New Heuristics for Multi-Objective Worst-Case Optimization in Evidence-Based Robust Design

Ortega C.[1] and Vasile M.[2]

Abstract— **This paper presents a non-nested algorithm for the solution of multi-objective min-max problems (MOMMP) in worst-case optimization. The algorithm has been devised for evidence-based robust optimization, where the lack of a defined probabilistic behaviour of the uncertain parameters makes it impossible to apply sample-based techniques and forces the designer to identify the worst case over the subdomains of the uncertainty space. In evidence theory, the robustness of the solutions is measured in terms of the Belief in the realization of the value of the design budgets, which acts as a lower bound to the unknown cumulative distribution function of the budget. Thus a means of finding robust solutions in preliminary design consists on applying the minimax model, where the worst-case budget over the uncertainty space is optimized over the control space. The paper proposes a novel heuristic to solve MOMMP and demonstrates its capability to approximate the worst-case Pareto front at a very reduced cost with respect to approaches based on nested optimization**

I. INTRODUCTION

Worst-case scenario optimization provides the best possible solution in the worst possible condition. From a game-theoretic point of view it can be seen as the best response of a system to the best move of its opponent, Nature. Worst-case problems are common in several fields: decision making, robust control, risk analysis, resilient design, etc.

In the context of model-based space systems engineering, worst-case scenario optimization can be used to effectively account for both aleatory and epistemic uncertainty in the framework of evidence theory [9], and thus obtain optimal solutions under severe uncertainty. Furthermore, these solutions constitute a first step towards the derivation of optimal design margins in a more general case.

In evidence theory, both input and model uncertainty is defined by means of basic probability assignments (bpa) associated to subsets of the events space Ω. After combination of several possibly conflicting evidence sources [5] [14], a multivalued mapping of probability masses is assigned to all non-zero-probability subsets, or focal elements. Assuming uncorrelated uncertainties, this mapping can be generally represented by considering a family of potentially superposing and/or disjoint hyperrectangular subsets of Ω that we will from now refer to as U, the uncertainty space. The bpa structure of U can be used to infer the Belief and Plausibility curves of a design, which can be viewed as a lower and upper

[1]PhD Candidate, Department of Mechanical & Aerospace Engineering, University of Strathclyde, 75 Montrose Street, Glasgow, UK (carlos.ortega@strath.ac.uk).

[2]Professor, Department of Mechanical & Aerospace Engineering, University of Strathclyde, 75 Montrose Street, Glasgow, UK (massimiliano.vasile@strath.ac.uk).

bound, respectively, to the unknown cumulative probability function of the system budget for that design.

The drawback of this holistic approach for uncertainty quantification is that it leads to exponential complexity with respect to the number of uncertain variables and is therefore rarely affordable for complex engineering systems.

Nonetheless, with clever transformations and adequate optimizers, the min-max approach can tackle the uncertainty space as a whole, detecting rare events and offering a first, conservative solution to the robust optimality problem. This solution provided, strategies for reduced-cost estimation of the Belief curve by means of worst-case optimization under certain assumptions were proposed in [13] and applied to single-objective estimation of robust-optimal spacecraft designs in [1], and will be refined in future works by the authors. Cost-efficient min-max optimization is hence a key building block towards computationally-affordable robust design under inhomogeneous sources of uncertainty.

II. WORST-CASE OPTIMIZATION

Worst-case optimization can be formulated without loss of generality as bi-level min-max optimization over the design space D and the uncertainty space U, i.e.:

$$\{\mathbf{d}^*, \mathbf{u}^*\} = \arg \min_{\mathbf{d} \in D} \max_{\mathbf{u} \in U} (f(\mathbf{d}, \mathbf{u})), \tag{1}$$

where $f(\mathbf{d}, \mathbf{u})$ is the model of the system budget to be minimized, and $\mathbf{d}^*$ represents the worst-case-optimal design, with $\mathbf{u}^*$ its worst-case coordinates in the uncertainty space. When multiple conflicting but statistically uncorrelated criteria $f^1, f^2, \ldots, f^{n_f}$ need to be simultaneously satisfied, one can formulate worst-case scenario optimization as a multi-objective min-max optimization problem with parallel maximization of each objective over the uncertainty space,

$$\{\mathbf{d}^*, \mathbf{u}^{l*}\} = \arg \min_{\mathbf{d} \in D} \max_{\mathbf{u} \in U} [f^l(\mathbf{d}, \mathbf{u}^l)]^T, \quad l \in \{1, \ldots, n_f\}. \tag{2}$$

The maxima over U representing the worst possible case for each of the criteria are simultaneously optimized, in the Pareto sense, over the design space D. Note that this differs from multi-objective bi-level optimization as in [4] in the mode of failure not being assumed equal for all the criteria during the design phase. In this paper MACSminmax, a cost-efficient approach for the solution of such problems building on the work presented in [2], will be proposed an discussed.

