

EPiC Series in Computing

Volume 114, 2026, Pages 191–205

Proceedings of the Workshop on Mathematical Modeling and Scientific Computing



Preference Incorporation into Generalized Multi-Objective Island Model for Flight Control System Clearances

Patrick Piprek and David Schwalb

Airbus Defence and Space GmbH, Rechliner Straße, 85077 Manching, Germany
patrick.piprek@airbus.com, david.schwalb@airbus.com

Abstract

Before release-to-service of an A/C, FM assessments over the entire flight envelope, including different failure cases, pilot inputs, and configurations, are performed to ensure its safe operation. These assessments must cover all operationally relevant scenarios and no critical ones may be overlooked. In this context, global optimisation strategies have proven effective because they enable the identification of multiple problematic cases and areas within a vast input parameter space and for multiple objectives. These objectives range from AoA and AoS exceedance to control-surface saturation times. Specifically, NSGA-based multi-objective optimisation algorithms analyse these objectives using a domination approach, to ensure optimality and the crowding-distance metric to preserve diversity. For FM assessments, some criteria, like exceedances of AoA limits, are considered more critical than, for example, longer control surface saturation times. Accordingly, this paper presents a method for incorporating so-called "secondary" criteria in the optimisation algorithm, which are not treated as primary optimisation objectives, i.e., objectives governing Pareto dominance, but instead influence solution diversity through modification of the crowding distance. The proposed framework is demonstrated using an A/C-clearance scenario to highlight its benefits.

Nomenclature

A/C	Aircraft
ADS	Air Data System
AoA	Angle of Attack
AoS	Angle of Sideslip
DEA	Differential Evolution Algorithm
FCS	Flight Control System
FM	Flight Mechanics
GWOA	Grey Wolf Optimization Algorithm
HPC	High Performance Computing
HQ	Handling Qualities
HV	Hypervolume

V. Turova, A. Kovtanyuk, A. Maslovskaya and J. Zimmer (eds.), MMSC 2026 (EPiC Series in Computing, vol. 114), pp. 191–205

MPI	Message Passing Interface
NSGA	Non-dominated Sorting Genetic Algorithm
SLURM	Simple Linux Utility for Resource Management (Workload Manager)
WOA	Whale Optimization Algorithm

CD_{adapt}	[−]	Adapted Crowding Distance
CD_{orig}	[−]	Original Crowding Distance
PR	[−]	Preference Value
w_n	[−]	Preference Weight in n-th generation

1 Introduction

This paper presents an extension of the author's previous work on generalised island models for FCS clearances [6,20,21] aimed at incorporating preference information, which may e.g., be related to HQ requirements as defined in [25], into the framework. The objective of incorporating preference information is to identify worst-case scenarios that are in desired domains of the solution space, i.e., are of greatest operational relevance for FCS clearances. The foundation of the island model [6,20,21] are multi-objective optimisation algorithms of the NSGA-type [3–5]. Further algorithms, like the DEA [7], WOA [18], and GWOA [19], are also applicable.

These algorithms ensure reaching a good spread of the solutions on the Pareto front, as well as the parameter space, by using the crowding distance metric [4]. This metric is calculated as a distance between different solutions (in the objective and/or parameter space) and is used to sort the solutions within the different fronts. The aim is to retain solutions in sparsely populated areas of the front (i.e., have a large crowding distance) to ensure a complete exploration of the Pareto front and its associated parameter domain [3–5]. It is used as a first entry point to incorporate preference in the proposed framework update.

In general, the calculation of the crowding distance itself has already been improved from its original definition in [4] by adapting the centring point in [2], as well as dealing with cases where individuals have the same fitness [8]. Moreover, the introduction of a weighted sum and neighbourhood mutation operator [11] as well as intra-front crowding distance calculation have successfully been applied [10]. Pruning methods re-calculating the crowding distance after eliminating the minimum one [17] or pruning a set of members with the smallest crowding distances in a bi-objective setup [13] have also been developed. Additionally, a minimum spanning tree was introduced in [15] to improve diversity preservation of Pareto problems. While all of these approaches tackle some of the drawbacks in the original works [3–5], they mainly try to ensure spread of the solutions on the Pareto front. However, a complete spread across the Pareto front may not always be necessary, at least not globally, in FCS clearances, specifically when considering preferences and operational relevance.

Consequently, this study first addresses the introduction of preferences in the crowding distance calculation, which may be based, for example, on certification requirements [25] or operational relevance. Reviews and studies related to preference in optimisation were done in [14,22]. A weighted sum approach was introduced in [16] and has proven viable. In addition to the adaptation of the crowding distance itself, the approach proposed in this study also utilizes the power of the already developed island model framework [6,20,21], which allows to exchange information between differently parametrised algorithms and islands. Consequently, solutions from algorithms running with and without preference can be exchanged, facilitating

the convergence to a desired solution space, while still exploring all critical worst-case scenarios required for FCS clearances.

2 Theory

This section summarises the theory of incorporating preference within the existing island-model-framework [6, 20, 21]. It specifically introduces the crowding distance adaptation based on so-called "secondary" objectives, which describe the preference to be incorporated in the algorithm.

2.1 Optimisation problem

The following multi-objective optimisation problem is solved in FCS FM assessments [6, 20, 21]:

$$\begin{aligned} \underset{\mathbf{z}}{\text{minimise}} \quad & \mathbf{J}\left(\mathbf{x}\left(\mathbf{z}\right),\mathbf{z}\right), && \mathbf{J}:\mathbb{R}^m\times\mathbb{R}^n\to\mathbb{R}^o \\ \text{subject to} \quad & \mathbf{z}_{\text{lb}}\le\mathbf{z}\le\mathbf{z}_{\text{ub}}, && \mathbf{z}\in\mathbb{R}^n, \\ & \mathbf{c}_{\text{lb}}\le\mathbf{c}\left(\mathbf{x}\left(\mathbf{z}\right),\mathbf{z}\right)\le\mathbf{c}_{\text{ub}}, && \mathbf{c}:\mathbb{R}^m\times\mathbb{R}^n\to\mathbb{R}^p, \\ & \boldsymbol{\psi}_{\text{eb}}=\boldsymbol{\psi}\left(\mathbf{x}\left(\mathbf{z}\right),\mathbf{z}\right), && \boldsymbol{\psi}:\mathbb{R}^m\times\mathbb{R}^n\to\mathbb{R}^q, \end{aligned} \tag{1}$$

where $\mathbf{x}\left(\mathbf{z}\right)\in\mathbb{R}^m$ is the state trajectory defined as the solution to the parametric initial value problem depending on the optimization variable vector $\mathbf{z}\in\mathbb{R}^n$

$$\dot{\mathbf{x}}(t;\mathbf{z})=\mathbf{f}(\mathbf{x}(t;\mathbf{z}),\mathbf{z}), \quad \mathbf{x}(0;\mathbf{z})=\mathbf{x}_0(\mathbf{z}), \quad t\in[0,t_f]. \tag{2}$$

