2023 IEEE Symposium Series on Computational Intelligence (SSCI)
Mexico City, Mexico. December 5-8, 2023

Airport ground movement optimization revisited: Coupling airport runway spacing to multi-objective routing and scheduling through genetic algorithms

Francisco Ruben Parra Perea
School of Engineering and Material Science
Queen Mary University of London
London, United Kingdom
f.r.parra-perea@qmul.ac.uk

Jun Chen
School of Engineering and Material Science
Queen Mary University of London
London, United Kingdom
j.chen@qmul.ac.uk

Michal Weiszer
Architecture and Cities
University Of Westminster
London, United Kingdom
m.weiszer@westminster.ac.uk

John Korna
Research & Development
NATS
Fareham, United Kingdom
john.korna@nats.co.uk

Richard Cannon
Research & Development
NATS
Fareham, United Kingdom
richard.cannon@nats.co.uk

***Abstract*—A routing and scheduling optimization approach for the airport ground movement problem considering runway spacing is introduced. An integrated modeling that considers both the routing of aircraft and runway required separations, is implemented through Aircraft Multi-Objective Optimization Algorithm AMOA* and a correct spacing validation module, coupled by a genetic algorithm in search of real-world feasible, yet optimized solutions, for a modern-day aviation setting based on London's Stansted Airport. The proposed genetic algorithms successfully optimize taxiing time and fuel consumption for different airport traffic scenarios while fully respecting runway separation constraints. The difference between algorithms is emphasized to stress the risk of over-evaluation of savings by overlooking real-world operational conditions in the modeling phase of the problem.**

***Index Terms*—Airport, Ground, Optimization, Scheduling, Genetic, Algorithm**

I. Introduction

The task of routing and scheduling aircraft to reach their designated locations within the airport promptly is referred to as the Airport Ground Movement Problem (AGMP) [1]. The goal is to minimize one or more objectives of stakeholder interest such as overall taxiing time, overall fuel consumption, and emissions, while meeting target time windows. The complexity of the problem can vary according to the number of constraints and the fidelity of the modeling of airport operations. The complexity and fidelity of the models should guide the choice of the solution approach. For small airports with few aircraft, few potential conflicts, and a single operational objective, optimal routing can be achieved by simply applying the shortest path algorithm [2] to each aircraft. However, for larger airports, especially during peak hours and in situations in which multiple operational objectives are optimized, and multiple constraints exist, the interaction between aircraft routes often demands a more delicate algorithm. Different stages of the AGMP might be focused on a standalone fashion. For example, AGMP is reduced to a simple scheduling problem when aircraft taxiing is immutable [3]. In contrast, when there are no restrictions imposed on aircraft taxiing, the scheduling problem is neglected and the AGMP is reduced to a routing problem [4]. Another formulation of the AGMP selects from a predefined set of routes that comply to a set of restrictions [5].

A. Single vs Multi-Objective AGMP

In single-objective routing and scheduling algorithms the task of the algorithm is to find a singular solution that excels at minimizing or maximizing a particular value [6]. The shortcoming of this approach is that it can only find a single solution rather than a group of optimal solutions that hold compromises among multiple objectives. An important milestone within AGMP was set in [7] where it was demonstrated that reducing taxing time required longer and more frequent acceleration phases, resulting in higher fuel burnt. This finding mandates current research to step away from a simple shortest-path problem to a multi-objective shortest-path problem. Multi-Objective Airport Ground Movement Problems require algorithms to find a set of solutions that can describe the trade-off between objectives and require the user to, based on posteriori criteria, select the solution that best suits the situation at hand. Numerous efforts have been made to approach this problem, to coalesce different stakeholders' interests to push airport operations to a more optimized state [8].

978-0-7381-4408-5/23/$31.00 ©2023 IEEE

B. Sequential vs Global Approach

AGMP can be tackled by sequential or global approaches. In the sequential approach, aircraft are successively routed one at a time in a predetermined sequence with later aircraft respecting the routes of aircraft routed beforehand. Notable algorithms developed so far include variants of Dijkstra's algorithm [2], such as the quickest path problem with time windows (QPPTW) algorithm which enhances Dijkstra's algorithm by considering the time windows in which previously routed aircraft traverse taxiways on an airport [9]. The A* algorithm [10] further upgrades the previously mentioned algorithms by using heuristics to save computational resources in the search process. Multi-Objective A* (MOA*) and an improved version that selects and expands segments instead of nodes NAMOA* [11] are the backbone of modern sequential AGMP algorithms. In contrast, global approaches aim to select the best routes and schedules for all aircraft simultaneously, within a predefined set of routes and disregarding the order in which aircraft are originally scheduled. Genetic algorithms have been applied to this category with some success [12], [13]. Variables used in such algorithms often focus on the node selection process when routing aircraft.