III. MACSMINMAX

MACSminmax (Algorithm 2) is a multi-objective min-max optimization meta-algorithm inspired by the relaxation

Algorithm 1 Relaxation procedure (single-objective)

1: Initialise archive $A_u = \{\mathbf{u}_1\}$ and $i = 1$

2: **while** the termination condition is not met **do**

3: $\quad \mathbf{d}_i \leftarrow \arg\min_{\mathbf{d} \in D} \left\{ \max_{\mathbf{u} \in A_u} f(\mathbf{d}, \mathbf{u}) \right\}$

4: $\quad \mathbf{u}_{i+1} \leftarrow \arg\max_{\mathbf{u} \in U} f(\mathbf{d}_i, \mathbf{u})$

5: $\quad A_u \leftarrow A_u \cup \mathbf{u}_{i+1}$

6: $\quad i \leftarrow i + 1$

7: **end while**

8: **return** $\{\mathbf{d}_i, \mathbf{u}_{i+1}, f(\mathbf{d}_i, \mathbf{u}_{i+1})\}$

---

procedure proposed by Shimizu and Aiyoshi in [10]. It includes as well additional heuristics for enhanced performance, namely surrogate-assisted minimization and local-search refinement. Let us break down this definition in the following subsections.

A. The relaxation procedure

The relaxation approach to min-max optimization is detailed in Algorithm 1. The basic idea behind such an approach is to relax the global optimality requirement in the lower-level problem to a discrete search space A_u that grows at each iteration, thus progressively approximating the global optimality condition. Convergence to the exact solution has been proven in the single-objective case [10], provided that the termination condition is adequately selected.

This approach aims at drastic reduction of the computational cost, with respect to nested methods, via sequential minimization and maximization steps in lines 3 and 4 of Algorithm 1. Nevertheless, in the experience of the authors, its convergence rate is not sufficient to achieve so in a reasonable number of function evaluations in problems complex enough, e.g. presenting a multi-modal landscape in U whose peaks change position with $\mathbf{d}$. This was described as a sort of red queen effect in [11], and becomes especially a drawback when the method is extended directly to the multi-objective case, in which the lower-level problem needs to be solved for every point in the Pareto Front obtained in the minimization step. In such a situation, the oscillation between suboptimal and superoptimal candidate solutions can lead to a total required amount of function evaluations equal or higher than one would expect from a nested approach.

MACSminmax can be viewed as an extension of Algorithm 1 to the multi-objective case, endowed with additional heuristics to soften the aforementioned oscillations. Note that this procedure does not implement any kind of single-level optimization, but embeds calls to the solvers of choice, the choice here being MACS (Multi-Agent Collaborative Search, [16]) and MPAIDEA (Multi-Population Adaptive Inflationary Differential Evolution Algorithm, [12]). In order to preserve modularity of the implementation, this property has been maintained through the extension presented hereby. Nevertheless, future versions might dismiss this feature in order to facilitate the exploration of self-adaptive heuristics in min-max optimization.

B. Surrogate-assisted minimization

Unlike in Algorithm 1, an archive A_d is kept throughout the optimization, containing a record of the outputs of the minimizations in lines 11 and 13 of Algorithm 2 as well as a sparse initial sample. The associated archive A_f with the fitness values of their maxima over each A_u^l is updated as the A_u^l grow. Note that a maximization step has been conducted for each individual in A_d, nonetheless the updates are still necessary to mitigate as much as possible the well-known noisy-function effect of bi-level heuristic optimization, especially when revisiting regions containing individuals found early in the process, when the archive A_u was small. This allows a reduction in the number of function evaluations of the maximization steps to have minimal impact on the overall performance.

The minimization step is assisted by optimization of a response surface built on the archives A_d and A_f. This is key for the softening of the oscillations mentioned in Section III-A; the response surface models the structure of worst cases for values of $\mathbf{d}$ not yet visited. This surrogate modelling of the maxima in the archive catalyses the transfer of information between the maximization and minimization steps and thus largely accelerates min-max convergence in problems where the arguments of the maxima in U change chaotically along D.

On the other hand, applying only such a procedure is insufficient to achieve adequate spreading of the solutions in the Pareto Front. This is due to the low resolution of the surrogate in parts of the D space that have not been sufficiently explored, where maxima over U might be overestimated. Therefore in the proposed procedure some evaluations are devoted to the traditional minimization step with maximization over the archive A_u (line 5 of Algorithm 2), which always underestimates the maxima, favouring unexplored regions. A yet unexplored option to compensate this effect would be to use minimization procedures that account for uncertainty in the surrogate predictions such as EGO [6].

C. Local-search refinement

After the min-max optimization loop, a local search is launched from each point in the archives A_u^l using SQP (Sequential Quadratic Programming). This feature is triggered by a flag, since it adds a substantial number of function evaluations to the procedure and it is only necessary if the problem to solve is multimodal and presents narrow basins of attraction in U that change position with $\mathbf{d}$. Nevertheless in such problems it leads to huge gain in robustness of the Pareto front, since it reduces notably the probability of false outliers presenting deficient maximization in the solution, at a cost proportional to the amount of outliers.