In (1) and (2), $\mathbf{z}$ are the optimisation variables, while the objective functions, equality, and inequality constraints are defined by $\mathbf{J}(\cdot)$, $\boldsymbol{\psi}(\cdot)$, and $\mathbf{c}(\cdot)$, respectively. The state dynamics are given by the nonlinear mapping functions $\mathbf{f}(\cdot)$ for the state trajectory $\mathbf{x}$ in (2), which can be solved via standard simulation methods until the final time t_f in the island model [1, 20]. The indices "lb", "ub", and "eb" denote the lower, upper, and equality bounds, respectively.

A practical challenge associated with the problem specification in (1) is its multi-objective form. Such problems are difficult to solve as they require Pareto optimality [12]. They are, however, of high relevance in FCS FM assessments because there are generally multiple objectives to be considered based on the different requirements to be fulfilled [24]. For the solution of such problems previous studies [6, 20, 21] introduced a framework based on an island model, which is extended in this study and briefly discussed in Section 2.2.

2.2 Island model and genetic algorithm

To solve the problem specified in (1), evolutionary algorithms are often suitable. The following paragraphs give an overview on the existing work [6, 20, 21] extended in this study. The basic workflow of an evolutionary algorithm (e.g., NSGA2 [4]) is visualised in Fig. 1: Based on initial individuals of the optimisation variables, created e.g., by random sampling, the state dynamics and constraints are evaluated to retrieve the initial results. Then, non-dominated sorting and crowding-distance evaluation are conducted to sort the individuals from best to worst according to the specified objectives. Afterward, "selection", "crossover", and "mutation" are applied to generate new offspring individuals. This procedure is iterated until a user-defined stopping criterion, like the maximum number of generations, is met.

The island-model framework [6, 20, 21], described in Fig. 2, is implemented in Java™, with interfaces to SQLite™ and other programming and scripting languages such as Fortran,

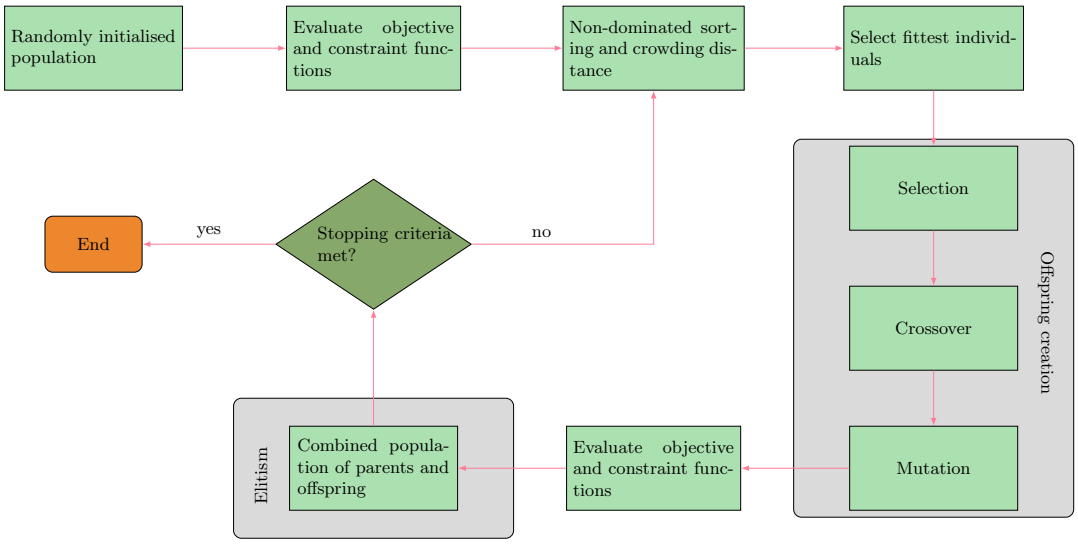


Figure 1: Flowchart of a generic evolutionary algorithm with clustering applicable in the islands of the generalised island model (graphic taken from [21]).

Python™, or Matlab® to execute external code and software. The algorithm has each island running in parallel, but also the evaluation of the objective/constraint functions is performed in parallel: This parallelisation is provided by Open MPI (cyan box) [9]. The allocation of the cores on the HPC nodes is handled by SLURM [27].

A significant benefit of the framework in Fig. 2 is its adaptability and versatility: It can easily handle different types of algorithms being executed and also different parameterizations of the same algorithm. This enables a thorough analysis of the FM problem and a high confidence in finding the worst cases. More specifically, a benefit is also within the migration between different islands, which facilitates the exchange of information regarding currently identified critical domains of each island, thus facilitating the worst-case search. These benefits of the framework are used to directly incorporate preference as introduced in Section 2.3.

2.3 Incorporation of preference

As indicated in Section 1, the preference adaptation is initially introduced in the crowding distance metric by means of a weighted sum approach with linear weighting [16]. The values for this preference adaptation are calculated based on the normalisation introduced in [20]. As the crowding distance, CD, is always positive, the preference adaptation, PR, which is also positive due to the normalisation, is subtracted from the crowding distance calculated up until this point. This is due to the fact that smaller crowding distances are considered less optimal in the sorting of a front [4]. Therefore, the following crowding distance calculation is implemented:

$$\mathrm{CD}_{\mathrm{adapt}} = (1 - w_n) \cdot \mathrm{CD}_{\mathrm{orig}} - w_n \cdot \sum_i \mathrm{PR}_i \tag{3}$$

It should be taken into account that the weight, $w_n \in [0, 1]$, in (3) can either be constant or change e.g., over the number of generations. Some common strategies for this purpose include linear or exponential change. Specifically, strategies that change the weight have the benefit

