND NUMERICAL EXAMPLE

To evaluate the behavior of the evaluation functions proposed in the reward structure presented in Section 4, experiments were run with data obtained from the Brazilian Department of Airspace Control. The data included aircraft movements over four days, in the form of both recurrent and occasional flight plans between April 30 and May 2, 2008. This period of time was chosen because in Brazil May 1st is a holiday with a lot of flights; hence the simulated scenario has a significant number of congestion situations.

6.1. Evaluation functions

Since the agent's learning process is based on repeated cycles, the prototype was implemented in a controlled environment, with a fixed number of 5,000 learning cycles.
It should be mentioned that in the researches (Dib et al., 2007; Agogino and Tumer, 2009; Wolfe et al., 2008), the evaluation function was developed considering the factors such as safety and traffic flow, which is called *congestion function*. In this research, the evaluation function is defined including the above-mentioned aspects, and also the impact of restrictive actions, which is called *impact function*.
In Table 4, two proposed functions are presented. The related parameters are defined in Section 4.

Table 4- Functions used by agents: Safety, Fluency and Impact Factors

Congestion Function (safety and fluency)	*Impact Function (safety, fluency and impact)*
$G_c(z) = \beta\, C(z) + \delta\, T(z)$	$G_i(z) = \beta\, C(z) + \delta\, T(z) + \gamma\, I(z)$

For better understanding the evaluation of the proposed model, the three case studies were planed in the following six phases:

1. *Data collection*: The data was collected along four days at FIR-BS, from April 29th to May 2nd. Air traffic is considered high on April 30th;
2. *Data input*: Since MBF and MAAD are activated by the predicted congestion, the times at which congestion occurred more frequently were input and used in the agent's learning process;
3. *Testing policy definition*: the results were computed such that the restrictive measures were determined every time a congestion state was predicted, throughout the four days of flights used for testing. Since congestion occurs on specific days and times, every time the agent was activated, it was computed as a learning cycle;
4. *Test runs*: the tests run with a fixed number of cycles to be parameterized. The algorithm converged to the best actions computed. Also, the processing history was validated by the Agent_Result submodule and the results of the evaluation functions were compared;
5. *Comparison of the evaluation functions*: the prototype randomly chooses which restrictive measures will be taken by the air traffic manager, enabling the agent to learn from new possible actions and not only from the ones previously suggested; and
6. *Analysis of the results*: the agents were configured to start their learning process with experience values set to zero. After the end of the learning cycles, the experience values obtained were stored and later used for validation.

The test results were selected according to pre-defined day and time. The results of the simulation were formatted and stored in the database within Agent_Result, containing the history of the learning cycle, the actions, and states analyzed by the agents. Table 5 shows a selection of records that make up the agent-results database.

Table 5- Restrictive Measures by Terminal

Id	*Date-Time*	*Action recommended by*	*Vitoria Terminal*			…	*São Paulo Terminal*		
			Min	*Max*	*Real*		*Min*	*Max*	*Real*
12	2008-05-02 14:40:00	MBF	5	5	25	…	3	3	3
13	2008-05-02 14:40:00	MAAD	3	5	25	…	5	5	3
14	2008-05-02 14:40:00	Manager	5	10	25	…	3	5	3
15	2008-05-02 17:10:00	MBF	15	20	20	…	3	3	3
16	2008-05-02 17:10:00	MAAD	5	10	20	…	3	3	3
17	2008-05-02 21:40:00	Manager	5	10	20	…	3	3	3

As an example of the differences in restrictive measures, in Table 5 it can be seen that both MBF and MAAD recommended time separations between aircraft of up to 5 minutes for the Vitória Terminal, whereas the manager used 10 minutes. In all the cases, the separations suggested are significantly shorter than the 25-minute separation that was actually applied. As seen in the results analyzed when the congestion level is greatly increased, sometimes traffic managers increase the margin of flight safety by applying delays greater than necessary (Souza et al., 2008).

The terminal areas chosen for the three case studies were:

- São Paulo TMA, which is considered a heavy traffic movement for all the four days examined with 1339 aircraft;
- Vitória TMA, which is considered moderate traffic days with 863 aircraft movements; and
- Amazonas TMA, which is considered days with low traffic volumes, 467 movements.

6.2. Case study 1: Heavy traffic terminal

Table 6 shows the results of the São Paulo TMA simulation, in which the two functions were compared.