IV. NUMERICAL SET-UP

In Section V the performance of MACSminmax will be demonstrated by means of six bi-objective test-cases defined in Table I, where $n = \dim(D) = \dim(U)$, i.e. the total dimension of the problems is $2n$. The expressions for the

Algorithm 2 MACSminmax

1: $n_{feval} = 0$
2: Initialise archive $A_d = \{d_1, d_2, \ldots, d_{n_{d,0}}\}$
3: **for all** $\mathbf{d}_j \in A_d$ **do**
4:     **for all** $l \in \{1, \ldots, n_f\}$ **do**
5:         $\mathbf{u}_{max,j}^l \leftarrow \arg\max_{\mathbf{u} \in U} f^l(\mathbf{d}_j, \mathbf{u})$ **[MPAIDEA]**
6:         $A_u^l \leftarrow A_u^l \cup \{\mathbf{u}_{max,j}^l\}$
7:     **end for**
8: **end for**
9: $A_f \leftarrow \{[\max_{\mathbf{u}^l \in A_u^l} f^l(\mathbf{d}_j, \mathbf{u}^l), l \in \{1, \ldots, n_f\}], \mathbf{d}_j \in A_d\}$
10: **while** $n_{feval} < n_{feval,max}$ **do**
11:     $\mathbf{d}_{min}^A \leftarrow \arg\min_{\mathbf{d} \in D}[\max_{\mathbf{u}^l \in A_u^l} f^l(\mathbf{d}, \mathbf{u}^l), l \in \{1, \ldots, n_f\}]^T$ **[MACS]**
12:     Fit surrogate $\mathbf{S}(\mathbf{d})$ on data points $\{A_d, A_f\}$
13:     $\mathbf{d}_{min}^S \leftarrow \arg\min_{\mathbf{d} \in D} \mathbf{S}(\mathbf{d})$ **[MACS]**
14:     $A_d \leftarrow A_d \cup \mathbf{d}_{min}^A \cup \mathbf{d}_{min}^S$
15:     **for all** $\mathbf{d}_j \in \mathbf{d}_{min}^A \cup \mathbf{d}_{min}^S$ **do**
16:         **for all** $l \in \{1, \ldots, n_f\}$ **do**
17:             $\mathbf{u}_{max,j}^l \leftarrow \arg\max_{\mathbf{u} \in U} f^l(\mathbf{d}_j, \mathbf{u})$ **[MPAIDEA]**
18:             **if** $\mathbf{d}_j \notin \mathbf{d}_{min}^A$ **or** $f^l(\mathbf{d}_j, \mathbf{u}_{max,j}^l) > \max_{\mathbf{u}^l \in A_u^l} f^l(\mathbf{d}_j, \mathbf{u}^l)$ **then**
19:                 $A_u^l \leftarrow A_u^l \cup \{\mathbf{u}_{max,j}^l\}$
20:             **end if**
21:         **end for**
22:     **end for**
23:     $A_f \leftarrow \{[\max_{\mathbf{u}^l \in A_u^l} f^l(\mathbf{d}_j, \mathbf{u}^l), l \in \{1, \ldots, n_f\}], \mathbf{d}_j \in A_d\}$
24: **end while**
25: **if** *local search refinement enabled* **then**
26:     $A_d^{ref.} \leftarrow \{\emptyset\}$, $A_d^{ND} \leftarrow \{$arguments $\mathbf{d}_j \in A_d$ of non-dominated entries $\mathbf{f}_j \in A_f\}$
27:     **while** $A_d^{ND} \not\subset A_d^{ref.}$ **do**
28:         **for all** $\mathbf{d}_j \in A_d^{ND} \setminus A_d^{ref.}$ **do**
29:             **for all** $l \in \{1, \ldots, n_f\}$ **do**
30:                 Run multi-start local search with $\mathbf{X}_0 \subseteq A_u^l$ **[SQP]**
31:                 Use result to refine $\mathbf{u}_{max,j}^l$ associated to $\mathbf{d}_j$ and update A_f
32:             **end for**
33:             $A_d^{ref.} \leftarrow A_d^{ref.} \cup \{\mathbf{d}_j\}$
34:         **end for**
35:         $A_d^{ND} \leftarrow \{$arguments $\mathbf{d}_j \in A_d$ of non-dominated entries $\mathbf{f}_j \in A_f\}$
36:     **end while**
37: **end if**
38: **return** Non-dominated $\mathbf{f}_i^* \in A_f$, their arguments $\mathbf{d}_i^* \in A_d$ and associated $\mathbf{u}_{max,i}^{l,*}, l \in \{1, \ldots, n_f\}$

---

fitness functions of these test cases are detailed in Table II and their domains in U in Table III. Note that the definition of different domains in U for each objective is not coherent with a real-world problem, but has been allowed to maintain control over the properties against which the algorithm is put to test.

In all these cases, the only parameters that have been adjusted are the number of function evaluations in the calls to MACS and IDEA (see Algorithm 2), the fraction of individuals selected from the archive-based or surrogate-assisted minimizations each iteration, keeping the total constant, and the enabling/disabling of local-search refinement via SQP. Exhaustive tuning has not been conducted; the selection has been based on the properties of the problem and on qualitative performance assessment rather than on maximization of the metrics.

The results are presented in Section V and compared to those obtained by:

- A finely-tuned nested approach, using 10 to 20 times the number of function evaluations of MACSminmax

TABLE I

TEST CASES.

Test case	f_1	f_2	D	n
TC1	MV1	MV3	$[1,5]^n$	2
TC2	MV2	MV8	$[0,3]^n$	8
TC3	MV2	EM1	$[1,5]^n$	8
TC4	MV8	MV9	$[1,3]^n$	2
TC5	MV8	EM1	$[1,5]^n$	4
TC6	MV10	MV9	$[-4,2\pi]^n$	1

TABLE II

TEST FUNCTION EXPRESSIONS.