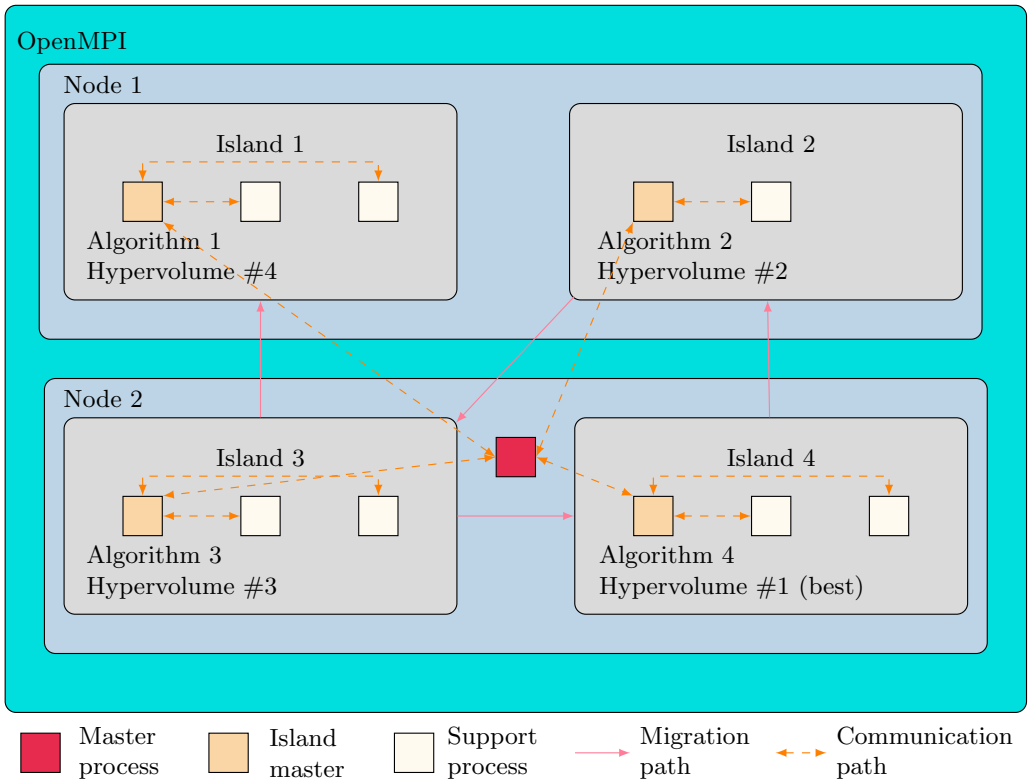
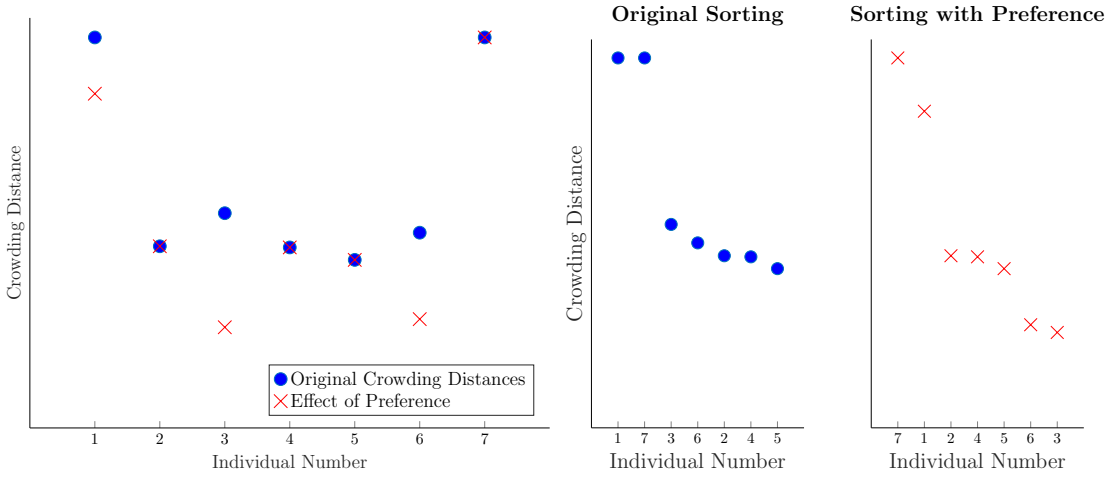


Figure 2: High-level illustration of the island model implementation including different parallelisation layers (graphic taken from [20, 21]).

that the influence of the preference can be changed as the algorithm progresses: for instance, a smaller weight in the initial generations puts a focus on exploring the full Pareto front, while a larger weight in later stages then focuses on solutions in preferred regions. Take into account that the choice of a proper weight in (3) has significant effect on the representation of the Pareto front, due to the adaptation of the crowding distance, and must therefore be done with care, specifically if a proper representation of the Pareto front is desired. Different values should be checked, which is one benefit of the island model introduced in Section 2.2 as it allows the islands to run with different weights and weighting schemes.

The effect of (3) is also visualized in Fig. 3a, which shows an exemplary reduction of the crowding distances for individuals (numbers: 1, 3, and 6) based on different-magnitude preference values. Here, this crowding distance reduction can directly be seen in the sorting of individuals within the front, which is shown by Fig. 3b: Here, in the left plot without preference, individual 3 has the third largest crowding distance, i.e., is sorted into the third position. However, considering the preference, the individual is in last place. If the front were not considered as a whole, individual 3 would not be selected for subsequent generations [3–5]. Similarly, individual 6 is less likely to be chosen than e.g., individual 4 with the effect of preference. Thus, preference incorporation provides a mechanism to balance exploration of the Pareto front, given by the crowding distance, with the desire to find critical cases in preferred domains, which can be shaped by the definition of suitable preference representation. Consequently, preference



(a) Effect of the preference incorporation on the crowding distance value.

(b) Effect of the preference incorporation in the sorting of the individuals within a front.

Figure 3: Effects of preference incorporation on crowding distance calculation and sorting of individuals inside the same front.

acts as a "lexicographic selection pressure" gradually pushing the optimisation towards desired regions, however never becoming the primary objective and thus, resulting in a completely different solution space. This becomes particularly clear once all preference values in the population reach optimality, i.e., their value is "0" [20], for which the original behaviour of the crowding distance operator, in a weighted form, is restored by (3).

