# Tackling the Air Traffic Flow and Capacity Management Problem with Explainable Answer Set Programming

Alexander Beiser[1,*,†]

[1]TU Wien, Vienna, Austria

**Abstract**

Air Traffic Flow and Capacity Management (ATFM) is a key area for achieving and sustaining safe and efficient air traffic. However, due to a shortage of air traffic controllers and increasing levels of air traffic, ATFM becomes increasingly challenging. To address this, stakeholders proposed the integration of Artificial Intelligence into the ATFM domain. Due to the nature of ATFM being a safety-critical dynamic system, any usage of AI must be trustworthy. However, while recent studies have focused primarily on the modeling dimension, the issue of ensuring trustworthiness has largely been neglected. This project will tackle the lack of trustworthiness in AI for ATFM by combining the logic programming paradigm Answer Set Programming (ASP) with explanations. We aim towards solving and explaining real-world-sized instances of the European airspace. To achieve scalability and explainability, we will develop a prototype that combines several state-of-the-art ASP techniques.

**Keywords**

Answer Set Programming, ASP, Trustworthy AI, Explainable AI, Problem Decomposition, Air Traffic Management, Air Traffic Flow and Capacity Management Problem, ATFM, ATM

## 1. Introduction

Air Traffic Flow and Capacity Management (ATFM) is one of the key areas in Air Traffic Management (ATM). While ATM is concerned with the efficient and safe flow of air traffic, ATFM aims to achieve a balance between demand and capacity [1]. *Demand* in ATFM refers to the number of flights, while *capacity* is the maximum number of flights a region of airspace, called a *sector*, can handle. When demand exceeds capacity for a sector, air traffic safety is jeopardised. Controllers must therefore mitigate the imbalance by taking *measures* to either reduce demand or increase capacity. Measures for reducing demand include *delaying* or *rerouting* airplanes. Measures for increasing the capacity of regions is a long-term goal, where the latest research focuses on *dynamic airspace configuration* (DAC) [2, 3].

At the time of writing, ATFM faces unprecedented challenges due to shortages in air traffic controllers, an increase in air traffic, and the development of innovative air mobility concepts, such as drones. These challenges lead to an increase in demand, with a simultaneous decrease in capacity. One key technology towards easing this imbalance is the proposed integration of *Artificial Intelligence* (AI) into ATFM. The integration of AI should increase capacity by automatically suggesting or taking the optimal measures. This integration was proposed by the key stakeholders in ATM, resulting in the *Single European Sky ATM Research* (SESAR) master plan [4] and the AI roadmap of the *European Union Aviation Safety Agency* (EASA) [5]. However, the integration of AI into ATFM is highly challenging, due to the nature of ATFM being a safety-critical dynamic system - minor errors can lead to catastrophic consequences. Therefore, any incorporation of AI must be *trustworthy* [6]. Trustworthy AI is *lawful*, *ethical*, and *robust* [7], where one of the key pillars for enabling trustworthy AI is the development of *explainable AI* (XAI) [8].

Work on the AI models for ATFM has so far primarily focused on mathematical optimization, such as mixed-integer programming (MIP), and machine learning techniques [9]. These techniques show promising results in that they can model relatively large sections of airspace efficiently [10]. Still,

*ICLP-DC 2025 - 21st Doctoral Consortium (DC) on Logic Programming, September 12-19, 2025, University of Calabria, Rende, Italy*
*Corresponding author.
alexander.beiser@tuwien.ac.at (A. Beiser)
https://beiser.eu (A. Beiser)
0009-0009-4252-1043 (A. Beiser)

modeling the European airspace remains infeasible, and most models only consider a small set of applicable ATFM measures. Further, recent work does not properly address the challenge of trustworthy AI or the integration of XAI. Evidently, it remains open how an efficient, trustworthy, and explainable system can be designed.

This project aims to address these challenges by using state-of-the-art methods of the logic programming paradigm Answer Set Programming (ASP) [11] and combining ASP with useful explanations [8]. Our main contributions shall be three-fold: first, the creation of an abstract model of the ATFM challenge in the European airspace for ASP. Second, the development of a hybrid ASP system that is able to solve real-world-sized instances [12]. Third, the integration of an explanation component capable of delivering useful explanations why certain *measures* have been taken [8, 13].