ID	Expression
MV1	$\sum_{i=1}^{n} d_i u_i^2$
MV2	$\sum_{i=1}^{n} (d_i - u_i)^2$
MV3	$\sum_{i=1}^{n} (5 - d_i)(1 + \cos u_i) + (d_i - 1)(1 + \sin u_i)$
MV8	$\sum_{i=1}^{n} (2\pi - u_i)\cos(u_i - d_i)$
MV9	$\sum_{i=1}^{n} (d_i - u_i)\cos(-5u_i + 3d_i)$
MV10	$\sum_{i=1}^{n} (d_i + u_i)\cos(-u_i(5|d_i| + 5) + 3d_i)$
EM1	$\sum_{i=1}^{n} (u_i - 3d_i)\sin u_i + (d_i - 2)^2$

TABLE III

TEST FUNCTION DOMAINS IN U.

ID	U
MV1	$\left[[-5,-4] \cup [-3,0] \cup [-1,3]\right]^n$
MV2	$\left[[-5,-4] \cup [-3,0] \cup [-1,3]\right]^n$
MV3	$\left[[-5,-4] \cup [-3,0] \cup [-1,3]\right]^n$
MV8	$\left[[-\pi/2, -\pi/6] \cup [0,\pi] \cup [3\pi/4, 3\pi/2]\right]^n$
MV9	$\left[[-5,-4] \cup [-3,0] \cup [-1,3]\right]^n$
MV10	$\left[[-5,-4] \cup [-3,0] \cup [-1,3]\right]^n$
EM1	$\left[[-5,-4] \cup [-3,0] \cup [-1,3]\right]^n$

experiment.

- The extension to multi-objective optimization of Algorithm 1 without any additional heuristics, using computational resources similar to the MACSminmax experiment in terms of function evaluations.

The tuning procedure followed for the nested approach is unrealistic: as a precomputation, the design space has been sampled in $100n$ points and, with the reference worst cases of the sample made available, maximization with MPAIDEA has been conducted until convergence to a 0.1% of the reference. With the aim of obtaining reasonably robust solutions during the maximization, the lower-level algorithm of the nested loop, MPAIDEA, has been passed a maximum number of function evaluations corresponding to the quartile 99 of the aforementioned experiment, and the higher-level algorithm, MACS, has been passed a reasonable number of function evaluations so that the total cost of the nested optimization lays between 10 and 20 times the cost of MACSminmax.

The indicators used to assess the quality of the Pareto front obtained by MACSminmax or the other procedures with respect to a reference Pareto front are the convergence and spreading front-averaged metrics M_{conv} and M_{spr} :

$$M_{conv} = \frac{1}{N_p} \sum_{i=1}^{N_p} \min_{j \in M_p} \left\| g_j - f_i \right\|, \tag{3}$$

$$M_{spr} = \frac{1}{M_p} \sum_{i=1}^{M_p} \min_{j \in N_p} \left\| f_i - g_j \right\|, \tag{4}$$

where N_p is the cardinality of the solution Pareto front f, M_p is the cardinality of the reference front g, and each fitness function has been scaled with its span in g. In order to assess the quality of the maximization, i.e. the robustness of the solutions, the fraction of solution points that converged within 0.1% of the span in g is measured for each objective and reported as p_{max}. All these indicators are presented in Section V as mean and standard deviation over a number of runs (50 for MACSminmax, 20 for the other two approaches).

All reference solutions have been obtained by means of computationally-expensive ad hoc approaches that take advantage of the separability of the test functions. The reference worst cases are mapped via multi-start SQP maximization of each component. The reference Pareto fronts are a combination of the best solutions obtained by NSGA-II [3] and MOEA-D [15], 30 exhaustive runs each.

V. RESULTS

Table IV and Figure 1 illustrate the performance of MACSminmax after 50 runs. The results are overall very good both in terms of robustness — indicated by p_{max}^1 and p_{max}^2 —, and optimality — measured by M_{conv} and M_{spr}.

Regarding robustness, the only failure is test case 5 presenting quite a low probability of maximization of f_1 in the upper left part of the front. Analysis shows this is due to the minimization algorithm returning solutions in a pit of deceptively-maximized individuals over robust solutions of very close but lower optimality, especially in its last selection stages. In a pragmatical context this would translate into the possibility of selecting a promising design in that part of the front to later notice upon closest uncertainty analysis that the worst-case budget of f_1 was underestimated. Nevertheless, it is interesting to remark that the front in which these deceptive designs' true worst cases lay is still non-dominated by the truly robust solutions found by MACSminmax in the lower right part of the front, i.e. these individuals might still be considered interesting worst-case Pareto optimums after noticing the underestimation. Neither does their miscomputed front dominate the reference front, which would turn them into very attractive candidates for selection.

As for optimality, MACSminmax achieves remarkably good results in all test cases, with convergence metrics consistently under 5% and generally even lower, and a good fraction of solution points on the reference front. Spreading metrics appear slightly higher, but Figure 1 shows this is due to inhomogeneous density of front coverage rather than to regions not being covered (with the exception of the deceptive region in test case 5 discussed above), and

combined coverage of a few runs is almost total. All test cases present a high relation of robust points that are on or very close to the reference front; the results provide a sound basis for the selection of candidate worst-case designs.