C. Aircraft Separation Constrains

For safety purposes, aircraft must follow separation constraints during all times and are established for both taxiing and take-off/landing phases of the AGMP. For taxiing, a common practice is to set a separation of 60 meters [14]. For in-air required separation, due to wake vortex dissipation, the gap is dependent on the size and speed of the leading and trailing aircraft and if the aircraft pair are taking off or landing [15]. It is important to note that most routing and scheduling algorithms for AGMP overlook runway procedures thus retrieving solutions that violate the required aircraft spacing, thus rendering the theoretically optimal solutions calculated, inadmissible in real life. Previous work has been done on AGMP to optimize for multiple objectives using genetic algorithms without overlooking runway spacing [21], however, the routing phase of the problem was carried out by algorithms that are now outdated, such as the *kqpptw* algorithm, which has a restricted search space and carries out the routing process by assigning a constant speed for all routed aircraft, which in turn produces results that, when compared to AMOA*, are considered as low-quality solutions. Furthermore, the importance of correct modeling was not measured nor compared in these studies.

II. Problem Description

AGMP is a complex system that involves both routing and scheduling parts. It has been approached differently, respectively considering different modeling options, some being closer to reality than others.

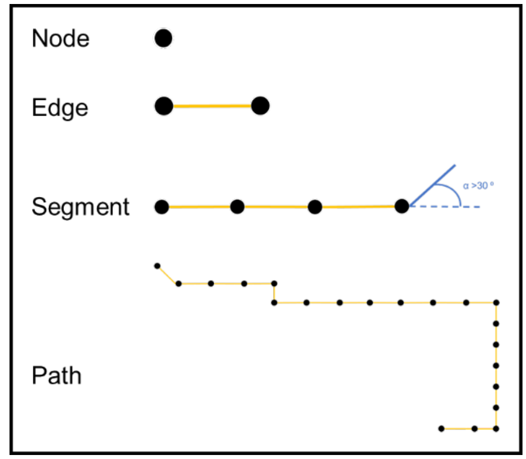

Fig. 1. Element definition in the context of Aircraft Multi-Objective Optimization Algorithm.

A. Ground Movement Problem

State-of-the-art AGMP algorithm A* Aircraft Multi-Objective Optimization Algorithm (AMOA*) [16], has been adopted in this paper as both the starting point for all the approaches presented and as the benchmark result. AMOA* alone has proved to be effective in routing and scheduling aircraft while maintaining cost variables at a minimum. Nevertheless, special care must be taken since the algorithm is purely an algorithm for the taxiing phase and, by itself, it does not consider runway scheduling. Some solutions might not be feasible to implement according to current aviation regulations. A summary of how AMOA* works is included below. For more information regarding AMOA* please refer to [16]. For algorithm comprehension, the following terms are explained upfront:

- Node: Point in space with coordinates (Lat, Lon) in which aircraft operate. It is the building block of a directed graph.1
- Edge: Line formed between two nodes. Aircraft traverse across edges whenever changing nodes across the airport.
- Segment: Group of consecutive edges. Angles formed between consecutive edges must be less than 30 degrees for them to be part of the same segment.
- Path: Group of consecutive segments that form the entire aircraft journey from start to end throughout the airport.

Fig. 1 illustrates these building blocks in the context of AMOA*.

The algorithm follows these enlisted steps:

1) Given a predefined directed graph, the algorithm starts by identifying the neighboring nodes from the starting node of a given operation.
2) For each of the neighboring nodes, segments are constructed: A segment is defined as a group of edges with angles of less than 30 degrees between them (Fig. 1).
3) For each of the segments identified and depending on how many speed profiles are being considered, the

algorithm will output a set of costs assigned to the segment's ending nodes, that, together with the heuristic function, which estimates the remaining costs until route completion, will be used to discriminate non dominated solutions from dominated solutions.
4) Segment ending nodes take the place of the start node in an iterative process until the ending node is reached. At this point, the algorithm outputs a set of non-dominated paths that conform a Pareto front. Based on an a-posteriori established criteria, a single path is selected from the Pareto front.
5) The path is recorded, and time window constraints are updated to consider the path of the scheduled aircraft while scheduling the next aircraft in a sequential fashion.