## 2. Background

**ATM and ATFM**. ATM aims at achieving safe and efficient air traffic. For ensuring safety, one typically distinguishes between the *tactical* and *strategic* level. On the tactical level, safety is achieved by separating aircraft [1]. Typically, any two aircrafts must be separated by at least 5 $nm$ horizontally, and 1000 $ft$ vertically [14]. *Air traffic controllers* (ATCs) are responsible for achieving this separation. They monitor traffic, keep contact with the pilots, and provide pilots with instructions. Moving to the strategic level, we observe that ATCs are typically responsible for a *sector*, which is a defined region of airspace. Conversely, airspace is divided *laterally* and *vertically* into sectors. Each sector can handle at most a certain number of airplanes safely per hour - this is its *capacity*. Conversely, the number of airplanes that fly through a sector is the *demand*. Capacity is dependent on various factors, ranging from ATC staffing, over the geometrical shape of the sector, to temporary effects, such as thunderstorms or strikes [1]. On the strategic level, safety is ensured by balancing demand and capacity. This is achieved by taking measures, such as rerouting or delaying airplanes, or reconfiguring sectors.

**ASP**. Answer Set Programming (ASP) [11] is a symbolic AI and logic programming paradigm that has seen a rise in popularity for modeling and solving industrial problems [15]. A program is written as a set of rules, where we will briefly introduce the basics of the syntax and semantics for programs with variables (non-ground programs). A program $\Pi$ consists of rules $r$ of the form $p_1(\mathbf{X}_1) \vee \ldots \vee p_\ell(\mathbf{X}_\ell) \leftarrow p_{\ell+1}(\mathbf{X}_{\ell+1}), \ldots, p_m(\mathbf{X}_m), \neg p_{m+1}(\mathbf{X}_{m+1}), \ldots, \neg p_n(\mathbf{X}_n)$, where $p_i(\mathbf{X}_i)$ is a literal, $l, m, n$ are non-negative integers s.t. $l \leq m \leq n$, and $\mathbf{X}_i = \langle x_1, \ldots, x_z \rangle$ is a term vector that comprises of constants or variables. We say $H_r \coloneqq \{p_1, \ldots, p_l\}$ is the head, $B_r \coloneqq \{p_{l+1}, \ldots, p_n\}$ is the body, where $B_r^+$ and $B_r^-$ is the restriction to the positive and negative occurrences, respectively. *Grounding* refers to the instantiation of the variables with all possible domain values, resulting in a ground program [16].

Semantics is defined over a ground program. Let the Herbrand base HB($\Pi$) be the set of all atoms of $\Pi$. An *interpretation* $I$ is a set of atoms $I \subseteq \text{HB}(\Pi)$. $I$ *satisfies* a rule $r$ iff $(H_r \cup B_r^-) \cap I \neq \emptyset$ or $B_r^+ \setminus I \neq \emptyset$. $I$ is a *model* of $\Pi$ iff it satisfies all rules of $\Pi$. The *Gelfond-Lifschitz (GL) reduct* of $\Pi$ under $I$ is the program $\Pi^I$ obtained from $\Pi$ by first removing all rules $r$ with $B_r^- \cap I \neq \emptyset$ and then removing all $\neg z$ where $z \in B_r^-$ from the remaining rules $r$ [17]. $I$ is an *answer set* of a program $\Pi$ if $I$ is a *minimal model* (w.r.t. $\subseteq$) of $P^I$. For additional information, we refer to the standard literature [18, 19].

**Trustworthy and XAI**. Due to the nature of ATM being a safety-critical dynamic system, every AI application in the ATM domain has to be *trustworthy* [7, 5, 4]. Crucial, for achieving trustworthiness, is the principle of *explicability*, i.e., the ability to generate an explanation *why* a model has taken a particular decision [7]. Moving to a psychological perspective, explanations are a well-studied subject, where explanations for humans are social and generally *contrastive* [8, 13]: people prefer the form "Why P rather than Q?" over "Why P?".