Some considerations regarding the manipulation of the crowding distance to incorporate preference for practical applications are given in the following: For one-dimensional objective functions, an ε-dominance [5] approach or similar, has to be used as otherwise each front will generally only contain one individual, and thus, the crowding distance will have no effect on the selection. Additionally, in standard NSGA algorithms the crowding distance of boundary solutions (i.e., the ones currently sitting at the edges of the front) are assigned a large crowding distance for which subtracting the preference has a limited effect (see Fig. 3). This is a desired effect in this study as the fronts should still be explored, but the behaviour can be updated dependent on the defined scope of the application.

Besides manipulating the crowding distance, as introduced in (3), the island model framework introduced in Section 2.2 also allows for multiple algorithms to be run in parallel. Although this functionality is generally intended for different types of e.g., NSGA-like algorithms, it also allows to execute the same algorithm with different hyperparameters. This means that it is possible to run islands with and without preference incorporation (e.g., by setting a constant weight of $w_n = 0$ in one island). This leads to islands evolving differently and consequently finding different critical regions and scenarios, which is a desirable feature for FCS clearance assessments. In addition, the island model allows for the exchange of information between the algorithms via migration (see Fig. 2). The migration direction can be controlled using a performance metric like the HV [23]. In this context, this allows a further means to incorporate preference: While the HV is normally calculated based on the current Pareto front, which is made up of the currently best objective values of the different population members, it can also be extended to include preference. For this purpose, no modifications are required to the cal-

culation of the HV [20,23,26] as the preference values are normalised just like objectives in the proposed framework [20] and therefore can be treated as such. Consequently, preference can be directly included (or even also be the sole input) to calculate the HV. By this, even if an algorithm does not actually consider the preference in the crowding distance, it can still share and receive information based on its current "preference optimality" via the migration and the associated HV.

3 Examples

The following sections display the proposed framework of Section 2 with the Himmelblau test function example (Section 3.1) as well as an FCS clearance (Section 3.2).

3.1 Himmelblau function

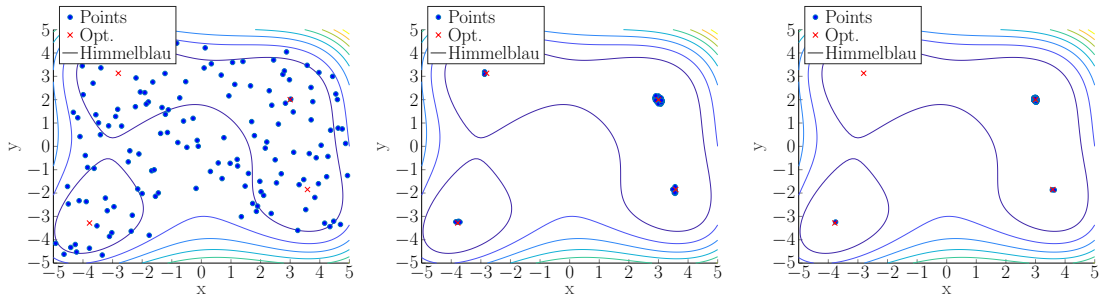
As a baseline example, this study uses the Himmelblau function defined as follows:

$$J(x, y) = (x^2 + y - 11)^2 + (x + y^2 - 7)^2 \tag{4}$$

The function in (4) is defined by a two-dimensional parameter space, $x, y \in [-5, 5]$, and has four distinct optima in that domain.

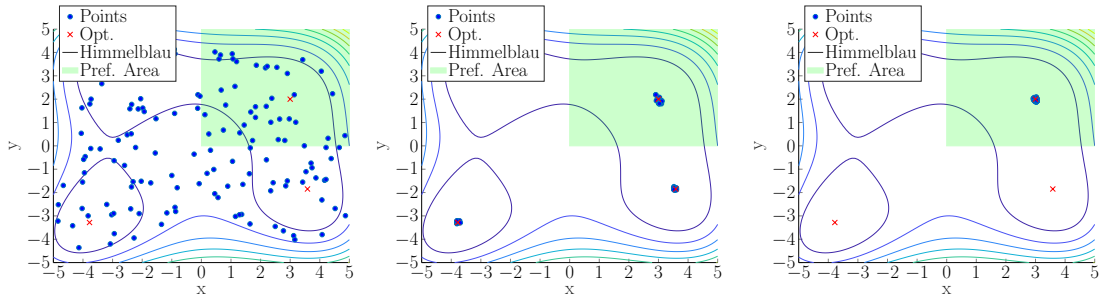
To solve the problem given by (4), this study uses the omni-optimizer with ε-dominance [5] as it ensures a good spread of solutions in both objective as well as parameter space (one island and 100 generations with 100 population members each are used). The solution without preference incorporation is displayed in Fig. 4. It is apparent that after ten generations (see Fig. 4b) all optima are fairly well explored, with most population members converging toward the top-right optimum, which also explains the subsequent evolution in Fig. 4c. This behaviour can be attributed to the random initialisation in Fig. 4a, which has one population member almost at this top-right optimal point from the beginning. Still, it is apparent that the algorithm is able to explore the whole domain and find all the optimal, and therefore critical, regions, which is specifically desired in FCS clearance scenarios.

Figure 5 now displays an example with the same optimizer setup, but considering preference incorporation according to (3). The weight is chosen constant with a value of $w_n = 0.9$, which is intentionally selected large to highlight the effect of the preference in a fast manner, and



(a) Generation 0. (b) Generation 10. (c) Generation 15.

Figure 4: Himmelblau function solved by omni-optimizer over different generations without preference incorporation (blue dots: current population members; red crosses: optimal points).



(a) Generation 0. (b) Generation 10. (c) Generation 15.

Figure 5: Himmelblau function solved by omni-optimizer over different generations with preference incorporation preferring solution with positive x and y component (green area) and constant weight of $w_n = 0.9$ (blue dots: current population members; red crosses: optimal points).

the preference criterion is to find solutions in the first quadrant (i.e., with positive x and y components; indicated as a shaded green area in the figures). In general, the algorithm behaves very similarly to the scenario without preference (Fig. 4) between the initial (Fig. 5a) and the tenth generation (Fig. 5b), i.e., all optima are properly explored by the Pareto dominance and crowding distance concepts. However, as displayed in Fig. 5c, afterward the effect of the preference becomes visible, and only solutions in the first quadrant remain in the population of generation 15. This behaviour is exactly as desired: The algorithm should still explore all potential critical regions first and only then be driven by the preference. Thus, this test function example directly displays the viability and power of the proposed approach, which is extended in the next section to an FCS clearance assessment.