Let y_i be an indexed speed profile for aircraft i used to traverse a route q_i [17]. The function $T(q_i, y_i)$ returns travel time of aircraft i taxiing on route q_i with speed profile y_i.

$$t^{taxi} = \sum_{i=1}^{M} T(q_i, y_i) \tag{1}$$

Fuel consumption corresponding to speed profiles is calculated by setting thrust levels for each phase of the speed profile [18]. Following a previous study [19], thrust levels are set to 5% for breaking and rapid breaking phases, and at 7% for turning. For Acceleration and constant speed phases, the thrust levels are estimated as a ratio of the calculated thrust and the maximum power output of the power plant R.

$$\eta = \frac{Thr}{R} \tag{2}$$

Thrust Thr is obtained directly from a free-body diagram analysis in which it is assumed that the aircraft is in a dynamic equilibrium:

$$Thr = weight * a + FR \tag{3}$$

The fuel flow $\phi_(v_i)$ associated with thrust level η is obtained by interpolation or extrapolation through the ICAO database at 7% and 30%. Following the implementation in [20], fuel consumption for a given segment is set as the product of fuel flow times the time spent in that state.

$$fuel_s^{ICAO} = \sum_{j=1}^{4} \phi_{v_i} * t_j \tag{4}$$

Comparable to (1), a function $F(q_i, y_i, v_i)$ is defined as the amount of fuel burned for aircraft i of weight category v_i during taxiing on the route q_i following the speed profile y_i [21].

$$f^{taxi} = \sum_{i=1}^{M} F(q_i, y_i) \tag{5}$$

In this work, the ground problem is established as a multi-objective optimization problem that seeks to minimize equations (1) and (5).

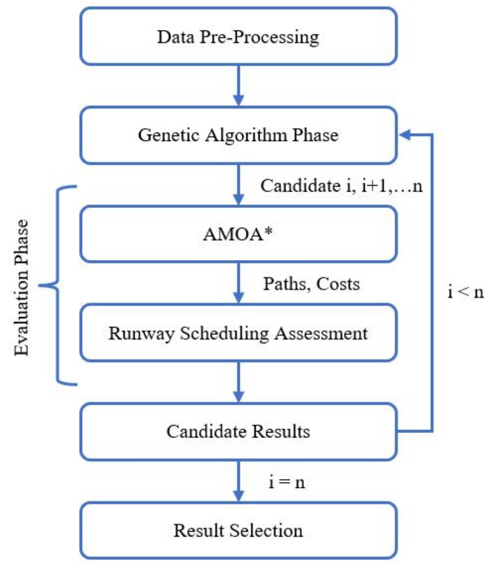

Fig. 2. Flow diagram for GA-Based solutions. Note how AMOA* is part of the evaluation phase of the genetic algorithm.

In the genetic algorithm-based integrated approaches, AMOA* is part of the evaluation phase of the genetic algorithm (Fig. 2). It will be executed along the runway scheduling assessment block every time a new candidate is evaluated. AMOA* inputs are a file containing aircraft information such as the estimated off-block time, assigned gate, and aircraft weight category, candidate encoded in genes from the genetic algorithm problem definition, number of parallel edges to consider, and path selection criteria. Refer to subsection D where segment selection criteria is described thoroughly.

B. Runway Scheduling Problem

Once aircraft are successfully routed by AMOA*, the next step in the process is to ensure proper runway scheduling. According to [15], a certain pre-established time window must be kept between aircraft using an airport runway for safety and operability purposes. These time windows enable wing tip vortex dissipation and guarantee airport proper operation, by complying with in-flight separation constraints. These time windows depend on the size and weight of both leading and trailing aircraft. Different required time windows can be found in [15]. It is important to keep in mind that these times are not immutable in practice, since extraordinary weather and visibility conditions might affect runway spacing from time to time.