The field of XAI is a rapidly progressing field, which incorporates the insights from psychology and the social sciences with research from both symbolic and subsymbolic AI. Recall from standard XAI terminology that if a model is inherently explainable, it is called *ante-hoc*, otherwise *post-hoc* explainable. As in ASP the model is defined by rules, ASP is *ante-hoc* explainable [20, 21]. Evidently, we are not required to build surrogate models for ASP explanations. The literature of XAI for ASP is primarily

concerned about why an atom is in an answer set, or absent from it. Different XAI methods have been proposed for ASP [22], which includes systems such as `xclingo` [23], `spock` [24], or `xASP` [25].
**Related Work**. To the best of our knowledge, the current employed method for ATFM is based on a heuristic first-come first-serve basis: flights which are *filed* (announced) later are rerouted or delayed first. Further, (truly) dynamic sector allocation is currently not used [1].

In 1996, a first mathematical optimization-based approach was proposed to tackle the ATFM problem [26]. Later approaches incorporated en-route sector capacities more thoroughly [27] and allow planes to be rerouted [28]. Current research focuses on improvements of the computational aspect [10], or on multi-objective formulations [29]. Related to this are approaches that tackle the airspace sector design for optimal capacity [2], hybrid approaches specially crafted for drones in dense urban areas [30], heuristic approaches using large neighborhood search [31], and the future of air traffic control in high-density settings [32]. While the domains of the so far mentioned work overlap with our project, the methodologies differ. Most of the discussed work uses mathematical optimization, with integer programming, while we plan to use ASP. Further, no explanations are given, which is a known problem for optimization approaches in ATM in general [9].

Also related are approaches that focus entirely on the logistic perspective of drones, thereby ignoring ATFM constraints. These methods are called *drone scheduling problems* and include adaptations of the traveling salesperson problem or the vehicle routing problem [33, 34, 35, 36, 37]. In the drone scheduling literature, a promising approach exists which uses ASP [38]. In contrast to these approaches, we consider a different problem domain - mixed airspace - and a different problem setting - ATFM.

Approaches that aim at ensuring safety at the tactical level focus on a single sector. The goal is to identify potential losses of separation between aircraft and perform evasive manoeuvres [39]. Methods include reinforcement learning (RL) [40] or encoder-decoder structures for trajectory prediction [41]. As these methods are currently hard to compare, `BlueSky-Gym` [42] proposes a platform for comparing RL-based tactical separation methods. Some tactical methods focus on explainability, due to their post-hoc nature [40, 43]. In contrast to these methods, we focus on the strategic level.

ASP has been used in the industry [44] and especially in transport logistics, such as train scheduling [15]. In the aviation domain, ASP was used for the already mentioned drone scheduling [38] and the proposed decision support system for the space-shuttle [45]. To the best of our knowledge, no other usage of ASP in aviation is known. Reviews have been published, investigating AI and XAI in aviation [9], the usage of RL for the tactical level [39], or the ATFM problem [46].

## 3. Research Goals

We plan to alleviate the challenges that ATFM faces, with a system based on ASP, capable of suggesting ATFM measures. We ensure trustworthiness by developing an explanation component for the already ante-hoc explainable ASP system. We proceed to describe our three main contributions:

- **ASP model for ATFM**. To the best of our knowledge, no ASP model has thus far been published for ATFM. We will create an abstract model of the European airspace, which allows us to predict and react to capacity overloads of sectors. When a sector overload is detected, the model will be able to propose different approaches towards easing the overload, from delaying and rerouting airplanes, to restructuring sectors. The model will be multi-objective, as there are multiple conflicting objectives, such as not overloading a sector or minimizing ATFM delay.

- **Scalability and Performance**. The aim of the project is to be able to handle real-life example instances sizes comprising over 30.000 flights per day in the European airspace. We will achieve this through a hybrid approach, possibly using multi-shot solving [12], heuristics [47], problem decomposition [48], advanced grounding approaches [49, 50, 51], and the combined usage with of other state-of-the-art paradigms [52]. The data will be gathered through multiple sources, where the primary source is the *EUROCONTROL Aviation Data Repository for Research* [53]. Finally, we plan to investigate the applicability of our model by simulating it on state-of-the-art ATM simulators, such as `BlueSky` [54] and *Mercury* [55].

- **XAI for ASP**. We plan to develop a useful XAI component for ASP, which is able to explain *why* a certain flight was rerouted or delayed - or a sector reconfigured. To integrate latest research, we enable contrastive explanations, such as asking *why was flight $X$ rerouted and not flight $Y$* [8, 13].