Table V and Figure 2 show the analogous results for 20 runs of the finely-tuned nested approach with a cost of 10 to 20 times that of MACSminmax. Table VI and Figure 3, for the relaxation procedure without additional heuristics and roughly the same cost as MACSminmax. Comparison with these sets of results illustrates the positive effects of the additional heuristics discussed in Section III.

The relaxation approach performs well in test cases 1 and 3. In these problems, the positions of the maxima in U vary very smoothly along D, nevertheless MACSminmax still gives quite better coverage of the front due to the presence of the surrogate-assisted minimization. This is not the case with test case 6, that presents a very rugged landscape in $D \times U$ for f_1, and arguments in U of the maxima of f_2 moving along D, hence its p_{max} indicators are low. But the differences between the modes when one moves along D are small or smooth enough so that a good approximation of the reference front can be detected after several runs, in other words the algorithm is capable of detecting the front but the proportion of false outliers is still large. In this case, the solution of MACSminmax is much neater thanks to the local-search refinement, which filters most outliers. On the other hand, for test case 2 the robustness of the solutions is excellent but the relaxation procedure gets stuck in a deceptive Pareto front optimality-wise. In such a case, running a maximization refinement is worthless. Analysis shows that the algorithm quickly detects that front but remains oscillating because of the multi-modality of f_2 in $D \times U$; the problem is the transfer of information between the minimization and maximization steps (note that the nested approach outperforms the relaxation procedure in overcoming this deceptive front). Neither does MACSminmax reach the slippery reference front, but the surrogate-assisted minimization heuristic allows it to return a much better estimate. The same occurs in the well-approximated region of test case 5. Finally, a combination of the two effects just described allow MACSminmax to find an acceptable solution for test case 4, which presents both minimization convergence issues and outliers in f_2.

As for the nested approach, in spite of the high number of function evaluations devoted to mitigating the noisy-function effect that inaccurate maximizations produce in the multi-objective minimizer, the p_{max} indicators are overall worse than MACSminmax and Figure 2 shows a high number of unfiltered false outliers in test cases 1, 2, 4 and 6. This behaviour is of course not unusual in nested optimization; since false outliers will outperform true min-max individuals in the Pareto-ranking selection steps of the minimizer, it is often necessary to combine several runs and remove outliers by hand based on density of individuals, but note that in test case 4 only a very small fraction of the front is inferable at all even after 20 runs, and in test case 2 the probability of mistaking a false outlier whose true worst-case happens to be in the deceptive front discussed in the former paragraph is

annoyingly high amongst the non-dominated solutions with $f_2 \in [35, 45]$. On the other hand, in test cases 3 and 5 the nested solution achieves good robustness, but gives fronts of much worse optimality (convergence and spreading) than MACSminmax, failing to explore a whole region of the front.

VI. CONCLUSIONS

This paper presents MACSminmax (Algorithm 2), a meta-algorithm for multi-objective worst-case optimization, and demonstrates the gain that the heuristics deployed achieve both with respect to a much more expensive nested approach and with respect to the procedure detailed in Algorithm 1, which MACSminmax takes inspiration from. It has been proven, by means of six min-max optimization toy problems of variate landscape and complexity:

1) that surrogate-assisted minimization catalyses the transfer of information between the minimization and maximization loops of Algorithm 1, and enhances the convergence rate and spreading properties of the Pareto fronts obtained,
2) that archive-based local-search refinement manages to filter a great portion of false outliers in worst-case problems with a complex min-max structure,
3) and that, endowed with such heuristics, MACSminmax can be used to effectively provide a means for selection of worst-case Pareto-optimal solutions in preliminary model-based design that vastly outperforms nested optimization.

Research is in progress to further reduce the cost of multi-objective worst-case optimization by exploring efficient techniques based on Kriging metamodels for single-objective [6], multi-objective [7], and single-objective min-max [8] optimization. From the broader point of view of the evidence-based robust design approach which MACSminmax constitutes a first building block of, future works will put to the test several approaches for the approximate evaluation of the Belief in the realisation of a system budget, and pave the way towards a model-based systems design paradigm where the problem of quantifying the optimal gain-risk curve of a preliminary design under inhomogeneous sources of uncertainty can be effectively solved.

REFERENCES

[1] S. ALICINO AND M. VASILE, *Evidence-based preliminary design of spacecraft.*, in SECESA 2014, Oct. 2014.
[2] S. ALICINO AND M. VASILE, *An evolutionary approach to the solution of multi-objective min-max problems in evidence-based robust optimization*, in 2014 IEEE Congress on Evolutionary Computation (CEC2014), Jul. 2014.
[3] K. DEB, A. PRATAP, S. AGARWAL, AND T. MEYARIVAN, *A fast and elitist multiobjective genetic algorithm: Nsga-ii*, IEEE Transactions on Evolutionary Computation, 6 (2) (2002), pp. 182–197.
[4] K. DEB AND A. SINHA, *An efficient and accurate solution methodology for bilevel multi-objective programming problems using a hybrid evolutionary-local-search algorithm.*, Evolutionary Computation Journal, 18(3) (2010), pp. 403–449.

TABLE IV
MACSMINMAX RESULTS METRICS OVER 50 RUNS.