Let $M = (A \cup D)$ be the set of total $|M| = m$ arriving aircraft A and departing aircraft D. The wake vortex separations are estimated using minimum separation distance. $V(v_i, v_j)$ returns the wake vortex separation required for weight categories v_i, and v_j, for leading aircraft i and trailing aircraft j. When dealing with runway scheduling, an adequate optimization objective would be to minimize both the time and fuel spent while waiting for proper separation from a previous runway operation. Following the approach by [21], Let r_i be the actual landing time for aircraft $i \in A$. For arriving aircraft, r_i is given, while for departing aircraft it can be calculated as follows. Let d_i denote the time the departing aircraft $i \in D$ arrives at the runway holding point. It can take off immediately, i.e. $d_i = r_i$ if there is enough time elapsed from landing/take-off time r_{i-1} of the previous aircraft $i-1$ to comply with the separation given by $V(v_i, v_j)$, otherwise, the departing aircraft i has to wait at the runway holding point until it is safe to take-off.

$$r_i = \begin{cases} d_i & \text{if } d_i - r_{i-1} \geq V(v_i, v_{i-1}) \\ d_i + w_i & \text{otherwise} \end{cases} \tag{6}$$

The waiting time w_i of the departing aircraft $i \in D$ is denoted below.

$$w_i = \begin{cases} 0 & \text{if } d_i - r_{i-1} \geq V(v_i, v_{i-1}) \\ V(v_i, v_{i-1}) - (d_i - r_{i-1}) & \text{otherwise} \end{cases} \tag{7}$$

The objectives of an optimization algorithm centered on runway scheduling and correct aircraft separation can be formulated as follows:

$$\min \qquad t^{rwy} = \sum_{i=1}^{D} w_i(q_i, y_i) \tag{8}$$

$$\min \qquad f^{rwy} = \sum_{i=1}^{D} w_i(q_i, y_i) * \phi_{v_i}^{idle} \tag{9}$$

C. Algorithms Explored

In this work, 7 algorithms are explored and compared, as to identify both perks and shortcomings in different situations and under different considerations. These algorithms are listed below along with a brief explanation of their differences.

- AMOA* 1PL: AMOA* with 1 Parallel Edge. This algorithm does not asses proper runway spacing.
- AMOA* 3PL: AMOA* with 3 Parallel Edges. This algorithm does not asses proper runway spacing.
- AMOA*-RS 3PL: A runway scheduling step is added at the end of AMOA* 3 PL, in which aircraft are held at runway holding points whenever the solution from AMOA* is not complying with the required runway spacing. Runway scheduling operates as described in previous sections.
- AMOA* Built-in RS 1PL: An additional constraint is incorporated in AMOA* during the routing process: whenever an aircraft reaches an edge that is adjacent to the runway, the algorithm automatically updates the time window constraints for all runway adjacent edges, as to prevent any aircraft entering or approaching the runway for a specified amount of time. In contrast to AMOA*-RS 3PL, there aren't any waiting time additions after the algorithm finds routes for aircraft since runway scheduling is guaranteed by the time window constraint added in this algorithm.
- AMOA* Built-in RS 3PL: Same as AMOA* Built-in RS 1PL but operates with 3 Parallel Edges instead of one.
- AMOA*-RS 3 PL & NSGA-II SI: Genetic Algorithm NSGA-II is used to generate different candidates of AMOA*-RS 3 PL Solutions. A Speed Profile Selection Index variable is used, along a pushback delaying time variable, to compose the search space for a time and fuel-optimal solution.
- AMOA*-RS 3 PL & NSGA-II PI: Genetic Algorithm NSGA-II is used to generate different candidates of AMOA*-RS 3 PL Solutions. A Path Selection Index variable is used, along with a pushback delaying time variable, to compose the search space for a time and fuel-optimal solution.

This previous explanation intends to vaguely generate a draft of how the genetic algorithms operate, and what relationships are established between them, before going on an in-depth explanation in the following sections of this work.

D. Genetic Algorithms

In this article, two genetic algorithms are implemented in order to reduce airport overall aircraft taxiing-related expenditures. Two objectives considered in this study are taxiing time and fuel consumption, described as follows:

$$\min \qquad f_1(x) = t^{taxi} + t^{rwy} \tag{10}$$

$$\min \qquad f_2(x) = f^{taxi} + f^{rwy} \tag{11}$$

Substituting (1), (5), (8) and (9) into (10) and (11) results in:

$$\min \qquad f_1(x) = \sum_{i=1}^{M} T^i(q_i, y_i) + \sum_{i=1}^{D} w_i(q_i, y_i) \tag{12}$$

$$\min \qquad f_2(x) = \sum_{i=1}^{M} F^i(q_i, y_i) + \sum_{i=1}^{D} w_i(q_i, y_i) * \phi_{v_i}^{idle} \tag{13}$$

Since time-delimited periods were considered for this study, all aircraft that complete their respective operations within an established time period are considered. This means that aircraft must leave a gate and successfully reach the runway and take off, or touchdown at the runway and reach a gate within the established time period for them to be considered.