## 4. Research Status & Outlook

The project is currently in its first phase, that is, the creation of relatively small abstract models. We briefly describe the ASP parts of the first (exact) model. Our model is split into two encodings - one for deriving solutions to the ATFM problem and one for explanations.

We briefly describe the ATFM solutions encoding, where in the following we show the most important rules and predicates. We assume that all flights are properly *filed* (known in advance). We denote filed flight plans with the predicate `flightPlan/3` (`FlightID`,`Time`,`Location`) and sectors with capacities as `sector/2` (`SectorID`,`SectorCapacity`). Sector connections are modeled as a graph $G = (V, E)$, where $V$ are the sectors and $E$ the edges between sectors, denoted as `sectorEdge/2` (`Sector1`,`Sector2`). When flights are conflicting, we may either delay them or reroute them, which we denote as rescheduling and the predicate `reschedule/1` (`FlightID`). We accept the flight plan, unless a flight must be rescheduled. We denote an accepted flight as `flight/3` with the same terms as for `flightPlan`. This results in the rule: `flight(ID,T,X)` $\leftarrow$ `flightPlan(ID,T,X)`, $\neg$`reschedule(ID)`. We model sector capacity overload as a hard constraint $\leftarrow$ `sector(X,C)`, `time(T)`, `#count{ID:flight(ID,T,X)}` $>$ `C`. Lastly, we optimize for minimized ATFM delay, where we model the ATFM delay of a flight as `arrivalDelay/2` (`FlightID`,`Time`), which results in the weak constraint $\rightsquigarrow$ `arrivalDelay(ID,DIFF).[DIFF@1,ID]`. The not-shown rules encode time and world consistency. An example of world consistency is that if a flight is rescheduled, then the planned departure sector must correspond to the actual departure sector.

We proceed with the description of the XAI component, where we assume a given answer set $A$ of the solution encoding. Rescheduled flights are represented with the predicate `rescheduled/1`. Provided a user wants to know why a flight with ID `FlightID` was rescheduled, the XAI component can answer two questions: "Why was the flight `FlighID` rescheduled?" and "Does there exist another solution where `FlightID` is not rescheduled?" To obtain the answer to the first question, we assume that flight `FlightID` is not rescheduled, while all other flights remain as in the answer set $A$. Further, we relax the hard constraint regarding sector capacity in a separate encoding, which enables us to gather sector capacity violations in a predicate `overload/2`. The answer to the first question is then which sectors are overloaded, when flight `FlighID` is not rescheduled. The second question is answered by assuming that flight `FlightID` is not rescheduled, while solving for a new solution in the solution encoding.

The next steps towards achieving the goal of our project are to create real-world-sized instances of the European airspace and to assess the performance of our initial model and make necessary adjustments accordingly. This is followed by the integration of the dynamic sector allocation into the model and the development of a sophisticated XAI component. Beyond this project, we envision three main pillars of further research: First is the development of a Human-Centered-AI based interface for XAI [56, 57], the integration of a tactical component [39] and the dynamic prediction of sector capacity, affected by environmental effects, by *neurosymbolic AI* [58].

## Declaration on Generative AI

During the preparation of this work, the authors used OpenAI O3 and Grammarly in order to: Grammar and spelling check.

## Acknowledgements

This research was supported by Frequentis and FWF grant 10.55776/COE12.

## References

[1] A. Cook (Ed.), European Air Traffic Management: Principles, Practices and Research, 2007.

[2] G. N. Lui, G. Lulli, M. F. Lema-Esposto, R. L. Martinez, Airspace Sector Design: An Optimization Approach, in: SID24, 2024. doi:10.61009/SID.2024.1.32.

[3] P. Criscuolo, L. Perrotta, G. D. Bitonto, G. Grani, Enhanced Dynamic Airspace Configuration Algorithm, in: SID24, 2024. doi:10.61009/SID.2024.1.27.

[4] SESAR 3 Joint Undertaking, SESAR Master Plan 2025, 2024.

[5] EASA, EASA Artificial Intelligence Roadmap 2.0: A human-centric approach to AI in aviation, 2023.

[6] H. Werthner, C. Ghezzi, J. Kramer, J. Nida-Rümelin, B. Nuseibeh, E. Prem, A. Stanger (Eds.), Introduction to Digital Humanism: A Textbook, Springer Nature Switzerland, Cham, 2024. doi:10.1007/978-3-031-45304-5.