Table 6- São Paulo TMA: Comparison of best restritictive actions

Terminal	*Congestion Function*	*Impact Function*	*Result Analysis*	*Aircraft Movements*
TMA-SP	CY (20,25) SP (15,20)	CY(20,20) SP (15,20)	Delay reduced	1339

In order to reduce congestion in São Paulo TMA (TMA-SP), a delay of 20 to 25 minutes must be applied to the Cuiabá TMA (CY 20, 25), see Table 6, the congestion function. A delay of 15 to 20 minutes must be applied to São Paulo TMA (SP 15, 20). With the application of the impact function, delays were adjusted in the Cuiabá TMA as (CY 20, 20). In this case study, implementation delays were suggested to be applied for some flights with origin and destination in TMA-SP.

With the impact function, if the traffic manager normalizes the delays on the ground for about 20 minutes, the problem of congestion will be solved without affecting safety factors, which are also evaluated by the functions of agents.

The relationship between the congestion points and the TMAs is given by the air routes that were input in the system and modeled using the Graph Theory (Souza et al., 2008). Its integrity is evaluated by the maximum flow algorithms in the model (Souza et al., 2009).

This small action reduces the delays and chances of flight cancellations both in the affected TMAs and the adjacent ones.
It is worth noting that the adjustment of 5 minutes up or down for a restrictive action is considered a high value when it comes to ATFM. Such actions affect several aircraft in a given period of time. However, in order to restrict takeoffs – thereby reducing the amount of aircraft on the airways – increasing delays is allowed to properly manage the outbreaks of congestion in the sectors mentioned, in order to ensure flight safety.
With this improvement, the structure to evaluate functions has the following features: lower the value of restrictive measures applied and lower the index value of equity and financial impact caused to the affected aircraft. It reinforces the relationship of the evaluated terms of the functions. This behavior indicates that the impact index calculated is directly proportional to the amount of delay time applied to the air scenario.
The suggestion of delay to the São Paulo terminal area, SP (15, 20), was similar in the two evaluation functions. It should also be noted that the congestion level in the simulation was relatively low, with the existence of a maximum of 13 aircraft in a sector, which partly explains the small difference in the best actions recommended by the two different functions in Table 6.

6.3. Case study 2: Moderate traffic

The second case study consists of a scenario with a moderate volume of aircraft traffic. Table 7 shows the results of the congestion observed in the simulation of the Vitória TMA (TMA-VT), where a maximum of 12 aircraft per sector were observed.

Table 7- Vitória TMA: Comparison of best restritictive actions

Terminal	*Congestion Function*	*Impact Function*	*Result Analysis*	*Aircraft Movements*
TMA-VT	CY (05,10)	CY(05,10)	Equal	863

It can be seen that there was no difference in the actions suggested by the congestion and impact functions. In both cases, 5-10-minute delays to TMA-CY were recommended. In this case, the application of either function by the traffic manager will lead to excellent actions to air traffic flow, according to the results computed by maximum flow algorithms and reinforcement learning.
Note that the level of congestion in this case was very low and the restrictive measures affected only one TMA. In this case, the real value of the impact functions lies in its ability to evaluate the impact the measures will have on airline operations, an analysis that is not currently performed by traffic managers.
The actions evaluated in this case study also validate the proposed rewards structure. The impact indices calculated by the agents are shown to decrease with the delays applied through the restrictive measures. With low congestion levels, the actions suggested by the agent that is configured with the impact functions will converge to the same actions recommended by the agent that has only the congestion function.

6.4. Case study 3: Low traffic volumes

In the Amazon Terminal Area (TMA-MN), the delays recommended were longer than in the previous case studies. This situation is explained by the concentration of flights at certain times of the day, leading to longer delays and more flights being delayed – 15 at one point in time.
Table 8 presents the restrictive actions recommended by the model, according to the restrictive actions suggested by the congestion function, the traffic manager must apply a delay between 15 and 20 minutes in the Cuiabá terminal area (CY 15, 20), between 5 and 10 in the São Paulo terminal area (SP 5, 10) and between 15 and 20 in the Amazon terminal area (MN 15, 20), to solve congestion problems.

Table 8- Amazon TMA: Comparison of best restrictive actions

Terminal	*Congestion Function*	*Impact Function*	*Result Analysis*	*Aircraft Movements*
TMA-VT	CY (15,20) SP (05,10)	CY(30,30) SP (05,10)	Delay increased	467

When using the impact function, the suggested delays were 30 minutes in the Cuiabá terminal area (CY 30, 30); 5 to 10 minutes in the São Paulo terminal area (SP 5, 10) and 20 minutes in the Amazon terminal area (MN 20, 20). It is worth verifying that the Cuiabá and the Amazon terminal areas are in the model because they had flights bound to congested sectors and airports in other TMAs.