The first algorithm, AMOA*-RS 3 PL & NSGA-II PI, considers two sets of variables or genes: x_1 Holding time before pushback, which is only considered for departures and ranges from 0 to 300 seconds, and x_2 path selection index for all aircraft. In the previous section it was presented that AMOA* generates, depending on the parallel edges considered and both the starting and ending points, a series of P non-dominated paths. These non-dominated paths are generated based on the combination of different speed profiles for different segments along the route of the aircraft. Path selection index is an originally randomized variable ranging from 0 to 100, as shown in equation (15). Once the paths are calculated, these variable values are normalized according to the number of paths found, and a path is selected. Thus, the genetic algorithm problem is defined by equations (12) and (13) with variables:

$$x_1 \text{ for } i \in D$$
$$0s \leq x_1 \leq 300s \tag{14}$$
$$x_2 \text{ for } i \in M$$
$$0 \leq x_2 \leq 100 \tag{15}$$

The second algorithm, AMOA*-RS 3 PL & NSGA-II SI, also considers two sets of genes. While the first set of variables is the same as in the first algorithm, the second is changed for a speed profile selection index for each aircraft: In the original AMOA, depending on the multi-graph setting, a number of n parallel edges are generated on the multi-graph, each of the edges is assigned a speed profile from a predefined speed profile database, based on [22] and [16]. This genetic algorithm bypasses the speed profile selection by randomly choosing a speed profile from the database, via the speed profile selection index. This approach results in a reduction in the search space. Since 10 speed profiles are available from the database, the speed profile selection index is a randomized variable ranging from 0 to 10, as shown in equation (16).

$$x_2 \text{ for } i \in M$$
$$0 \leq x_2 \leq 10 \tag{16}$$

Note how the problem equations are fairly alike, however, the change infused by different variables is drastic. In the first algorithm:

1) The start node and its neighboring nodes are identified.
2) Segments are expanded from the neighboring edges.
3) For each segment several speed profiles are explored depending on AMOA* parameters: number of parallel edges, selection with preference, and intermediate holding option [16]. In this work, both 1 and 3 parallel edge options were explored, selection without preference was used and the intermediate holding option was disabled.
4) Fuel consumption and taxiing time are calculated for the ending node of the segment.
5) At this point a set of positions, with different costs and estimates until completion are known. All dominated solutions are scrapped.
6) This process is repeated until the end node is reached.
7) When the end node is reached, a series of non-dominated paths is available, ranging from quickest and most fuel-consuming to slowest but least fuel-consuming.
8) The path selection index variable is responsible for selecting the one that will be recorded for the aircraft.
9) The process is repeated for all aircraft.

In the second algorithm:

1) The start node and its neighboring nodes are identified.
2) Segments are expanded from the neighboring edges.
3) For each of these possible segments, a singular speed profile is explored, this speed profile is selected by the speed profile selection index variable from the genetic algorithm.
4) Fuel consumption and taxiing time are calculated for the ending node of the segment.
5) At this point a set of positions, with different costs and estimates until completion are known. All dominated solutions are scrapped.
6) This process is repeated until the end node is reached.
7) When the end node is reached, a series of non-dominated paths is available, ranging from quickest and most fuel-consuming to slowest but least fuel-consuming.
8) A path is selected Based on AMOA* parameters and the aircraft is scheduled.
9) The process is repeated for all aircraft.

Since both the path and the speed profile selection index variables in both genetic algorithms influence the final path taken by each aircraft they are referred to as path selection criteria and play a major role in the search for global optimized operations, through the genetic algorithms.

III. Results

In this section, results for three case scenarios are presented. A comparison process is carried out by outlining the main differences between the results of the aforementioned algorithms. All taxiing time results are in seconds, fuel consumption calculation results are in kg of fuel, and all economical costs are in Euros, following the calculation approach in [21].

A. Case 1: Different Scheduling for Non-Conflicting Departures

The first scenario shows four non-conflicting sequential aircraft departures. In the original operation, the aircraft did not encounter other aircraft before reaching the holding positions before entering the runway. Spacing infringement is present in the solutions that do not include runway scheduling. Built-in algorithms appear to improve over the solutions that do not include runway scheduling. However, this might be based on the fact that aircraft are scheduled with an offset that resolves the spacing infringement. This transports the aircraft into the future where all taxiways are clear, in an optimal taxiing situation.