[7] E. Commission, ETHICS GUIDELINES FOR TRUSTWORTHY AI, 2019.

[8] T. Miller, Explanation in artificial intelligence: Insights from the social sciences, Artif. Intel. 267 (2019) 1–38. doi:10.1016/j.artint.2018.07.007.

[9] A. Degas, M. R. Islam, C. Hurter, S. Barua, H. Rahman, M. Poudel, D. Ruscio, M. U. Ahmed, S. Begum, M. A. Rahman, S. Bonelli, G. Cartocci, G. Di Flumeri, G. Borghini, F. Babiloni, P. Aricó, A Survey on Artificial Intelligence (AI) and eXplainable AI in Air Traffic Management: Current Trends and Development with Future Research Trajectory, Applied Sciences 12 (2022) 1295. doi:10.3390/app12031295.

[10] T. Bolić, L. Castelli, L. Corolli, D. Rigonat, Reducing ATFM delays through strategic flight planning, Transportation Research Part E: Logistics and Transportation Review 98 (2017) 42–59. doi:doi.org/10.1016/j.tre.2016.12.001.

[11] M. Gelfond, N. Leone, Logic programming and knowledge representation—The A-Prolog perspective, Artif. Intell. 138 (2002) 3–38. doi:10.1016/S0004-3702(02)00207-2.

[12] M. Gebser, R. Kaminski, B. Kaufmann, T. Schaub, Multi-shot ASP solving with clingo, TPLP 19 (2019) 27–82. doi:10.1017/S1471068418000054.

[13] T. Eiter, T. Geibinger, J. Oetsch, Contrastive Explanations for Answer-Set Programs, in: Logics in Artificial Intelligence, volume 14281 of *LNCS*, 2023, pp. 73–89. doi:10.1007/978-3-031-43619-2_6.

[14] M. Ribeiro, J. Ellerbroek, J. Hoekstra, Review of Conflict Resolution Methods for Manned and Unmanned Aviation, Aerospace 7 (2020) 79. doi:10.3390/aerospace7060079.

[15] D. Abels, J. Jordi, M. Ostrowski, T. Schaub, A. Toletti, P. Wanko, Train Scheduling with Hybrid Answer Set Programming, TPLP 21 (2021) 317–347. doi:10.1017/S1471068420000046.

[16] R. Kaminski, T. Schaub, On the Foundations of Grounding in Answer Set Programming, TPLP 23 (2023) 1138–1197. doi:10.1017/S1471068422000308.

[17] M. Gelfond, V. Lifschitz, Classical negation in logic programs and disjunctive databases, New. Gener. Comput. 9 (1991) 365–385. doi:10.1007/BF03037169.

[18] T. Eiter, G. Ianni, T. Krennwallner, Answer Set Programming: A Primer, in: Reasoning Web. Semantic Technologies for Information Systems, volume 5689 of *LNCS*, 2009, pp. 40–110. doi:10.1007/978-3-642-03754-2_2.

[19] T. Schaub, S. Woltran, Special Issue on Answer Set Programming, Künstliche Intell. 32 (2018) 101–103. doi:10.1007/s13218-018-0554-8.

[20] A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, F. Herrera, Explainable Artificial Intelligence

(XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI, Information Fusion 58 (2020) 82–115. doi:10.1016/j.inffus.2019.12.012.
[21] C. O. Retzlaff, A. Angerschmid, A. Saranti, D. Schneeberger, R. Röttger, H. Müller, A. Holzinger, Post-hoc vs ante-hoc explanations: xAI design guidelines for data scientists, Cognitive Systems Research 86 (2024) 101243. doi:10.1016/j.cogsys.2024.101243.
[22] T. Geibinger, Explainable Answer-set Programming, in: ICLP23 - DC, volume 385 of *EPTCS*, 2023, pp. 423–429. doi:10.4204/EPTCS.385.52.
[23] P. Cabalar, J. Fandinno, B. Muñiz, A System for Explainable Answer Set Programming, EPTCS 325 (2020) 124–136. doi:10.4204/EPTCS.325.19.
[24] M. Gebser, J. Pührer, T. Schaub, H. Tompits, A meta-programming technique for debugging answer-set programs, in: AAAI08, volume 1, 2008, pp. 448–453.
[25] M. Alviano, L. L. Trieu, T. Cao Son, M. Balduccini, Explanations for Answer Set Programming, EPTCS 385 (2023) 27–40. doi:10.4204/EPTCS.385.4.
[26] P. B. M. Vranas, Optimal Slot Allocation for European Air Traffic Flow Management, Air Traffic Control Quarterly 4 (1996) 249–280. doi:10.2514/atcq.4.4.249.
[27] D. Bertsimas, S. S. Patterson, The Air Traffic Flow Management Problem with Enroute Capacities, Operations Research 46 (1998) 406–422.
[28] D. Bertsimas, G. Lulli, A. Odoni, The Air Traffic Flow Management Problem: An Integer Optimization Approach, in: IPCO08, 2008, pp. 34–46.
[29] R. Dalmau, G. Gawinowski, J. Kopec, Multi-Objective Air Traffic Flow Management Through Lexicographic Optimisation, in: SID24, 2024. doi:10.61009/SID.2024.1.05.
[30] T. Chauvin, D. Gianazza, N. Durand, ORCA-A* : A Hybrid Reciprocal Collision Avoidance and Route Planning Algorithm for UAS in Dense Urban Areas, in: SID24, 2024. doi:10.61009/SID.2024.1.30.
[31] R. Dalmau, G. Gawinowski, C. Anoraud, Optimal Air Traffic Flow Management Regulations Scheme with Adaptive Large Neighbourhood Search, in: SID22, 2022.
[32] N. Patrinopoulou, I. Daramouskas, V. Lappas, V. Kostopoulos, A. M. Veytia, C. A. Badea, J. Ellerbroek, J. Hoekstra, V. De Vries, J. Van Ham, E. Sunil, P. M. Ponte Alonso, J. Pedrero Gonzalez, D. Bereziat, A. Vidosavljevic, L. Sedov, Metropolis II: Investigating the Future Shape of Air Traffic Control in Highly Dense Urban Airspace, in: MED22, 2022, pp. 649–655. doi:10.1109/MED54222.2022.9837201.
[33] C. C. Murray, A. G. Chu, The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery, Transportation Research Part C: Emerging Technologies 54 (2015) 86–109. doi:10.1016/j.trc.2015.03.005.
[34] J. Pasha, Z. Elmi, S. Purkayastha, A. M. Fathollahi-Fard, Y.-E. Ge, Y.-Y. Lau, M. A. Dulebenets, The Drone Scheduling Problem: A Systematic State-of-the-Art Review, IEEE Transactions on Intelligent Transportation Systems 23 (2022) 14224–14247. doi:10.1109/TITS.2022.3155072.
[35] G. Macrina, L. Di Puglia Pugliese, F. Guerriero, G. Laporte, Drone-aided routing: A literature review, Transportation Research Part C: Emerging Technologies 120 (2020) 102762. doi:10.1016/j.trc.2020.102762.
[36] S. J. Kim, G. J. Lim, J. Cho, Drone flight scheduling under uncertainty on battery duration and air temperature, Computers & Industrial Engineering 117 (2018) 291–302. doi:10.1016/j.cie.2018.02.005.
[37] S. H. Chung, B. Sah, J. Lee, Optimization for drone and drone-truck combined operations: A review of the state of the art and future directions, Computers & Operations Research 123 (2020) 105004. doi:10.1016/j.cor.2020.105004.
[38] A.-D. Nguyen, L. Pham, N. Lindsay, L. Sun, S. C. Tran, Optimized eVTOL Aircraft Scheduling - An Answer Set Programming Based Approach, in: AIAA AVIATION FORUM AND ASCEND 2024, American Institute of Aeronautics and Astronautics, Las Vegas, Nevada, 2024. doi:10.2514/6.2024-3939.
[39] Z. Wang, W. Pan, H. Li, X. Wang, Q. Zuo, Review of Deep Reinforcement Learning Approaches for Conflict Resolution in Air Traffic Control, Aerospace 9 (2022) 294. doi:10.3390/