Test case	N_{feval}		M_{conv}		M_{spr}		p^1_{max}		p^2_{max}	
	Mean	Std. dev.	Mean	Std. dev.	Mean	Std. dev.	Mean	Std. dev.	Mean	Std. dev.
1	$5.670 \cdot 10^4$	$1.808 \cdot 10^3$	0.004	0.001	0.022	0.006	1.000	0.000	0.998	0.008
2	$1.042 \cdot 10^6$	$9.976 \cdot 10^3$	0.037	0.004	0.055	0.011	1.000	0.000	0.972	0.023
3	$3.781 \cdot 10^5$	$2.064 \cdot 10^4$	0.012	0.002	0.025	0.004	1.000	0.000	1.000	0.000
4	$3.980 \cdot 10^5$	$3.858 \cdot 10^4$	0.027	0.007	0.042	0.008	1.000	0.000	0.994	0.020
5	$5.482 \cdot 10^5$	$1.101 \cdot 10^4$	0.015	0.005	0.035	0.005	0.760	0.064	0.954	0.049
6	$4.351 \cdot 10^4$	$3.350 \cdot 10^3$	0.009	0.003	0.042	0.008	0.997	0.014	1.000	0.000

TABLE V
NESTED APPROACH RESULTS METRICS OVER 20 RUNS.

Test case	N_{feval}		M_{conv}		M_{spr}		p^1_{max}		p^2_{max}	
	Mean	Std. dev.	Mean	Std. dev.	Mean	Std. dev.	Mean	Std. dev.	Mean	Std. dev.
1	$6.198 \cdot 10^5$	$1.731 \cdot 10^4$	0.019	0.013	0.050	0.079	0.930	0.040	0.967	0.032
2	$1.074 \cdot 10^7$	$3.296 \cdot 10^5$	0.037	0.006	0.067	0.037	0.888	0.042	0.986	0.024
3	$7.557 \cdot 10^6$	$9.512 \cdot 10^5$	0.098	0.026	0.395	0.060	0.992	0.036	0.990	0.032
4	$4.115 \cdot 10^6$	$8.739 \cdot 10^4$	0.046	0.076	0.120	0.172	0.973	0.056	0.827	0.130
5	$7.146 \cdot 10^6$	$1.109 \cdot 10^6$	0.062	0.039	0.127	0.043	0.986	0.035	0.853	0.111
6	$5.386 \cdot 10^5$	$3.435 \cdot 10^4$	0.010	0.003	0.031	0.015	0.993	0.015	0.961	0.028

TABLE VI
RELAXATION PROCEDURE RESULTS METRICS OVER 20 RUNS.

Test case	N_{feval}		M_{conv}		M_{spr}		p^1_{max}		p^2_{max}	
	Mean	Std. dev.	Mean	Std. dev.	Mean	Std. dev.	Mean	Std. dev.	Mean	Std. dev.
1	$5.508 \cdot 10^4$	$9.296 \cdot 10^2$	0.006	0.002	0.034	0.002	1.000	0.000	1.000	0.000
2	$1.042 \cdot 10^6$	$2.167 \cdot 10^2$	0.055	0.006	0.077	0.007	1.000	0.000	0.975	0.043
3	$3.646 \cdot 10^5$	$1.645 \cdot 10^2$	0.019	0.006	0.057	0.011	1.000	0.000	1.000	0.000
4	$4.064 \cdot 10^5$	$3.057 \cdot 10^2$	0.037	0.013	0.061	0.010	1.000	0.000	0.480	0.149
5	$5.228 \cdot 10^5$	$2.047 \cdot 10^2$	0.041	0.017	0.157	0.070	0.866	0.155	0.791	0.196
6	$4.089 \cdot 10^4$	$2.041 \cdot 10^3$	0.014	0.010	0.066	0.017	0.787	0.131	0.870	0.117

[5] A. P. DEMPSTER, *Upper and lower probabilities induced by a multi-valued mapping*, The Annals of Statistics, 28 (1967), pp. 325–339.

[6] D. JONES, M. SCHONLAU, AND W. WELCH, *Efficient global optimization of expensive black-box functions*, Journal of Global Optimization, 13 (1998), pp. 455–492.

[7] J. KNOWLES, *Parego: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems*, IEEE Transactions on Evolutionary Computation, 10 (1) (2005), pp. 50–66.

[8] J. MARZAT, E. WALKER, AND H. PIET-LAHANIER, *Worst-case global optimization of black-box functions through kriging and relaxation*, Journal of Global Optimization, 55 (2013), pp. 707–727.

[9] G. SHAFER, *A Mathematical Theory of Evidence*, Princeton University Press, 1976.

[10] K. SHIMIZU AND E. AIYOSHI, *Necessary conditions for min-max problems and algorithms by a relaxation procedure*, IEEE Transactions on Automatic Control, AC-25 (1980), pp. 62–66.

[11] M. VASILE, *On the solution of min-max problems in robust optimization*, in The EVOLVE 2014 International Conference, Jul. 2014.

[12] M. VASILE, E. MINISCI, AND M. LOCATELLI, *An inflationary differential evolution algorithm for space trajectory optimization*, IEEE Transactions on Evolutionary Computation, 15 (2011), pp. 267–281.

[13] M. VASILE, E. MINISCI, AND Q. WIJNANDS, *Approximated computation of belief functions for robust design optimization*, in 53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, Apr. 2012.

[14] L. ZHANG, *Representation, independence, and combination of evidence in the dempster-shafer theory*, (1994).