This is possible since this case is an isolated scenario and the solution is said to be “spilled” in time, specifically into the future. For this and future cases, solutions marked by red circles

TABLE I
CASE 1 RESULTS

Case 1	Taxiing Time	Fuel Consumed	Cost
Real Operation Estimate	1216	336.9	809.50
AMOA* 1PL	622.16	301.14	505.60
AMOA* 3PL	628.96	249.56	472.17
AMOA* RS	680.65	258.29	502.61
AMOA* BIN 1 PL	614.63	242.96	460.76
AMOA* BIN 3 PL	679.72	253.89	499.05
AMOA*, NSGA-II PI	604.00	268.62	474.00
AMOA*, NSGA-II SI	615.52	255.42	470.03

TABLE II
CASE 2 RESULTS

Case 2	Taxiing Time	Fuel Consumed	Cost
Real Operation Estimate	3270	665.78	2006.33
AMOA* 1PL	1538.77	456.24	1045.61
AMOA* 3PL	1592.42	405.21	1034.54
AMOA* RS	1639.32	413.13	1062.16
AMOA* BIN 1 PL	1565.88	401.04	1019.14
AMOA* BIN 3 PL	1611.32	397.88	1038.20
AMOA*, NSGA-II PI	1601.56	405.29	1038.89
AMOA*, NSGA-II SI	1575.58	410.37	1030.31

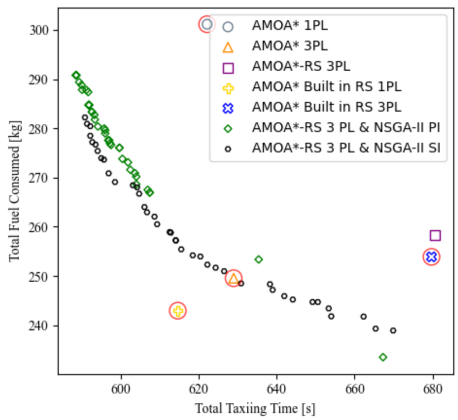

Fig. 3. Pareto front for different AMOA* algorithms optimising for case 1.

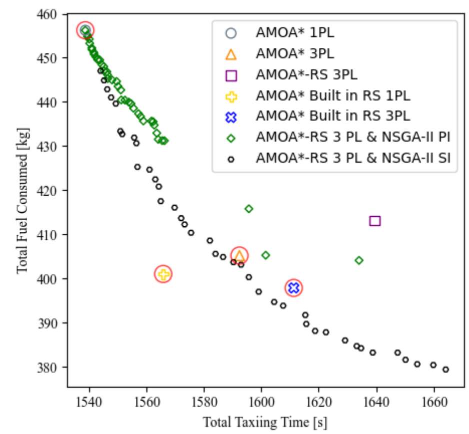

Fig. 4. Pareto front for different AMOA* algorithms optimising for case 2.

are infeasible in practice, whether because they violate runway spacing (AMOA*) or because they might interfere with gate and runway slot allocated times for future aircraft(Built-in). In turn, Genetic algorithms provide a range of solutions that are comparably efficient and can adapt to multiple stakeholder interests. They also demonstrate to optimize over the single case AMOA*-RS 3PL.

B. Case 2: Different Routing and Scheduling for a Mixture of Departures and Arrivals

The second case shows seven aircraft that are either taking off or landing. As the scenarios get more complex, it becomes more noticeable that previously described relations between algorithms are more and more noticeable. In this case, spacing infringements are present in the solutions that don't consider runway scheduling.

Genetic algorithms effectively generate a set of non-dominated solutions (If the time spilled solution of the Built-in variant is overlooked due to feasibility reasons). Genetic algorithms improve noticeably from the single AMOA-RS solution. Broader search spaces, which result in larger variability, are present in the Speed Profile Selection Index (AMOA-RS 3PL & NSGA-II SI) variant of the Genetic algorithm producing a larger Pareto front.

C. Case 3: Different Scheduling for Non-Conflicting Departures

This one-hour period encompasses the routing and scheduling of 38 aircraft and contrasts algorithm performance between small, isolated scenarios and normal airport operation in a high-traffic setting. Several spacing violations are carried out by AMOA*. Built-in versions are further separated from the GA results since the time-spilling effect from one aircraft to another is accumulated. AMOA-RS and the GA version of it are compliant with spacing and do not spill into the future. The difference between spacing-compliant and non-compliant solutions highlights the importance of correct constraint modelling. The larger the period, the greater the deviation of calculated objectives from what can actually be optimized.