This particular case study was affected by the occurrence of a national holiday on one of the days for which data was collected. This affected both the flight and the ATC staffing schedules, leading to long delays. In this case, the service quality was also affected and the financial costs were increased.

When the air sector is congested, the traffic controller will apply restrictive measures to avoid possible accidents. In some extreme situations, it involves the cancellation of flights. Therefore, with increasing complexity of air traffic management due to an overloaded scenario, human controllers seek to increase the flight safety margin, usually by applying actions with greater values than necessary. This action will cause more problems in traffic flow management (Souza et al., 2009).

This is the main reason for using reinforcement learning for analyzing the effectiveness of the restrictive measures with the optimal results generated by maximum flow algorithms. This process emphasizes a higher level of security with the result of increasing the total delay in the air scenario, and the impact function was also affected by these actions.

This result shows that the proposed evaluation functions are sensitive in this context. Since the impact function takes into account other factors such as safety and the cost of flight cancellations, the restrictive actions recommended by the model with the use of the impact function included longer delays applied to the Cuiabá and the Amazon TMAs.

6.5. Model performance evaluation

The results presented in Tables 6, 7 and 8 show that the departure delays suggested by the agents, based on the two evaluation functions, behaved as expected.

These behaviors are defined according to the different factors related to the number of aircraft in these sectors and the amount of delay in the air scenario. The impact function checks the sensitivity of these factors and also the impact of the restrictive measures applied, such as the equity in the distribution of the restrictive measures and the cost involved in the delay of the aircraft.

The recommendations made by the agents using the congestion function only included low levels of delay in case studies 1 and 2, but higher levels in case study 3, in which a high concentration of flights was observed in a short period of time. In these situations, the behavior of the reward structure indicates that the fewer the congestions in the air scenario, the lower is the value of the restrictive measures computed.

With the application of the impact function, in case studies 1, 2 and 3, the behavior of this function is similar to the congestion function. It demonstrated that the reward structure is correct, i.e., the amount of delay for the scenario generated is directly proportional to the level of congestion occurred, and the level of impact generated is directly proportional to the amount of delay applied to the air scenario. This behavior is consistent with the a priori expectation: longer delays – i.e. more restrictive actions – being necessary where congestion levels are higher.

It can also be observed that the use of the impact function resulted in recommended actions that are similar to those recommended by the agents with the congestion function in case studies 1 and 2, so the impact function remains as a constant in relation to the restrictive measures suggested, exclusively when the congestion level is relatively low.

In case study 3, the inclusion in the evaluation function of variables associated with the economic impact of the measures led to recommending more restrictive actions. In summary, high congestion levels predicted lead to longer departure delays being applied by the managers in order to minimize the value of the evaluation function. This is consistent with the results achieved by Bertsimas and Gupta (2009), in which the inclusion in the evaluation function of factors that are not

related to the aircraft flow led to increases of up to 10% in the departure delays. However, with a fair distribution of delays, airlines and passengers' satisfaction increases, too. This result can be explained as the high cost, calculated in delays, necessary to maintain equity in scenarios where there are several congested aircraft.

The effectiveness of the Q-learning algorithm can also be verified from samples extracted from the Q-Table. Prior to running the prototype, there were several different combinations observed. However, after approximately 300 learning cycles, the agents' results converged to unique recommendations for each state. It should be noted that the agent's learning speed depends upon the number of decisions it needs to make. Therefore, the convergence time will vary with the specific algorithm used and with the scenarios evaluated – the more often congestion happens, the faster the algorithm will converge.

The prototype was implemented in Java and tested on a notebook with an Intel Core2Duo with 4GB RAM at the TransLab of the University of Brasilia. Linux distribution Ubuntu Desktop (7.10 and 8.04) and Windows Vista were used as operating systems.

On average, running all SISCONFLUX modules, the continuous processing (batch) took 1 cycle with 0:05 minutes, 100 cycles with 4:33 minutes, 300 cycles with 14:43 minutes, and 1.000 cycles with 1:05:03 minutes. Considering the congestion does not occur with the same frequency of the cycles shown, the processing time is considered quite efficient.

To clarify the process used, we emphasize that the congestion forecast in the scenario will start a cycle, which is represented by Phases 1 to 6 of the model explained in Section 5. The prototype analyzes the movements of the first day to the last. When the last day of the flight movements are processed, the system returns to run the first day again.

MAAD is integrated into the the SISCONFLUX system to help air traffic controllers with the ATFM process to get more information about safety and fluency, and more visibility to observe the impact of the related actions.