[15] Q. ZHANG AND H. LI, *Moea/d: A multiobjective evolutionary algorithm based on decomposition*, IEEE Transactions on Evolutionary Computation, 11 (6) (2007), pp. 712–731.

[16] F. ZUIANI AND M. VASILE, *Multi-agent collaborative search based on tchebycheff decomposition*, Computational Optimization and Applications, (2013).
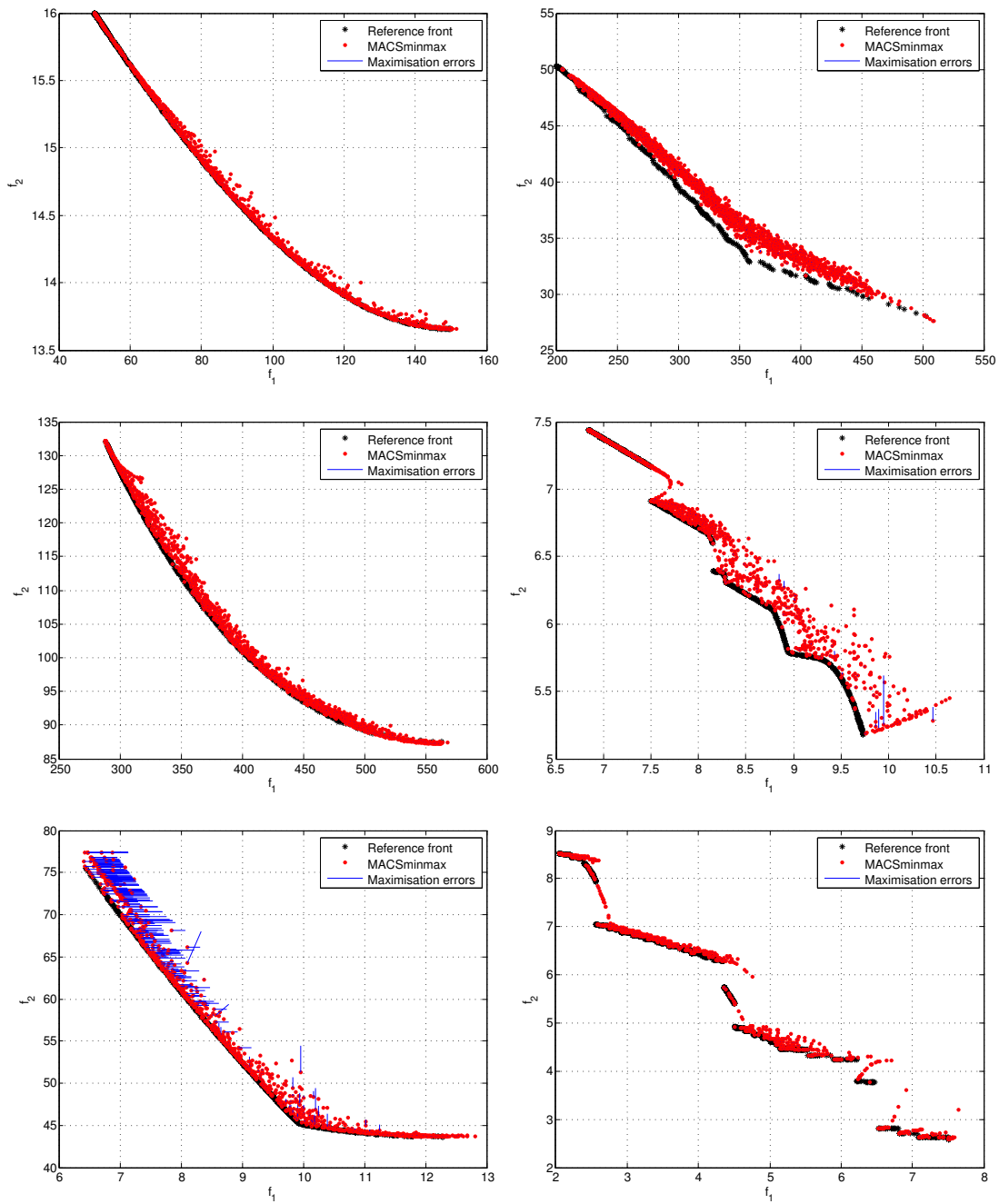
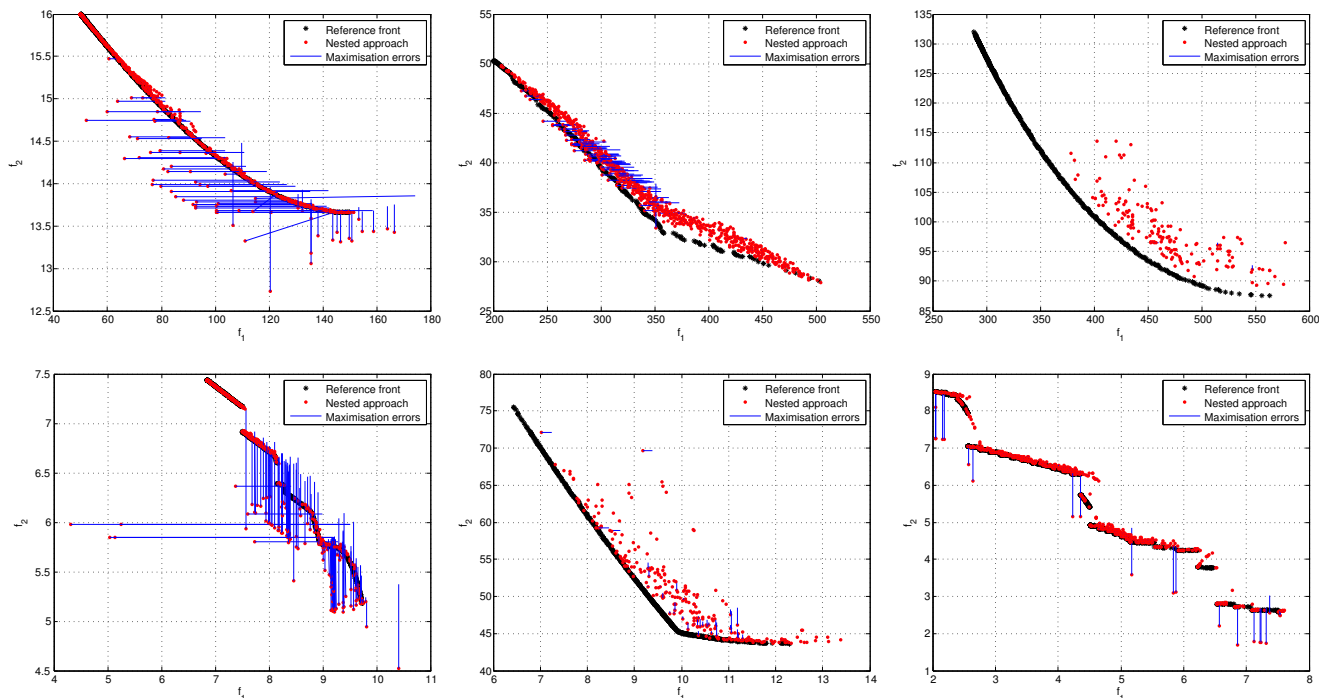Fig. 1. Pareto fronts obtained with 50 runs of MACSminmax.