TABLE III
CASE 3 RESULTS

Case 3	Taxiing Time	Fuel Consumed	Cost
Real Operation Estimate	16419	3946.67	10502.65
AMOA* 1PL	9652.58	2690.96	6437.64
AMOA* 3PL	9763.94	2510.18	6361.52
AMOA* RS	10429.13	2632.45	6760.30
AMOA* BIN 1 PL	9789.90	2440.98	6324.56
AMOA* BIN 3 PL	9740.02	2504.52	6346.28
AMOA*, NSGA-II PI	10215.80	2776.58	6762.58
AMOA*, NSGA-II SI	10261.51	2523.44	6604.29

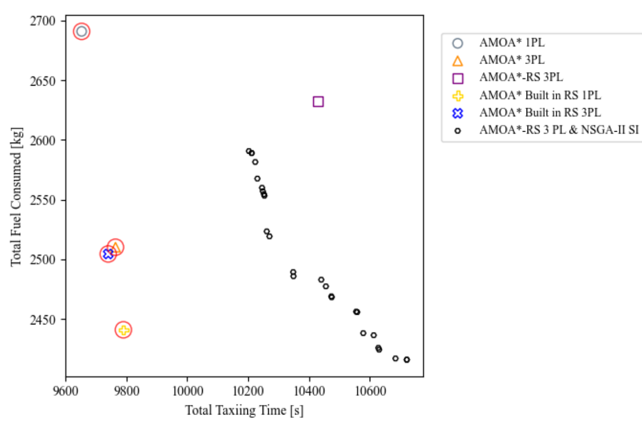

Fig. 5. Pareto front for different AMOA* algorithms optimising for case 3.

IV. Conclusions

The introduction of runway scheduling into the AGMP through algorithm AMOA*-RS perfectly depicts the loss in optimization previously believed as achieved by algorithms that overlooked this condition. Algorithms AMOA*-RS 3 PL & NSGA-II SI and AMOA*-RS 3 PL & NSGA-II PI excel at optimizing for AGMP considering runway scheduling.

By comparing these algorithms with algorithms that overlook runway scheduling and spacing, the importance of correct modelling for accurate objective optimization quantification is outlined. These genetic algorithms favorably generate a set of pareto optimal solutions that surpass AMOA*-RS in terms of solution quality. When optimizing for long periods, infeasible results further deviate from feasible results since infeasible fuel and time savings add up, further increasing the importance of correct modelling. In scenarios which AGMP algorithms solve without violating any spacing restrictions, the implementation of a Genetic Algorithm will result in an unnecessary expenditure of computational resources.

Broader search spaces, which result in larger variability, are present in the Speed Profile Selection Index (AMOA-RS 3PL & NSGA-II SI) variant of the Genetic algorithm, which generates larger pareto fronts. Pareto fronts might complement each other in the search for non-dominated scenario solutions. Different techniques used in different algorithms pose new challenges. Built-In models incorporate runway scheduling with low computational complexity, however, if left unbounded in time, the solution is spilled in time, becoming infeasible in practice once again, since future aircraft slot allocated times at the runway or at the gate will be impossible to be met. In future works slot allocated times for runway and gates will be explored and added into the algorithm to ensure that all solutions are compliant with these time restrictions.

References

[1] J. a D. Atkin, E. K. Burke, and S. Ravizza, “The airport ground movement problem: Past and current research and future directions,” Research in Air Transportation, 2010.

[2] E. W. Dijkstra, “A note on two problems in connexion with graphs,” Numer Math (Heidelb), vol. 1, no. 1, 1959, doi: 10.1007/BF01386390.

[3] J. W. Smeltink, M. J. Soomer, P. R. de Waal, and R. D. van der Mei, “An Optimisation Model for Airport Taxi Scheduling,” in INFORMS Annual Meeting, 2004.

[4] G. Clare and A. G. Richards, “Optimization of taxiway routing and runway scheduling,” IEEE Transactions on Intelligent Transportation Systems, vol. 12, no. 4, 2011, doi: 10.1109/TITS.2011.2131650.

[5] R. Deau, J. B. Gotteland, and N. Durand, “Airport surface management and runways scheduling,” in Proceedings of the 8th USA/Europe Air Traffic Management Research and Development Seminar, ATM 2009, 2009.

[6] S. Ravizza, J. A. D. Atkin, and E. K. Burke, “A more realistic approach for airport ground movement optimisation with stand holding,” Journal of Scheduling, vol. 17, no. 5, 2014, doi: 10.1007/s10951-013-0323-3.

[7] J. Chen et al., “Toward a more realistic, cost-effective, and greener ground movement through active routing: A multiobjective shortest path approach,” IEEE Transactions on Intelligent Transportation Systems, vol. 17, no. 12, 2016, doi: 10.1109/TITS.2016.2587619.

[8] C. Evertse and H. G. Visser, “Real-time airport surface movement planning: Minimizing aircraft emissions,” Transp Res Part C Emerg Technol, vol. 79, 2017, doi: 10.1016/j.trc.2017.03.018.

[9] A. E. I. Brownlee, M. Weiszer, J. Chen, S. Ravizza, J. R. Woodward, and E. K. Burke, “A fuzzy approach to addressing uncertainty in Airport Ground Movement optimization,” Transp Res Part C Emerg Technol, vol. 92, 2018, doi: 10.1016/j.trc.2018.04.020.

[10] P. E. Hart, N. J. Nilsson, and B. Raphael, “A Formal Basis for the Heuristic Determination of Minimum Cost Paths,” IEEE Transactions on Systems Science and Cybernetics, vol. 4, no. 2, 1968, doi: 10.1109/TSSC.1968.300136.

[11] L. Mandow and J. L. P. De La Cruz, “Multiobjective A* search with consistent heuristics,” Journal of the ACM, vol. 57, no. 5, 2010, doi: 10.1145/1754399.1754400.

[12] R. Mori, “Aircraft ground-taxiing model for congested airport using cellular automata,” IEEE Transactions on Intelligent Transportation Systems, vol. 14, no. 1, 2013, doi: 10.1109/TITS.2012.2208188.

[13] Y. Jiang, Z. Liao, and H. Zhang, “A collaborative optimization model for ground taxi based on aircraft priority,” Math Probl Eng, vol. 2013, 2013, doi: 10.1155/2013/854364.

[14] B. Pesic, N. Durand, and J.-M. Alliot, “Aircraft Ground Traffic Optimisation Using a Genetic Algorithm,” in Proceedings of the Genetic and Evolutionary Computation Conference, 2001.

[15] Civil Aviation Authority, Manual of Air Traffic Services – Part 1, vol. CAP 493. Gatwick Airport South, West Sussex: The Stationery Office, 2015.

[16] M. Weiszer, E. K. Burke, and J. Chen, “Multi-objective routing and scheduling for airport ground movement,” Transp Res Part C Emerg Technol, vol. 119, 2020, doi: 10.1016/j.trc.2020.102734.

[17] M. Weiszer, J. Chen, and P. Stewart, “A real-time Active Routing approach via a database for airport surface movement,” Transp Res Part C Emerg Technol, vol. 58, 2015, doi: 10.1016/j.trc.2015.07.011.

[18] H. Khadilkar and H. Balakrishnan, “Estimation of aircraft taxi fuel burn using flight data recorder archives,” Transp Res D Transp Environ, vol. 17, no. 7, 2012, doi: 10.1016/j.trd.2012.06.005.

[19] T. Nikoleris, G. Gupta, and M. Kistler, “Detailed estimation of fuel consumption and emissions during aircraft taxi operations at Dallas/Fort Worth International Airport,” Transp Res D Transp Environ, vol. 16, no. 4, 2011, doi: 10.1016/j.trd.2011.01.007.

[20] J. Chen, M. Weiszer, P. Stewart, and M. Shabani, “Toward a More Realistic, Cost-Effective, and Greener Ground Movement Through Active Routing - Part I: Optimal Speed Profile Generation,” IEEE Transactions on Intelligent Transportation Systems, vol. 17, no. 5, 2016, doi: 10.1109/TITS.2015.2477350.

[21] M. Weiszer, J. Chen, and G. Locatelli, “An integrated optimization approach to airport ground operations to foster sustainability in the aviation sector,” Appl Energy, vol. 157, 2015, doi: 10.1016/j.apenergy.2015.04.039.

[22] J. Chen and M. Mahfouf, “A population adaptive based immune algorithm for solving multi-objective optimization problems,” in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2006. doi: 10.1007/11823940_22.