7. CONCLUSIONS AND FUTURE RESEARCH

The development of intelligent system to manage the growing air traffic flow at airports and in the airspace is essential to the efficiency and economic viability of air transportation systems. This study proposed a model of Evaluation and Decision Support – MAAD that seeks to minimize a multi-objective function using reinforcement learning, for use in air traffic flow management. The model makes recommendations to air traffic managers of restrictive actions (ground holding times) that should be applied to air departures at terminal maneuvering areas. The proposed model uses the framework proposed by Crespo et al. (2008).

Our main objective was to validate a reward structure based on restrictive measure evaluation functions, seeking to minimize the value of those functions with restrictive measures applied to several sectors within the Brasília FIR.

MAAD is proposed to be incorporated in the existing SISCONFLUX framework, and a prototype including an impact function was tested in simulations using data collected from the Brazilian Air Navigation Management Centre (CGNA). The model using the Multi-agent system associated with Reinforcement Learning, produced satisfactory results, consistent with the history of real actions taken by traffic flow managers.

To perform these tests, real data were collected from the FIR-BS (FIR-Brasilia), because the movements of this region airspace have about 50% of all the Brazilian air traffic flow, with common scenarios of real congestion situations, enabling more reliable, effective and efficient testing and case studies.

The evaluation function proposed in this study includes variables that account for the financial impact of the restrictive measures, including the cost of delays to airlines. This additional information allows the traffic manager to evaluate the impact of their restrictive

measures in a much broader sense than just from the traffic flow perspective.

The case studies showed that the proposed evaluation functions that comprise the reward structure could help the agent to decide intelligently and objectively about the restrictive measures to be taken. The longer delays recommended during high congestion level times are consistent with the behavior expected from the model. Nevertheless, using the impact function, the measures suggested by the agents of MAAD have the effectiveness to reduce delay by 25%, less than the time calculated by the maximum flow algorithms of the Graph Theory. In addition, this study indicates that factors such as equity and financial costs can be used together, without affecting aircraft safety and flow.

Observe that the restrictive measures proposed by this model are limited to ground holding at the origin. A more comprehensive model should also include other possible measures to be taken by traffic flow managers, such as air holding, changes in aircraft speed and heading, among others. Still, the proposed model includes several innovations, such as:

- Multi-agent system associated with Reinforcement Learning applied to decision support systems in air traffic flow management;
- A model architecture focused on the Brazilian air traffic system, integrating several modules that use different approaches;
- An evaluation function that includes other factors in addition to those related to aircraft flow, such as the financial cost of the restrictive measures to airlines, using the Q-learning algorithm; and
- A model that recommends ground holding restrictive measures.

Future research is to introduce new approaches to coordinate agents' actions as well as the application of new evaluation functions based on other aspects that affect or are affected by restrictive measures in the air traffic management process.

We also propose to implement MAAD in the real environment, and to integrate all the FIRs operated in the Brazilian Air Navigation Management Centre (CGNA), and other related air traffic control centers.

8. REFERENCES

AGOGINO, A.; TUMER, K. Learning indirect actions in complex domains: action suggestions for air traffic control. **Advances in Complex Systems (ACS)**, v. 12, n. 4, p. 493–512, 2009.

ARRUDA JUNIOR, A. C.; LEITE, A. F.; ALMEIDA, C. R. F.; CRESPO, A. M. F.; WEIGANG, L. Fairness analysis with cost impact for Brasilia's flight information region using reinforcement learning approach. In: 13th International IEEE Annual Conference on Intelligent Transportation Systems, Ilha da Madeira. **Proceedings of the ITSC2010 Conference**, v. 1. p. 539-544, 2010.

BERTSIMAS, D.; GUPTA, S. Fairness in air traffic flow management. **INFORMS Meeting**, San Diego, October 2009. http://web.mit.edu/people/shubhamg/Papers/fairness-paper.pdf.

BIANCO, L.; DELL'OLMO, P.; ODONI, A. R. (eds.) **New concepts and methods in air traffic management.** Springer, p. 31-44, 2001.

CHERKASSKY, V.; MULIER, F. **Learning from data: concepts, theory, and methods.** New York: Wiley, p. 60-65, 1998.

CRESPO, A. M. F.; WEIGANG, L.; BARROS, A. Reinforcement learning agents to tactical air traffic flow management. **The International Journal of Aviation Management**, v. 1, n. 3, p. 145 - 161, 2012.

CRESPO, A.; AQUINO, C. V.; SOUZA, B. B.; WEIGANG, L.; MELO, A. C. M. A.; CELESTINO, V. R. R. Distributed decision support system applied to tactical air traffic flow management in the case of CINDACTA I. **Journal of the Brazilian Air Transportation Research Society**, v. 4, p. 47–60, 2008.