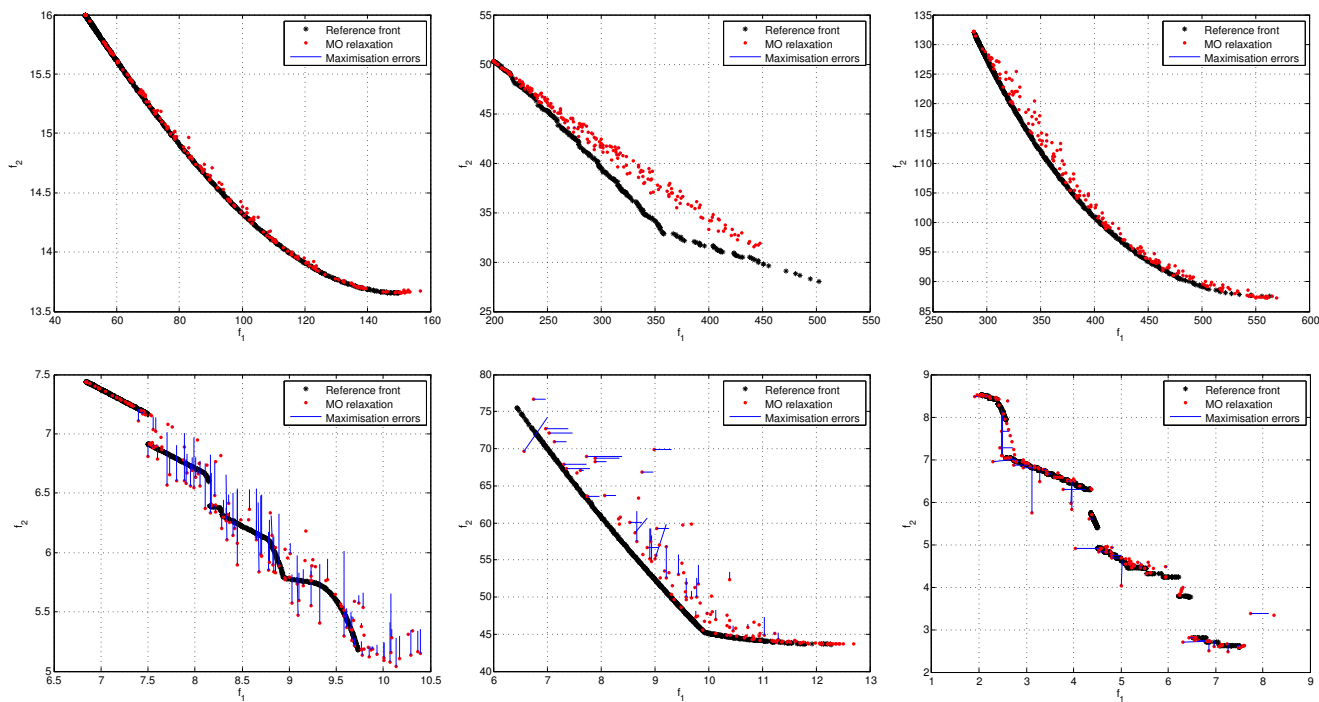Fig. 2. Pareto fronts obtained with 20 runs of the nested approach.

Fig. 3. Pareto fronts obtained with 20 runs of the relaxation procedure.