3.2 FCS clearance

This section displays an example of the application of "secondary" criteria for an A/C FM clearance assessment: The task is the assessment of reduced ADS accuracy within an otherwise failure-free A/C under severe turbulence. This A/C should behave carefree [24], for which the pilot inputs are modelled based on a set of pre-defined manoeuvres. Further optimisation variables are the envelope point, the air brake position, the configuration, like external stores, the throttle position, and the mass direction estimation error (i.e., higher or lower estimated mass properties compared to actual values). For the dynamics, a fully nonlinear rigid-body simulation of a high-dynamic A/C is used.

The optimisation process employs the omni-optimisation algorithm [5] in three islands, each evaluating 100 generations with 250 population members (first migration at generation 25). The optimisation finished in ≈ 90 minutes using 151 cores. The primary objectives are the maximum AoA and maximum absolute AoS, while the secondary objectives, i.e., the preferences, are the saturation times of the control surfaces in the pitch axis, q_{sat}, and yaw axis, r_{sat}. These are classic criteria related to both stability and handling of the A/C. Three islands are configured: The first island uses an adaptive preference weight in (3) growing exponentially from $w_0 = 0.1$ to $w_{100} = 0.9$, indicating that initially the standard crowding distance metric should be preferred, while, towards the final generations, the preference should have more effect. The second island has a fixed weight of $w_0 = w_{100} = 0.5$, while the third island also has a fixed

weight of $w_0 = w_{100} = 0$. The fixed weights indicate that for the second island crowding-distance preservation and preference incorporation are weighted equally, while the third island discards preference. It only receives preference-related information from the other islands via migration. For the migration, a HV calculated based on the "secondary" objectives, i.e., only the preference, is used, thereby ensuring exchange of preference-related information, even if the island itself does not consider them.

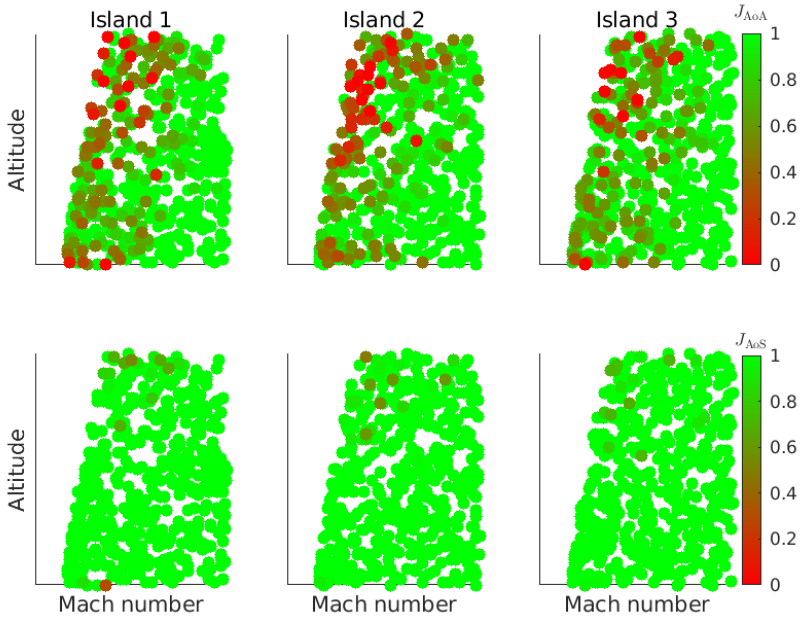
The results are presented as envelope plots of the objectives for the initial generation in Fig. 6. The top row in the primary objectives (see Fig. 6a) relates to the normalised AoA objective for each island (with 0 being optimal [20]), while the bottom row is the normalised AoS objective. For the secondary objectives (see Fig. 6b), the top row displays the normalised pitch axis saturation time, while the bottom row depicts the normalised yaw axis saturation time. It can be seen that both the AoA objective as well as the yaw axis saturation time already indicate potentially critical regions based on this random initialisation.

For generation 15, which is still before the first migration, the results are displayed in Fig. 7 (primary objectives: Fig. 7a; secondary objectives: Fig. 7b): It can be seen that island 3, which does not focus on preference, has found the largest amount of potentially critical AoA and AoS scenarios, while it only sparsely explores the preference objectives. On the other hand, island 2, which uses a constant preference weight, has significantly explored the yaw saturation preference, also already with critical seeds in the high altitude for pitch saturation. Island 1, which still has a comparably small preference weight at this generation has explored both primary as well as secondary objectives quite well.

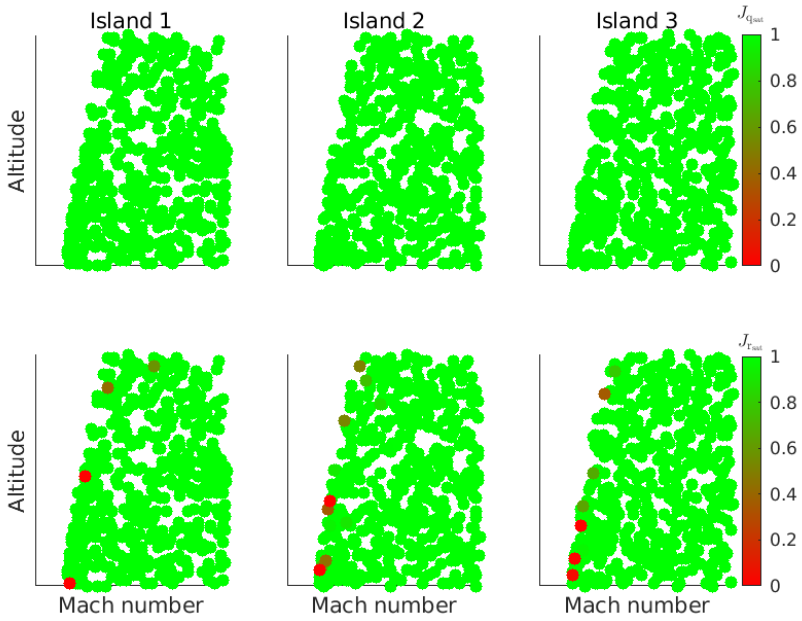
Furthermore, results for generation 25, which is right after the first migration, are depicted in Fig. 8 (primary objectives: Fig. 8a; secondary objectives: Fig. 8b): Here, the evolution follows the trends initially indicated in Fig. 7, i.e., island 3 shows the most primary objective violations, while island 2 does the same for secondary objectives. Seeds for potential secondary objective violations have also been exchanged between islands 2 and 3.