CRITES, R.; BARTO, A. Improving elevator performance using reinforcement learning. **Advances in Neural Information Processing Systems 8**, p. 1017–1023. MIT Press, 1996.

DIB, M. V. P.; WEIGANG, L.; MELO, A. C. M. A. Approach of balancing of the negotiation among agents in traffic synchronization. **Revista IEEE América Latina**, v.5, p. 338–345, 2007.

INFRAERO. Movimento operacional acumulado da REDE INFRAERO, **Tecnical Report**, 2009. In: http://www.infraero.gov.br/images/stories/Estatistica/2009/mov_operac._0109_revisado.pdf. Secretaria de Aviação Civil

JEC. Your flight has been delayed again, Joint Economic Committee, p.1-12, 2008.

KAELBLING, L. P.; LITTMAN, M. L.; MOORE, A. W. Reinforcement learning: a survey. **Journal of Artificial Intelligence Research**, v. 4, p. 237–285, 1996.

RIBEIRO, V. F.; RIBEIRO, W. R. J.; ARRUDA JUNIOR, A. C.; CRESPO, A. M. F.; TIMOSZCZUK, A. P.; WEIGANG, L. Previsão de cenário para o controle de tráfego aéreo do CINDACTA I. Simpósio de Transporte Aéreo, **VII SITRAER**, 2008.

ROLIM, T. H. L.; PORTELA, T. A. A.; ALVES, T. R. A. O controle do espaço aéreo. DECEA, Departamento de Controle do Espaço Aéreo. ASCOM/DECEA., **Technical Report External Release Version**, 2004.

RUSSEL, S.; NORVIG, P. Artificial intelligence - A modern approach. Englewood Cliffs, NJ, USA: Prentice-Hall, 1995.

SERRA, M. R. G. **Aplicações de aprendizagem por reforço em controle de tráfego veicular urbano**. Dissertação de Mestrado, Universidade Federal de Santa Catarina, Santa Catarina, SC, Brasil, 2004.

SOUZA, B. B.; CRESPO, A. M. F.; CELESTINO, V. R. R.; WEIGANG, L. Balance modelling and implementation of flow balance for application in air traffic management. In: Li Weigang, L.; Barros, A.; Oliveira, I. R. (Org.). **Computational models, software engineering, and advanced technologies in air transportation.** Hershey: IGI Global, 2009.

SOUZA, B. B.; WEIGANG, L.; CRESPO, A. M. F.; CELESTINO, V. R. R. Flow Balancing Model for Air Traffic Flow Management. In: The Twentieth International Conference on Software Engineering and Knowledge Engineering (SEKE'08), 2008, Redwood City, USA. **The proceedings of SEKE 2008**. Skokie-IL, USA: Knowledge Systems Institute, v. 1. p. 417-421, 2008.

STONE, P.; VELOSO, M. Multiagent systems: a survey from a machine learning perspective. **Auton. Robots**, v. 8, n. 3, p. 345–383, 2000.

SUTTON, R. S.; BARTO, A. G. **Reinforcement learning: an introduction.** Cambridge, Massachutts: MIT Press, 1998.

TESAURO, G. Temporal difference learning and TD-Gammon. **Computers and the Humanities**, v. 38, n. 3, p. 58–68, 1995.

TIMOSZCZUK, A.; PIZZO, W.; STANISCIA, G.; SIEWERDT, E. The SYNCROMAX solution for air traffic flow management in Brazil, In: Weigang, L.; Barros, A.; Oliveira, I. O. eds. **Computational models, software engineering, and advanced technologies in air transportation: next generation applications.** USA: IGI Global, 2009.

USDT. U.S. Dept. of Transportation, Bureau of Transportation Statistics. **Airline on-time statistics**, 2008. http://www.transtats.bts.gov/HomeDrillChart.asp

WATKINS, C. **Learning from delayed rewards.** PhD thesis, Cambridge, England, 1989.

WEIGANG, L.; SOUZA, B. B.; CRESPO, A. M. F.; ALVES, D. P. Decision support system in tactical air traffic flow management for air traffic flow controllers. **Journal of Air Transport Management**, v. 14, n. 6, p. 329 – 336, challenges in Latin American Air Transport, 2008.

WEISS, G. **Multiagent systems: a modern approach to distributed artificial intelligence.** The MIT Press, p. 331-376, 1999.

WOLFE, S.; JARVIS, P.; ENOMOTO, F.; SIERHUIS, M.; PUTTEN, B. J. V. **A multiagent simulation of collaborative air traffic flow management.** IGI Global, 2008.