aerospace9060294.
[40] W. Guo, P. Wei, Explainable Deep Reinforcement Learning for Aircraft Separation Assurance, in: DASC22, 2022, pp. 1–10. doi:10.1109/DASC55683.2022.9925786.
[41] P. N. Tran, H. Q. V. Nguyen, D.-T. Pham, S. Alam, Aircraft Trajectory Prediction With Enriched Intent Using Encoder-Decoder Architecture, IEEE Access 10 (2022) 17881–17896. doi:10.1109/ACCESS.2022.3149231.
[42] D. J. Groot, G. Leto, A. Vlaskin, A. Moec, J. Ellerbroek, BlueSky-Gym: Reinforcement Learning Environments for Air Traffic Applications, in: SID24, 2024. doi:10.61009/SID.2024.1.10.
[43] Y. Xie, N. Pongsakornsathien, A. Gardi, R. Sabatini, Explanation of Machine-Learning Solutions in Air-Traffic Management, Aerospace 8 (2021) 224. doi:10.3390/aerospace8080224.
[44] A. Falkner, G. Friedrich, K. Schekotihin, R. Taupe, E. C. Teppan, Industrial Applications of Answer Set Programming, Künstl. Intell. 32 (2018) 165–176. doi:10.1007/s13218-018-0548-6.
[45] M. Nogueira, M. Balduccini, M. Gelfond, R. Watson, M. Barry, An A-Prolog Decision Support System for the Space Shuttle, in: PADL01, volume 1990 of *LNCS*, 2001, pp. 169–183. doi:10.1007/3-540-45241-9_12.
[46] V. Aditya, D. S. Aswin, S. V. Dhaneesh, S. Chakravarthy, B. S. Kumar, M. Venkadavarahan, A review on air traffic flow management optimization: trends, challenges, and future directions, Discover Sustainability 5 (2024) 519. doi:10.1007/s43621-024-00781-7.
[47] T. Eiter, T. Geibinger, N. Higuera Ruiz, N. Musliu, J. Oetsch, D. Pfliegler, D. Stepanova, Adaptive large-neighbourhood search for optimisation in answer-set programming, AI 337 (2024) 104230. doi:10.1016/j.artint.2024.104230.
[48] M. El-Kholany, M. Gebser, K. Schekotihin, Problem decomposition and multi-shot asp solving for job-shop scheduling, TPLP 22 (2022) 623–639. doi:10.1017/S1471068422000217.
[49] C. Dodaro, G. Mazzotta, F. Ricca, Blending Grounding and Compilation for Efficient ASP Solving, in: KR24, 2024, pp. 317–328. doi:10.24963/kr.2024/30.
[50] A. Beiser, M. Hecher, K. Unalan, S. Woltran, Bypassing the ASP Bottleneck: Hybrid Grounding by Splitting and Rewriting, in: IJCAI24, 2024, pp. 3250–3258. doi:10.24963/ijcai.2024/360.
[51] V. Besin, M. Hecher, S. Woltran, Body-Decoupled Grounding via Solving: A Novel Approach on the ASP Bottleneck, in: IJCAI22, 2022, pp. 2546–2552. doi:10.24963/ijcai.2022/353.
[52] R. Kaminski, J. Romero, T. Schaub, P. Wanko, How to Build Your Own ASP-based System?!, TPLP 23 (2023) 299–361. doi:10.1017/S1471068421000508.
[53] EUROCONTROL, EUROCONTROL Aviation Data Repository for Research, 2025.
[54] J. M. Hoekstra, J. Ellerbroek, BlueSky ATC Simulator Project: an Open Data and Open Source Approach, in: ICRAT16, 2016.
[55] L. Delgado, G. Gurtner, M. Weiszer, T. Bolic, A. Cook, Mercury: an open source platform for the evaluation of air transport mobility, in: SID23, 2023. doi:10.61009/SID.2023.1.36.
[56] A. Beiser, S. Hahn, T. Schaub, ASP-driven User-interaction with Clinguin, in: ICLP24 - TC, volume 416 of *EPTCS*, 2024, pp. 215–228. doi:10.4204/EPTCS.416.19.
[57] T. Capel, M. Brereton, What is Human-Centered about Human-Centered AI? A Map of the Research Landscape, in: CHI23, 2023, pp. 1–23. doi:10.1145/3544548.3580959.
[58] A. d. Garcez, L. C. Lamb, Neurosymbolic AI: the 3rd wave, Artif. Intell. Rev. 56 (2023) 12387–12406. doi:10.1007/s10462-023-10448-w.