Finally, results for generation 100, which is the final one, are depicted in Fig. 9 (primary objectives: Fig. 9a; secondary objectives: Fig. 9b): Here, is is apparent that island 3 provided the most extensive exploration of the primary objectives, while it also received seeds of the secondary objectives to explore them as well. On the other hand, island 2 explored the secondary objectives very well but also found a reasonable amount of critical cases on the primary objectives. Island 1 seems to provide the best trade-off between the two more extreme algorithm parametrizations, i.e., it offers the best trade-off between primary and secondary objective exploration.

Overall, the FCS clearance example, introduced in the previous paragraphs, depicted that the proposed preference incorporation into the island model supported exploration of both primary and secondary objectives. However, the secondary objectives never dominated the optimisation process, as e.g., seen in the pitch axis saturations found by island 2, which were never sufficiently critical for the algorithm to investigate them further based on dominance. Specifically, the versatility of the island model allowing differently parametrised algorithms to be executed concurrently is beneficial in this setup, as some islands can concentrate on the primary objectives, while others do the same for the secondary ones. Consequently, a setup can be created that achieves the desired trade-off between primary and secondary objectives, which is desired in the scope of FCS clearances to reach operationally relevant critical scenarios.

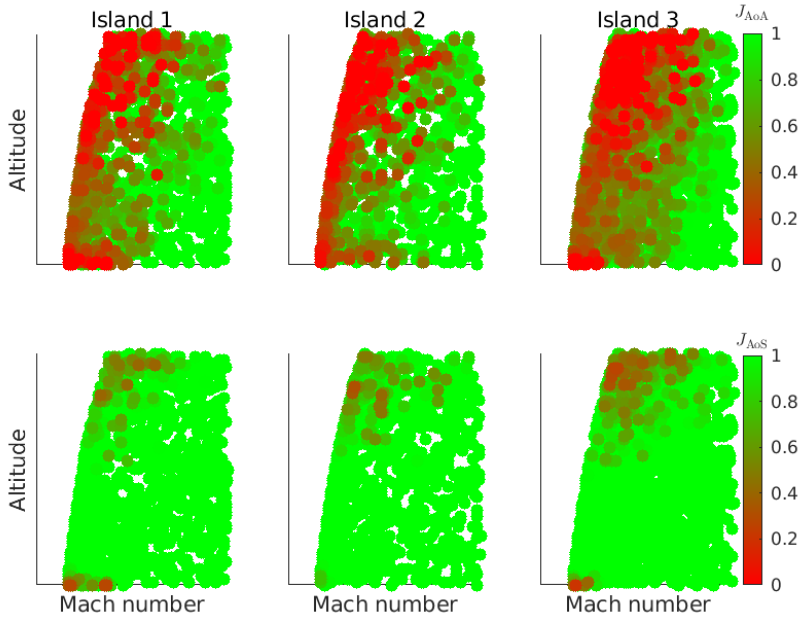


(a) Primary objectives.

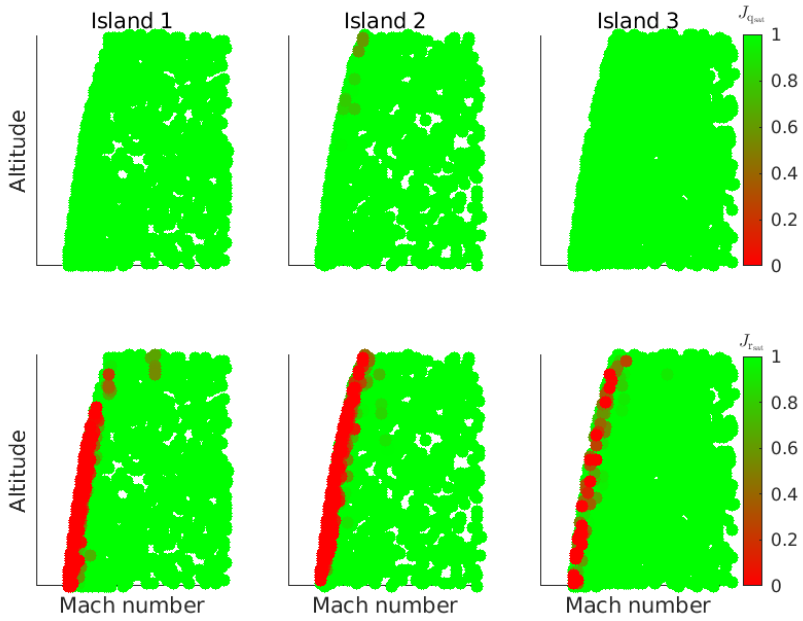


(b) Secondary objectives.

Figure 6: Initial Generation 0 of FCS clearance assessment case with three islands and different preference incorporation (Island 1: exponential preference weight; Island 2: constant weight; Island 3: No preference).

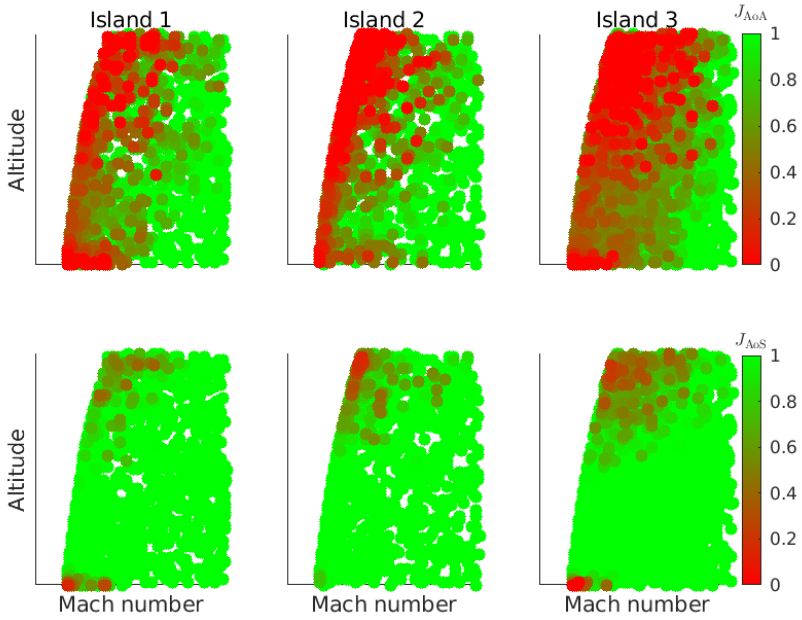


(a) Primary objectives.

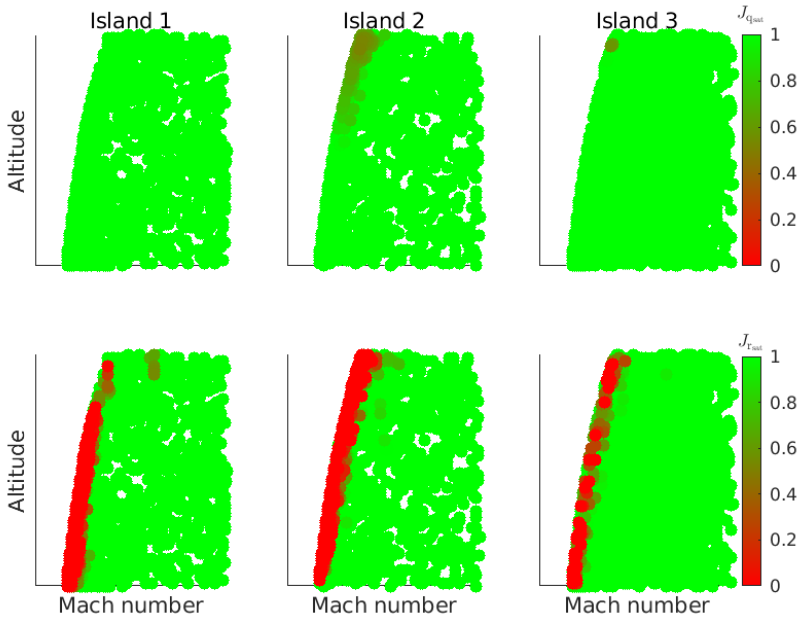


(b) Secondary objectives.

Figure 7: Evaluated Individuals up to and including generation 15 (before first migration) of FCS clearance assessment case with three islands and different preference incorporation (Island 1: exponential preference weight; Island 2: constant weight; Island 3: No preference).

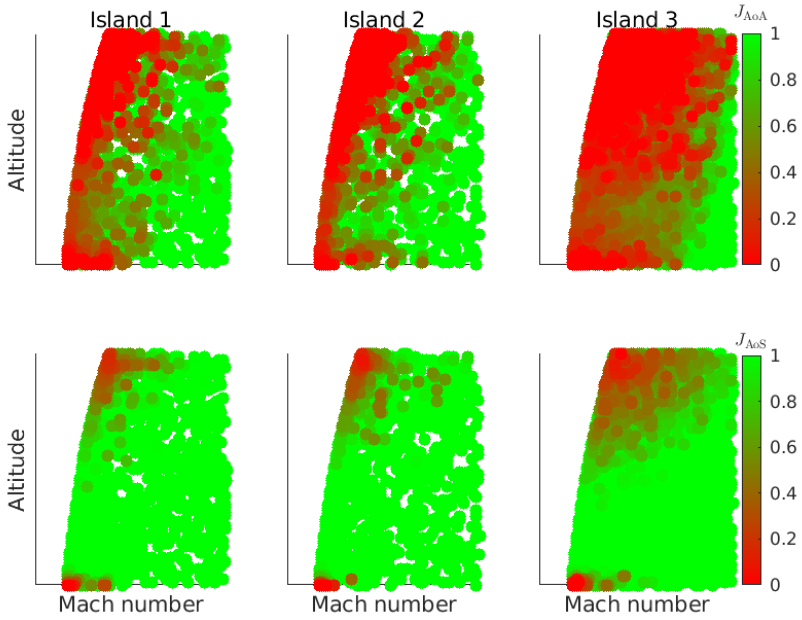


(a) Primary objectives.

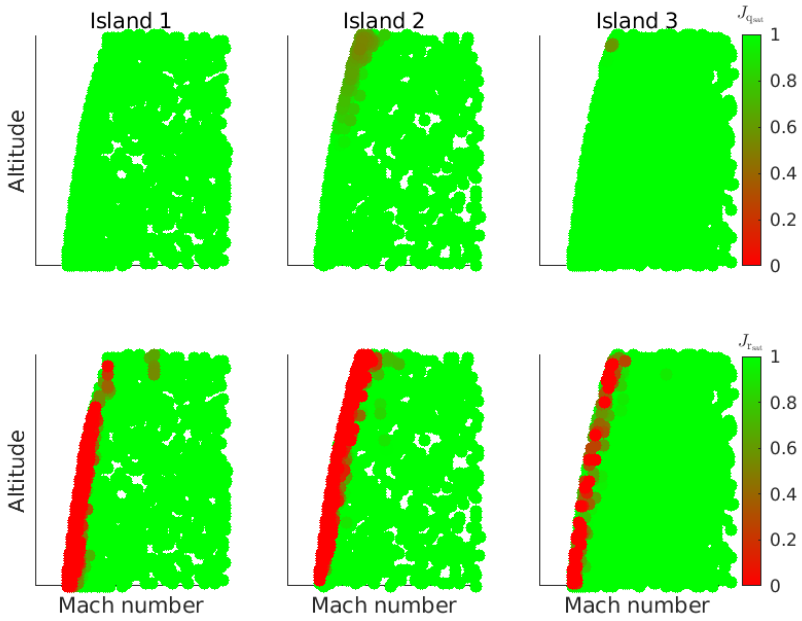


(b) Secondary objectives.

Figure 8: Evaluated Individuals up to and including generation 25 (directly after first migration) of FCS clearance assessment case with three islands and different preference incorporation (Island 1: exponential preference weight; Island 2: constant weight; Island 3: No preference).



(a) Primary objectives.



(b) Secondary objectives.

Figure 9: Evaluated Individuals up to and including generation 100 (final generation) of FCS clearance assessment case with three islands and different preference incorporation (Island 1: exponential preference weight; Island 2: constant weight; Island 3: No preference).

4 Conclusion

This paper presents a framework for FCS clearance using an island model approach with an extension for preference incorporation. The method is tailored to explore all critical regions, while allowing the optimisation process to focus on scenarios in preferred domains of the objective space, specifically also in later stages of the algorithm. By this, the proposed framework thrives in A/C clearances, because identifying operationally relevant critical scenarios is essential to reach operational relevance, which can be accomplished by preference. The framework offers great flexibility in terms of the problem structure and the mechanisms through which preference can be incorporated.

Future work may expand on the available weighting schemes for the preference as well as introducing adaptive measures to optimally balance exploration by the standard crowding distance operator with the exploitation of preferred solutions via the secondary objectives. Here, it may be worth to check using the preference as a third metric (similar to an ε-dominance approach) rather than as a weight subtracted from the crowding distance or using the preference only in crowded neighbourhoods of the front. Furthermore, it may be worth to explore the option of choosing a fixed set of individuals based on their crowding distance alone and the other set based on the weighted crowding distance. Finally, instead of using the secondary objectives to calculate the HV, the standard HV based on primary objectives could be used to preserve the performance of the island model, while the secondary objectives would then only be applied in the selection process of the migration within the island, i.e., the acceptance of migrated individuals into the island's population.

References

[1] J. T. Betts. *Practical methods for optimal control and estimation using nonlinear programming*. Advances in design and control. Society for Industrial and Applied Mathematics (SIAM 3600 Market Street Floor 6 Philadelphia PA 19104), Philadelphia, Pa., 2nd ed. edition, 2010.

[2] X. Chu and X. Yu. Improved Crowding Distance for NSGA-II, 2018.

[3] K. Deb and H. Jain. An Evolutionary Many-Objective Optimization Algorithm Using Reference-point Based Non-dominated Sorting Approach, Part I: Solving Problems with Box Constraints. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.

[4] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 18(4):577–601, 2013.

[5] K. Deb and S. Tiwari. Omni-optimizer: A procedure for single and multi-objective optimization. In C. A. Coello Coello, A. Hernández Aguirre, and E. Zitzler, editors, *Evolutionary Multi-Criterion Optimization*, pages 47–61, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

[6] P. M. Dias, P. Piprek, and D. Schwalb. Optimization-based flight control system clearance philosophy for fighter aircraft. In *AIAA SCITECH 2023 Forum*. American Institute of Aeronautics and Astronautics, January 2023.

[7] V. Feoktistov. *Differential Evolution: In Search of Solutions*. Springer, New York, 2006.

[8] F. Fortin and M. Parizeau. Revisiting the NSGA-II crowding-distance computation. In *Proceedings of the 15th Annual Conference on Genetic and Evolutionary Computation*, GECCO '13, page 623–630, New York, NY, USA, 2013. Association for Computing Machinery.

[9] R. L. Graham, T. S. Woodall, and J. M. Squyres. Open MPI: A Flexible High Performance MPI. In *Parallel Processing and Applied Mathematics*, pages 228–239. Springer Berlin Heidelberg, 2006.

[10] M. Javadi and S. Mostaghim. Analysis of inter and intra-front operations in multi-modal multi-objective optimization problems. *Natural Computing*, 22(2):341–356, September 2022.

[11] M. Javadi, H. Zille, and S. Mostaghim. The effects of crowding distance and mutation in multi-modal and multi-objective optimization problems. In António Gaspar-Cunha, Jacques Periaux, Kyriakos C. Giannakoglou, Nicolas R. Gauger, Domenico Quagliarella, and David Greiner, editors, *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, pages 115–130, Cham, 2021. Springer International Publishing.

[12] J. D. Knowles. *Local-search and hybrid evolutionary algorithms for Pareto optimization.* Doctoral thesis, University of Reading, Reading, 2002.

[13] S. Kukkonen and K. Deb. Improved Pruning of Non-Dominated Solutions Based on Crowding Distance for Bi-Objective Optimization Problems. In *2006 IEEE International Conference on Evolutionary Computation*. IEEE, 2006.

[14] K. Li, M. Liao, K. Deb, G. Min, and X. Yao. Does Preference Always Help? A Holistic Study on Preference-Based Evolutionary Multiobjective Optimization Using Reference Points. *IEEE Transactions on Evolutionary Computation*, 24(6):1078–1096, December 2020.

[15] M. Li, J. Zheng, and J. Wu. *Improving NSGA-II Algorithm Based on Minimum Spanning Tree*, page 170–179. Springer Berlin Heidelberg, 2008.

[16] G. Liu, G. Wu, T. Zheng, and Q. Ling. Integrating preference based weighted sum into evolutionary multi-objective optimization. In *2011 Seventh International Conference on Natural Computation*, page 1251–1255. IEEE, July 2011.

[17] J. Liu and X. Chen. An Improved NSGA-II Algorithm Based on Crowding Distance Elimination Strategy. *International Journal of Computational Intelligence Systems*, 12(2):513, 2019.

[18] S. Mirjalili and A. Lewis. The whale optimization algorithm. *Advances in Engineering Software*, 95:51–67, May 2016.

[19] S. Mirjalili, S. M. Mirjalili, and A. Lewis. Grey wolf optimizer. *Advances in Engineering Software*, 69:46–61, March 2014.

[20] P. Piprek, P. M. Dias, and D. Schwalb. A generalised multi-objective island model for flight control system clearances. In *33rd Congress of the International Council of the Aeronautical Sciences*, Stockholm, 09 2022.

[21] P. Piprek, P. M. Dias, J. Tonti, and D. Schwalb. Application of clustering techniques in optimisation-based flight control system clearance. In *34th Congress of the International Council of the Aeronautical Sciences*, Florence, 09 2024.

[22] L. Rachmawati and D. Srinivasan. Preference Incorporation in Multi-objective Evolutionary Algorithms: A Survey. In *2006 IEEE International Conference on Evolutionary Computation*, page 962–968. IEEE, 2006.

[23] N. Riquelme, C. Von Lücken, and B. Baran. Performance metrics in multi-objective optimization. In *2015 Latin American Computing Conference (CLEI)*, pages 1–11, 2015.

[24] R. Stich. *Clearance of Flight Control Laws for Carefree Handling of Advanced Fighter Aircraft*, pages 421–442. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[25] United States. Department of Defense. MIL-STD-1797A, Flying Qualities of Piloted Aircraft. Technical Report MIL-STD-1797A, US Air Force Department, 1990.

[26] L. While, P. Hingston, L. Barone, and S. Huband. A faster algorithm for calculating hypervolume. *IEEE Transactions on Evolutionary Computation*, 10(1):29–38, 2006.

[27] A. B. Yoo, M. A. Jette, and M. Grondona. Slurm: Simple linux utility for resource management. In D. Feitelson, L. Rudolph, and U. Schwiegelshohn, editors, *Job Scheduling Strategies for Parallel Processing*, pages 44–60, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg.