

# Diagnostic Assessment of the Borg MOEA for Many-Objective Product Family Design Problems

David Hadka  
Department of Computer  
Science and Engineering  
The Pennsylvania State University  
University Park, PA, USA  
Email: dmh309@psu.edu

Patrick M. Reed  
Department of Civil Engineering  
The Pennsylvania State University  
University Park, PA, USA  
Email: pmr11@psu.edu

Timothy W. Simpson  
Department of Industrial  
and Manufacturing Engineering  
The Pennsylvania State University  
University Park, PA, USA  
Email: tws8@psu.edu

**Abstract**—The recently introduced Borg multiobjective evolutionary algorithm (MOEA) framework features auto-adaptive search that tailors itself to effectively explore different problem spaces. A key auto-adaptive feature of the Borg MOEA is the dynamic allocation of search across a suite of recombination and mutation operators. This study explores the application of the Borg MOEA on a real-world product family design problem: the severely constrained, ten objective General Aviation Aircraft (GAA) problem. The GAA problem represents a promising benchmark problem that strongly highlights the importance of using auto-adaptive search to discover how to exploit multiple recombination strategies cooperatively. The auto-adaptive behavior of the Borg MOEA is rigorously compared against its ancestor algorithm, the  $\epsilon$ -MOEA, by employing global sensitivity analysis across each algorithm's feasible parameter ranges. This study provides the first Sobol' sensitivity analysis to determine the individual and interactive parameter sensitivities of MOEAs on a real-world many-objective problem.

## I. INTRODUCTION

As computers become faster, cheaper and more ubiquitous, engineers continue to increase the fidelity of their models to capture more complex phenomena and integrate analyses from multiple disciplines when examining system-wide tradeoffs [1]. Regardless of the nature of these systems, decision makers no longer want a single, numerical solution to a design problem — they want the ability to explore and visualize tradeoffs between multiple conflicting objectives to aid them in understanding the range of potential solutions that are available to them [2], [3], [4], [5].

Consequently, multiobjective evolutionary algorithms (MOEAs) are continuing to grow in popularity in many engineering fields due to their ability to handle problems with different types of decision variables, cope with non-linear, multi-modal and discontinuous search landscapes, and approximate the set of Pareto optimal (Pareto efficient) solutions in a single run (for more details, see [6]). While MOEAs have proven reliable on multiobjective problems with two or three objectives, many commonly-used MOEAs deteriorate on problems with four or more objectives [7]. At higher dimensions, many MOEAs become algorithmically inefficient, are incapable of producing a diverse set of solutions and/or fail to converge with a high probability [8]. For these reasons, the application of MOEAs to

many-objective engineering problems has the potential for significant search failures [6], [9].

The recently introduced Borg MOEA [10] has been designed with the goal of helping to overcome these problems. The Borg MOEA consists of three key components: 1) an  $\epsilon$ -dominance archive to maintain a diverse set of Pareto optimal solutions; 2) a restart mechanism triggered by search stagnation to avoid pre-convergence to local optima; and 3) the use of multiple search operators that adapt to a given problem's landscape. Performing one of the largest comparative studies of state-of-the-art MOEAs, Hadka and Reed [8] compared the effectiveness, reliability, efficiency and controllability of the Borg MOEA relative to eight state-of-the-art algorithms on 33 test problem instances from the DTLZ and CEC test problem suites. They found the Borg MOEA matched or exceeded the performance of the other MOEAs on the majority of test problem instances, and was particularly effective on the many-objective problems. Reed et al. [9] extended this study by applying the same nine algorithms to three real-world, water resource engineering applications. Across the three water resource engineering applications, the Borg MOEA again proved to be efficient, effective, reliable and easy-to-use (i.e., large parameter sweet spots [11]). This suggests the Borg MOEA is a strong candidate MOEA for application to many-objective engineering design problems.

To further explore the characteristics of the Borg MOEA, this study provides a detailed statistical analysis of the Borg MOEA's search controls relative to its non-adaptive ancestor, the  $\epsilon$ -MOEA, on a severely challenging real-world engineering design problem. It should be noted that our prior comprehensive assessment of MOEAs [8] showed that the Borg MOEA was best overall when compared against eight state-of-the-art MOEAs, and the  $\epsilon$ -MOEA was a top performer among the non-adaptive traditional MOEAs. The algorithms' search controls are rigorously assessed using Sobol' variance-based global sensitivity analysis [12], [13], [14].

The product family design problem tested in this study is the General Aviation Aircraft (GAA) design problem introduced by Simpson et al. [15]. Compared to existing product family design problems [16], it is a relatively small problem that involves the design of three general aviation aircraft (a product

TABLE I  
DESIGN PARAMETERS AND THEIR RESPECTIVE RANGES.

Design Variable	Units	Min	Max
Cruise Speed	Mach	0.24	0.48
Aspect Ratio	-	7	11
Sweep Angle	-	0	6
Propeller Diameter	ft	5.5	5.968
Wing Loading	lb/ft <sup>2</sup>	19	25
Engine Activity Factor	-	85	110
Seat Width	inch	14	20
Tail Length/Diameter Ratio	-	3	3.75
Taper Ratio	-	0.46	1

family) that share common subsystems but must satisfy the needs of various general aviation clients. “General aviation” refers to all flights excluding military and commercial operations, and thus caters to a diverse set of potential clientele, from recreational pilots to traveling business executives. As a single aircraft cannot meet all individual needs, three aircrafts are designed to accommodate 2, 4 and 6 passengers while satisfying a wide variety of performance and economic constraints. The over-arching goal in the GAA product family design problem is the design of the three aircraft to maximize the commonality of subsystems on all three aircraft to reduce costs while simultaneously addressing the conflicting goal of maximizing the tailored performance characteristics of the individual aircraft.

Simpson et al. [15] introduced the GAA problem and solved it using a two-objective formulation, but they found that they could not generate feasible points and had to allow 3% constraint violations to attain design alternatives. Subsequent to its introduction, further research into alternate formulations and solution strategies have also struggled when solving the GAA problem [17], [18]. Shah et al. [19] was the first successful application of a MOEA, the  $\epsilon$ -NSGA-II [20], to the GAA problem, successfully generating a large number of potential constraint-satisfying designs.

To characterize the difficulty of the GAA problem, Shah et al. [19] performed an experiment where they used Monte Carlo sampling to generate 50 million designs and obtained only four constraint-satisfying designs. Furthermore, these four designs were all dominated by designs produced by  $\epsilon$ -NSGA-II. This highlights that unguided sampling explorations of the problem hold little to no value for informing the decision makers. For these reasons, the GAA problem provides a compelling baseline for testing MOEAs on severely constrained problems.

In this study, we follow the 27 decision variable, 10 objective and 1 aggregate constraint violation formulation of the GAA problem used by Shah et al. [19]. Table I lists the decision variables and their allowable range for each aircraft in the family. Table II lists the ten objectives being optimized for each individual aircraft. Readers are referred to [15] and [19] for full details on the GAA problem. The  $\epsilon$ -NSGA-II was not included in this study because our primary focus is demonstrating how the auto-adaptive search operators of the

TABLE II  
OBJECTIVES AND  $\epsilon$  VALUES.

Objective	Units	Min/Max	$\epsilon$
Takeoff Noise	dB	min	0.15
Empty Weight	lb	min	30
Direct Operating Cost	\$/hour	min	6
Ride Roughness	-	min	0.03
Fuel Weight	lb	min	30
Purchase Price	1970 \$	min	3000
Flight Range	nm	max	150
Max Lift/Drag Ratio	-	max	0.3
Max Cruise Speed	kts	max	3
Product Family Penalty Function	-	min	0.3

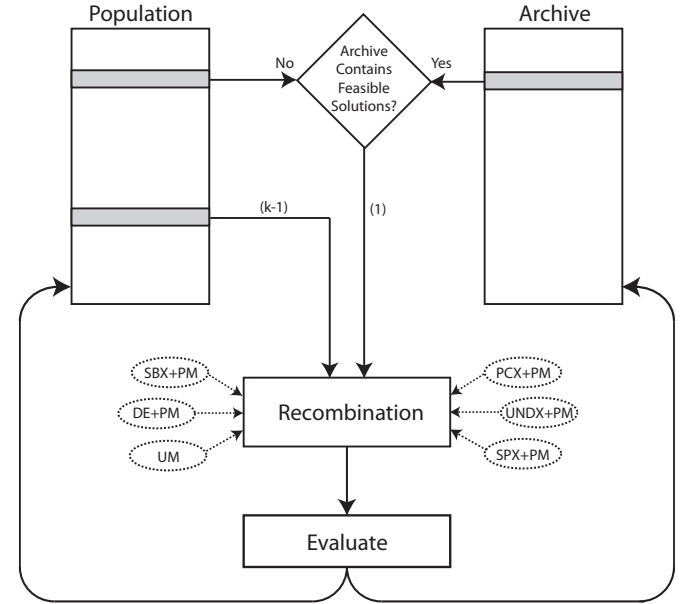


Fig. 1. Flowchart of the Borg MOEA main loop. First, one of the recombination operators is selected using the adaptive multi-operator procedure described in Section II-B. For a recombination operator requiring  $k$  parents,  $k-1$  parents are selected from the population using tournament selection. The remaining parent is selected randomly from the archive if the archive contains feasible solutions; otherwise it is selected randomly from the population. The offspring resulting from this operator are evaluated and then considered for inclusion in the population and archive.

Borg MOEA distinguish its performance from  $\epsilon$ -MOEA. We have verified that the Borg MOEA is fully superior to the  $\epsilon$ -NSGA-II on the GAA problem in a separate effort.

## II. METHODOLOGY

This study provides a rigorous statistical comparison of  $\epsilon$ -MOEA and the Borg MOEA.  $\epsilon$ -MOEA [21] is notable for being a steady-state algorithm, meaning only one individual is modified per iteration. Additionally,  $\epsilon$ -MOEA is the first MOEA to use  $\epsilon$ -dominance archives to maintain a diverse set of Pareto optimal solutions. Theoretically, an  $\epsilon$ -dominance archive combined with an algorithm capable of producing all feasible designs with non-zero probability will converge to the true Pareto optimal set given sufficient time [22].

The Borg MOEA, as illustrated in Figure 1, is an extension of  $\epsilon$ -MOEA and differs in several key areas. The following sections introduce each of the extensions included in the Borg MOEA. Full details of the Borg MOEA are available in [10].

#### A. $\epsilon$ -Progress Triggered Restarts

Since the  $\epsilon$ -dominance archive is the set of all non-dominated solutions produced by the MOEA, Hadka and Reed [10] propose monitoring the  $\epsilon$ -dominance archive to detect search stagnation. If no new non-dominated solutions are accepted into the  $\epsilon$ -dominance archive over a period of time, the MOEA has stagnated. For instance, the MOEA may be stuck at a local optima. This mechanism of monitoring the  $\epsilon$ -dominance archive for search stagnation is called  $\epsilon$ -progress. In the Borg MOEA, if the entire population is evolved and the  $\epsilon$ -dominance archive remains unchanged (no  $\epsilon$ -progress), then a restart is triggered.

A restart involves several steps designed to help the algorithm escape local optima and introduce additional diversity into the search population. First, the population is emptied. Second, the population is resized relative to the  $\epsilon$ -dominance archive. Several studies theoretically and experimentally demonstrate that maintaining a population size relative to the Pareto approximate set, as inferred by the  $\epsilon$ -dominance archive size, helps avoid preconvergence [23], [24], [25]. Finally, the population is filled with all solutions in the  $\epsilon$ -dominance archive. Any remaining slots in the population are filled with randomly-selected  $\epsilon$ -dominance archive members that undergo uniform mutation applied with probability  $1/L$ . This seeding reintroduces previously-discovered non-dominated solutions into the search population but also introduces additional diversity through the mutation operator.

Resizing is controlled by the injection rate parameter. For instance, an injection rate of 0.25 states that 25% of the new population will consist of unaltered individuals copied from the  $\epsilon$ -dominance archive. The remaining 75% undergo the uniform mutation perturbation. A restart may also occur if the ratio of the population size to the  $\epsilon$ -dominance archive size exceeds the injection rate by more than 25%. This ensures the population size remains proportional to the  $\epsilon$ -dominance archive size throughout the run.

#### B. Auto-Adaptive Multi-Operator Search

One of the problems encountered when using MOEAs is the inability to know *a priori* which recombination operator performs best on a given problem. Moral et al. [26] first proposed the use of multiple search algorithms in their switching algorithm. This switching approach involves switching randomly to a different search algorithm if certain criteria for progress are not met. Vrugt et al. [27], [28] proposed a more adaptive approach, AMALGAM, whereby the application of each search algorithm is controlled probabilistically based on the performance attained by each algorithm. The key limitations of both approaches is their use of algorithms which scale poorly on many-objective problems. The Borg MOEA improves upon these designs by extending  $\epsilon$ -MOEA,

which scales well on many-objective problems, and integrating multiple search operators in a highly-adaptive framework.

Furthermore, the full suite of variational operators utilized in the Borg MOEA were not considered in prior multi-method approaches. This is important as the Borg MOEA is better described as a MOEA framework that instantiates itself algorithmically based on which mating and mutation operators or operator combinations are most effective for a given problem. The issue of adaptive operator choice is both important and a core contribution demonstrated in this study for severely-constrained many-objective real-world problems.

Since the GAA problem uses real-valued encoding, six real-valued operators were selected for this study.

- 1) Simulated Binary Crossover (SBX) [29]
- 2) Differential Evolution (DE) [30]
- 3) Parent-Centric Crossover (PCX) [31]
- 4) Simplex Crossover (SPX) [32]
- 5) Unimodal Normal Distribution Crossover (UNDX) [33]
- 6) Uniform Mutation (UM) applied with probability  $1/L$

In addition, offspring produced by SBX, DE, PCX, SPX, and UNDX are mutated using polynomial mutation (PM) [29]. It should be noted that these operators provide a variety of offspring distributions. For instance, SBX, PCX, and PM produce offspring near one of the parents. Such small perturbations helps fine-tune existing designs. SPX and DE result in larger perturbations, allowing the MOEA to move across and search large landscapes efficiently. UNDX produces offspring about the centroid of the parents, quickly converging to valleys in the landscape. UM is the most disruptive of the operators, which aids in adding diversity to the population to prevent preconvergence.

Another key difference between these operators is rotational-invariance. In the ideal case, all decision variables are independent and can thus be optimized independently. However, it is common in real-world engineering problems to encounter large amounts of interaction (epistasis) between decision variables. SBX and PM are tailored for problems with independent decision variables. PCX, SPX, UNDX, and DE are rotationally-invariant, and will often perform better on epistatic problems.

The Borg MOEA uses all six operators, but adapts the probability that each operator is applied based on the success of each operator from prior iterations. Since the  $\epsilon$ -dominance archive maintains the best solutions in terms of diversity and convergence produced by the algorithm, the Borg MOEA favors operators which contribute solutions to the  $\epsilon$ -dominance archive. Thus, operators that consistently contribute high-quality and diverse solutions are given a higher probability of use in future iterations.

More concretely, given  $K > 1$  operators, the Borg MOEA maintains the probabilities  $\{Q_1, Q_2, \dots, Q_K\}$ ,  $Q_i \in [0, 1]$ , of applying each operator to produce the next offspring. These probabilities are initially set to  $Q_i = 1/K$  and are updated periodically by counting the number of solutions in the  $\epsilon$  dominance archive that were produced by each operator,

$\{C_1, C_2, \dots, C_K\}$ . Afterwards, each  $Q_i$  is updated by

$$Q_i = \frac{C_i + \varsigma}{\sum_{j=1}^K (C_j + \varsigma)}. \quad (1)$$

The constant  $\varsigma > 0$  prevents the operator probabilities from reaching 0, thus ensuring no operators are “lost” during the execution of the algorithm. In this study, we use  $\varsigma = 1$ .

### C. Constraint Handling

In this study, the  $\epsilon$ -MOEA and the Borg MOEA both employ the constraint handling technique proposed by Srinivas and Deb [34]. Their approach extends the binary tournament as follows:

- 1) If both solutions violate constraints, then the one with a lower aggregate constraint violation is selected.
- 2) If one solution is feasible and the other solution violates constraints, then the feasible solution is selected.
- 3) If both solutions are feasible, then Pareto dominance is used to select the solution.

Recall that  $\epsilon$ -MOEA selects one parent from the population and the other from the  $\epsilon$ -dominance archive. On constrained problems, if no feasible solutions have been found yet, then the  $\epsilon$ -dominance archive may only contain one solution — the solution that least violates the constraints. This is problematic because the lone solution in the  $\epsilon$ -dominance archive will always be selected as one of the parents. To avoid this issue, the parent selection mechanism in  $\epsilon$ -MOEA and the Borg MOEA were modified as follows:

- 1) If no feasible solutions have been found (i.e., the  $\epsilon$ -dominance archive contains a single solution), then both parents are selected from the population.
- 2) Otherwise, if feasible solutions have been found, then select one parent from the population and the other from the archive.

Figure 1 shows how constraint handling operates within the multioperator procedure. First, one of the six operators is selected using the operator probability distribution. Second, for an operator requiring  $k$  parents,  $k - 1$  are selected from the population using tournament selection. If the archive contains feasible solutions, then the remaining parent is selected randomly from the archive; otherwise, the remaining parent is selected randomly from the population. Lastly, the resulting offspring are inserted back into the population and archive following the same logic as  $\epsilon$ -MOEA.

### D. Sobol’ Sensitivity Analysis

Sobol’ sensitivity analysis is a form of variance decomposition that attributes the variation observed in a model’s output to perturbations of the model’s input [35]. Hadka and Reed [8] explored the application of Sobol’ sensitivity analysis to understanding the effects of an MOEA’s parameters (e.g., population size, number of function evaluations, mutation and crossover rates) on the end-of-run performance of the algorithm. In this study, we continue this work by applying Sobol’ sensitivity analysis to  $\epsilon$ -MOEA and the Borg MOEA for the GAA problem.

By using a special pseudo-random sampling technique proposed by Saltelli et al. [35], one can compute the first-, second- and total-order sensitivity indices using Sobol’ sensitivity analysis. For this application, first-order indices reflect the impact of a single input parameter on end-of-run performance, independent of all other parameters. Second-order effects capture the pairwise interactions between parameters, identifying parameter combinations which influence the behavior of MOEAs. Total-order effects sum the first-order effects with all interactive effects (second-order, third-order, and so on) for a given parameter. By capturing these interactions, researchers can identify the parameter combinations that are important to each MOEA.

Prior work [8] has shown that parameter interactions vary across problems and even vary across the same problem class for different numbers of objectives. When parameter interactions change dramatically across problems, the parameters of an MOEA need to be tuned for each application. It is hypothesized that the auto-adaptive search in the Borg MOEA overcomes these limitations to yield robust search regardless of the parameterization choices (i.e., it has been shown to be highly controllable). The sensitivity analysis in this study attempts to confirm this hypothesis on a real-world problem. Moreover, many existing MOEAs are strongly biased by only considering the directional search provided by the SBX and PM operators. This study examines the Borg MOEA’s multi-operator dynamics for the GAA.

### E. Experimental Setup

To perform Sobol’ sensitivity analysis and present robust statistical results in the form of attainment probabilities, each algorithm was run on the GAA problem using parameters sampled across the algorithm’s full parameter space (see Table III). The parameter samples are produced using the Sobol’ sequence generator, which ensures that the parameters are sampled uniformly from the parameter hyperboxes. For an MOEA with  $P$  parameters, the Sobol’ sequence generator produces  $(2P + 2) * N$  parameterizations. Furthermore, each parameterization is run by the MOEA using 50 random seed replications to fully characterize performance.

This sampling strategy represents a Monte Carlo approximation of each MOEA’s full joint probability distribution function (PDF) of performance from which we can rigorously assess the best achieved value and probability of attainment measures. In total, this study accumulates the results of 2,000,000 sets of MOEA results on the GAA problem.

Given the computational demands of this study, the codes for  $\epsilon$ -MOEA and the Borg MOEA were implemented using the MOEA Framework Java library<sup>1</sup> and executed on the CyberSTAR compute cluster at the Pennsylvania State University. CyberSTAR consists of 384 2.66 GHz Intel Nehalem processors and 128 2.7 GHz AMD Shanghai processors.

<sup>1</sup><http://www.moeaframework.org/>

TABLE III  
SAMPLED PARAMETER RANGES AND DEFAULT SETTINGS

Parameter	Min	Max	Default
(Initial) Population Size	10	1000	100
Max Evaluations	10000	1000000	50000
Injection Rate	0.1	1.0	0.25
SBX Rate	0.0	1.0	1.0
SBX Distribution Index	0.0	500.0	15.0
PM Rate	0.0	1.0	1/L
PM Distribution Index	0.0	500.0	20.0
DE Crossover Rate	0.0	1.0	0.1
DE Step Size	0.0	1.0	0.5
UM Rate	0.0	1.0	1/L
PCX # of Parents	2.0	10.0	3
PCX # of Offspring	1.0	10.0	2
PCX Eta	0.0	1.0	0.1
PCX Zeta	0.0	1.0	0.1
UNDX # of Parents	2.0	10.0	3
UNDX # of Offspring	1.0	10.0	2
UNDX Eta	0.0	1.0	0.5
UNDX Zeta	0.0	1.0	0.35
SPX # of Parents	2.0	10.0	3
SPX # of Offspring	1.0	10.0	2
SPX Epsilon	0.0	1.0	0.5

#### F. Performance Metrics

Performance metrics are used to evaluate the approximation sets produced by running an MOEA, allowing the comparison of approximation sets using numeric values. While hypervolume is a preferred performance metric [36], its use in this study is computationally infeasible due to the GAA problem having 10 objectives. Instead, the following three performance measures are employed by this study, which are detailed in the reference text by Coello Coello [6].

Generational distance (GD) is used as a measure of proximity to the reference set. GD is the average distance of approximation set solutions to the nearest reference set solution. Thus, approximation sets nearer to the reference set result in lower GD values.

Inverted generational distance (IGD) measures the diversity of the approximation set by averaging the distance of reference set solutions to the nearest approximation set solution. Approximation sets with solutions near each reference set solution yield lower IGD values.

Additive  $\epsilon$ -Indicator (AEI) measures the consistency of the approximation set. Since AEI measures the largest distance  $\epsilon$  that the approximation set must be translated to dominate the entire reference set, any region of the approximation set that poorly approximates the reference set will result in larger AEI values. An approximation set that consistently approximates the entire reference set will result in lower AEI values.

All three metrics are normalized by the bounds of the reference set. The ideal value of each is 0.

#### G. Best, Probability of Attainment and Efficiency

As discussed previously, Sobol' sensitivity analysis requires that we globally sample the full parameterization space of each MOEA to approximate the joint PDF for their performance. Consequently, we have defined rigorous measures of their performance to capture (1) the best overall result, (2) the probability of attaining high quality results and (3) the efficiency in attaining high quality results.

First, the best achieved value records the best metric value achieved across all runs of an MOEA, reflecting the absolute best performance observed using that algorithm.

Second, the probability of attainment records the probability that a MOEA surpasses a threshold of performance. For example, if the threshold is set to 0.1, then the probability of attainment records the number of approximation sets measuring a metric value of  $\leq 0.1$ . In this experiment, we vary this threshold from 0 to 1 in increments of 0.01, which allows us to show the change in attainment probabilities across a range of performance thresholds.

Finally, efficiency measures the minimum number of objective function evaluations (NFE) required by the MOEA to produce results exceeding a threshold of performance with high probability. This probability is computed by dividing the parameter hyperbox into bands of 10000 NFE each and determining the fraction of parameters in each band that produce results exceeding the threshold. The band with the minimum NFE that attains the threshold with a probability  $\geq 90\%$  defines our measure of efficiency. For example, if the threshold is set to 0.1 and an MOEA's efficiency is the band 70000 – 80000, then running the MOEA on the problem for 80000 NFE has a high likelihood of producing approximation sets measuring a metric value  $\leq 0.1$  across all of its sampled parameterizations. It is important to generalize performance to this probabilistic context in order to capture efficiency that is robust regardless of how an MOEA is parameterized (i.e., it is efficient and easy-to-use).

### III. RESULTS

$\epsilon$ -MOEA and the Borg MOEA were run using 50 random seed replicates for all of the statistically sampled parameter inputs prescribed in the experimental setup section. Each run produces an approximation set, all of which are combined to form the reference set. This reference set is subsequently used to compute the GD, IGD and AEI metrics. The reference set consists of 630 solutions total, with 112 produced by  $\epsilon$ -MOEA and 518 solutions produced by the Borg MOEA. Figure 2 shows the parallel coordinates plot of the reference set, with solutions produced by  $\epsilon$ -MOEA and the Borg MOEA colored red and blue, respectively. The figure is drawn such that the preferred direction for each objective is toward the bottom of each of the vertical lines for each objective. Figure 2 shows that the Borg MOEA finds a far more diverse set of solutions and that its solutions are more effectively discovering the extremes.

Note that we say an algorithm produced the reference set solution if it was not dominated by any other solutions produced



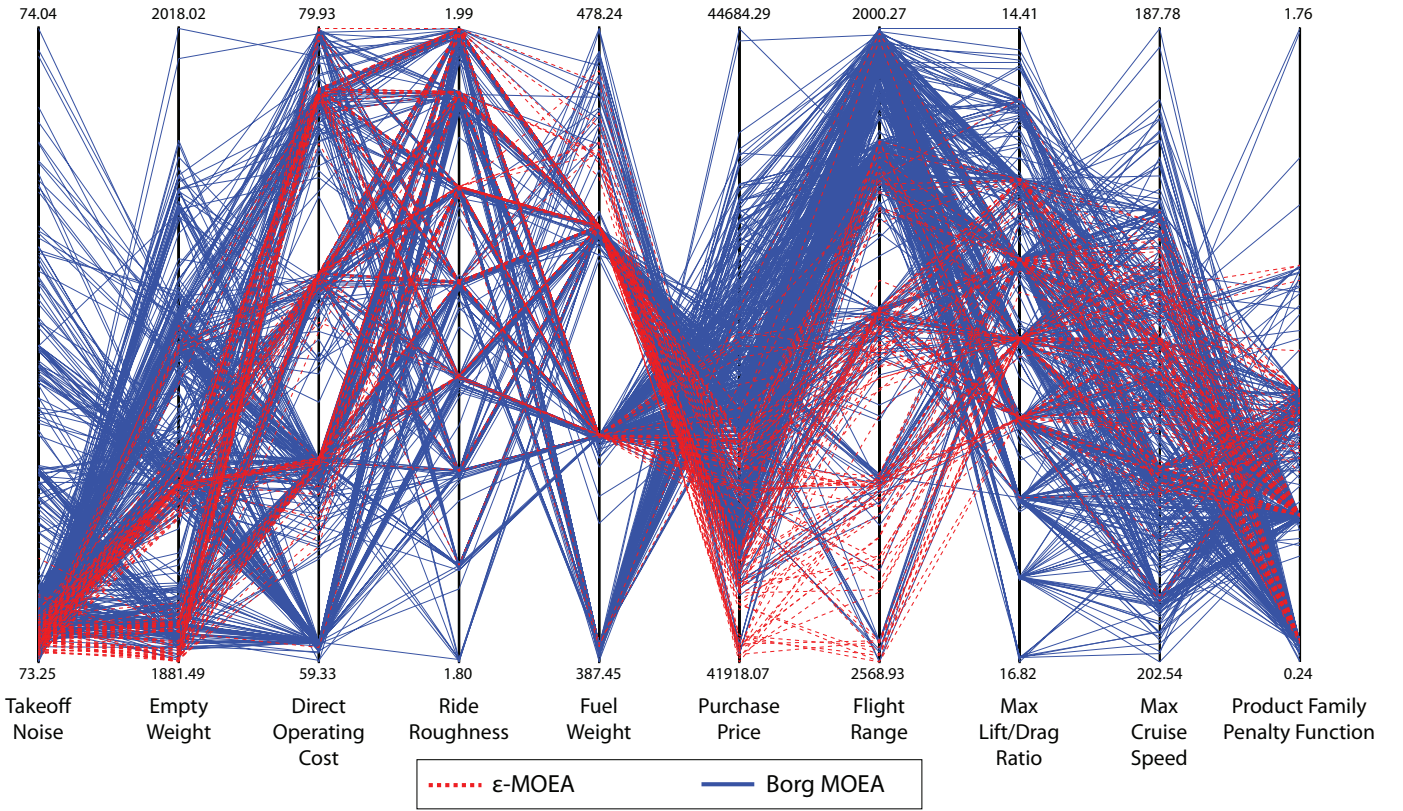


Fig. 2. Parallel coordinates plot of the reference set generated by  $\epsilon$ -MOEA and the Borg MOEA. The traces in the plot are colored by the algorithm which produced the solution. The ideal direction for each objective is downwards.

by the other algorithm. However, given that both  $\epsilon$ -MOEA and the Borg MOEA utilize  $\epsilon$  as a problem-specific resolution for determining significant differences between solutions, we can also ask how many reference set solutions are  $\epsilon$ -covered by each algorithm. A reference set solution is  $\epsilon$ -covered by an algorithm if there exists a solution in the approximation set whose distance from the reference set is smaller than  $\epsilon$ . Percentage-wise,  $\epsilon$ -MOEA  $\epsilon$ -covered 38.7% of the reference set when accumulating results across all its runs. The Borg MOEA  $\epsilon$ -covered 97.3% of the reference set across its runs. This implies the Borg MOEA nearly perfectly specified the entire reference set for this problem. It is also worth noting that the  $\epsilon$ -values when using  $\epsilon$ -dominance archiving are not algorithm parameters. They represent the “significant precision” for each objective for an engineering or real world calculation (i.e., in real world applications, overly precise non-domination is meaningless and can even be harmful). This fact is often lost when studies focus solely on test functions that have no physical meaning.

#### A. Best, Probability of Attainment and Efficiency

Next, we examine the overall best achieved value and attainment probabilities in Figure 3. Each subplot in Figure 3 shows the results for the GD, IGD, and AEI metrics for both algorithms. The y-axis provides the distance of the approximation set from the reference set. Ideal performance would have all runs measuring a distance of 0 from the reference

set. The solid dots indicate the best achieved metric value for each algorithm (smallest distance) across all of its runs. Thus, a solid dot nearer to the top of each subplot indicates at least one parameterization of the algorithm performed ideally for a given metric. For all three metrics, the Borg MOEA slightly outperforms  $\epsilon$ -MOEA with regards to the overall best achieved metric value. It should be noted that this is not a very strong measure of performance. Users would be interested being able to attain this level of performance regardless of their parameterization choices (i.e., a high probability of attainment across the sampled parameterizations for each algorithm).

In Figure 3, the probability of attainment is shown by the shaded bars. Recall that the y-axis shows the threshold, varying in distance from the reference set. The color represents the probability of the parameterizations exceeding the threshold value, where black indicates 100% attainment probability and white indicates 0% attainment probability. Intermediate probabilities appear as a shade of gray as noted in the key in the figure. For GD and IGD, the Borg MOEA has a 100% attainment probability up to metric values within a distance of 0.02 of the reference set.  $\epsilon$ -MOEA, on the other hand, begins to have trouble reaching GD and IGD values within a distance of 0.1 of the reference set.

An even more dramatic difference is seen in AEI. Here, it is very unlikely that  $\epsilon$ -MOEA produces AEI values less than 0.5.  $\epsilon$ -MOEA can only reliably attain AEI values larger than 0.8.

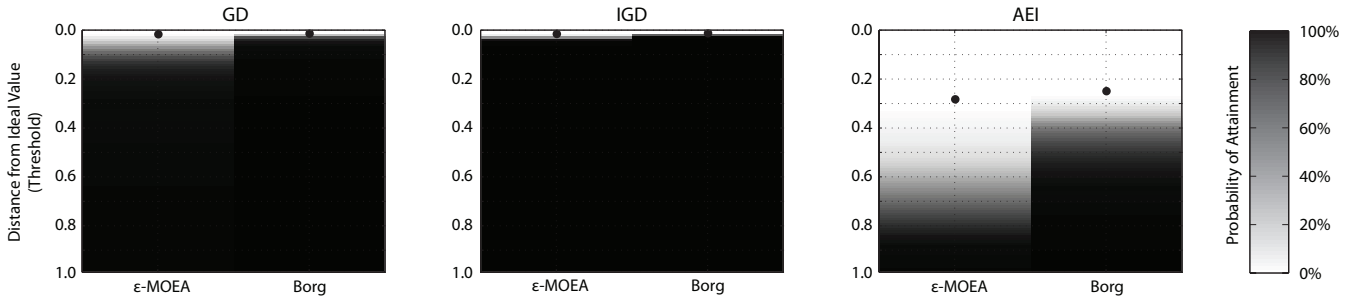


Fig. 3. Plots showing the best achieved metric value and probability of attainment for each performance metric. The y-axis ranges across the metric values from 0 to 1. The circle markers indicate the best achieved metric value by each algorithm. The shaded bars show the probability of each algorithm producing results which match or exceed a threshold. The threshold is the metric value in the y-axis. Black regions indicate 100% attainment; white regions indicate 0% attainment.

Recall that AEI is a measure of consistency. This result implies that  $\epsilon$ -MOEA is very inconsistent, and produces results that poorly approximate some portions of the reference set (i.e., it is prone to gaps in its approximation sets). The Borg MOEA provides more consistent results, showing strong probabilities up to AEI values of 0.25.

Similar in design to Figure 3, Figure 4 shows the efficiency of the algorithms. Here, the shading indicates the minimum NFE required for the algorithm to produce results meeting or exceeding the threshold of the y-axis with a high probability ( $\geq 90\%$  of the parameterizations sampled in a given band of NFE are successful in meeting or exceeding the metric threshold). We observe that the Borg MOEA exhibits substantially higher efficiency than  $\epsilon$ -MOEA. For GD and IGD, the Borg MOEA can produce results within a distance of 0.05 of the ideal metric value with as few as 50,000 NFE. For AEI, the Borg MOEA is dramatically more efficient and effective.  $\epsilon$ -MOEA requires more than 1,000,000 NFE to get within a distance of 0.8 of its ideal value. Figure 4 in combination with attainment results in Figure 3 show that beyond this point,  $\epsilon$ -MOEA is failing to attain any reliable search across its sampled parameterizations.

### B. Sobol' Sensitivity Analysis

Sobol' sensitivity analysis provides information about the importance of each individual MOEA parameter as well as its interactions with other parameters. Figure 5 shows the first-, second-, and total-order sensitivities in each plot with respect to their AEI performance. Around the outside of the plots, the filled circles correspond to each parameter of the algorithms. The size of the circle reflects the first-order sensitivity. A small circle indicates that the parameter has no effect on the performance of the algorithm, whereas a large circle indicates that the parameter has a significant effect on the algorithm. Strong first-order sensitivities are helpful if they exist because they distinguish which parameter(s) users should focus on when using that particular MOEA. The rings around the circles show total-order effects. Total-order sensitivities represent the fully interactive, non-separable multiparameter controls. Larger rings indicate larger total-order sensitivities. If the rings are significantly larger than the filled first-order

circles, then most of a parameter's influence emerges through its interactions with other parameters. The lines between the circles show second-order effects in Figure 5. Thicker lines indicate stronger second-order interaction between the two parameters.

Starting with  $\epsilon$ -MOEA, we see each parameter has small first-order sensitivities, moderate second-order sensitivities, yet large total-order sensitivities. Since the total-order sensitivity for a parameter is a sum of its first-order and all interactive sensitivities involving that parameter, this implies  $\epsilon$ -MOEA has many higher-order interactions among its parameters. In traditional non-adaptive MOEAs such as the  $\epsilon$ -MOEA such strong higher-order interactions among an algorithm's parameters suggest the algorithm is uncontrollable. It will be impossible to independently tune its parameters as they are all fully interdependent. This supports our prior observations that show traditional non-adaptive MOEA's parameter controls are dominantly interactive and change with problem type or dimension of the objective space even within the same problem [8]. This is a severe weakness for real-world application of non-adaptive MOEAs.

The Borg MOEA, on the other hand, shows a strong dependence on the maximum number of objective function evaluations. This does not imply it requires more function evaluations than  $\epsilon$ -MOEA; alternatively, it means that increasing the number of function evaluations is a clear and easy way to improve the Borg MOEA's performance. This result confirms previous observations of Hadka and Reed [8] on this real-world problem. This dependence is shown in the strong first-order sensitivities as well as the strong second-order interactions with other parameters. The remaining parameters have far less effect, showing only small amounts of first-, second- and total-order sensitivities. This suggests that the Borg MOEA is dramatically less sensitive to the parameterization of its operators. Figure 5 clearly shows that all of the Borg MOEA's search operators influence its overall performance given their strong interactions with the maximum number of evaluations. It is interesting to note that PCX, SBX and SPX do have some pairwise interactions, which indicate that the Borg MOEA's overall performance is influenced by how these operators work cooperatively.

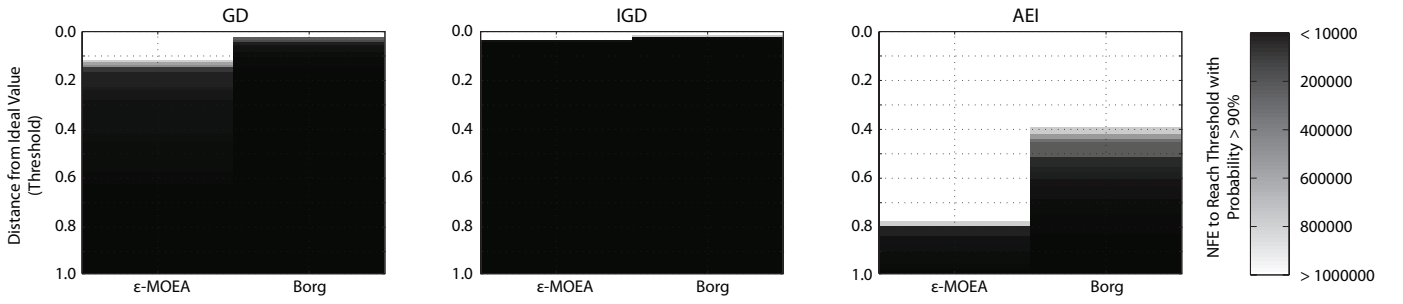


Fig. 4. Plots showing the efficiency for each performance metric. The y-axis ranges across the metric values from 0 to 1. The shaded bars show the minimum NFE required for each algorithm to match or exceed the threshold of the y-axis. Black regions indicate few NFE are required; white regions indicate more than 1,000,000 evaluations (the upper limit in this study) are required.

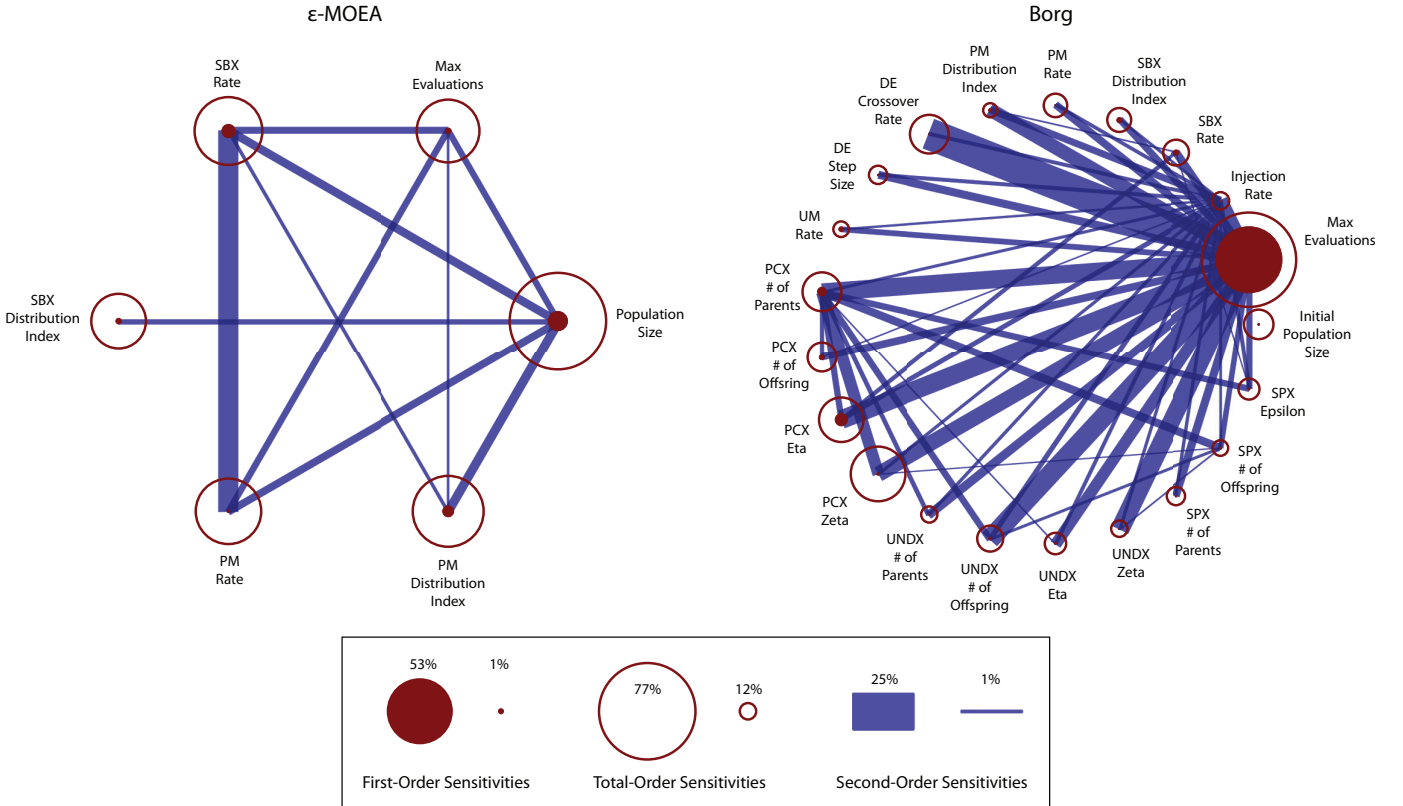


Fig. 5. First-, second- and total-order sensitivities between the parameters controlling  $\epsilon$ -MOEA and the Borg MOEA with respect to their AEI performance. The circles represent the first-order sensitivities of each parameter, where larger circles indicate the parameter has a strong impact on performance. Rings around each circle indicate total-order sensitivities, where larger rings indicate the parameter contributes many higher-order interactions. Lines between parameters indicate second-order sensitivities, where thicker lines indicate the two parameters interact strongly to affect performance.

### C. Auto-Adaptive Operator Probabilities

The Borg MOEA's auto-adaptive and cooperative multi-operator search can be further explored by analyzing the dynamics of its operator probabilities. Figure 6 shows traces from 50 seeds of the Borg MOEA using its default parameter settings (shown in Table III). The y-axis of each trace shows the probability each specific operator is used during a run of the Borg MOEA. Figure 6 shows that for the first 20,000 NFE, three operators are cooperating: SBX, PCX and SPX. Each operator is allocated approximately 30% by the auto-adaptive operation selection mechanism during this initial search phase.

This initial search phase accounts for the rapid convergence to the reference set, as observed in the efficiency results (see Figure 4). After 20,000 evaluations, PCX dominates. PCX's parent-centric behavior is well-suited for introducing small, beneficial perturbations to a design, allowing fine-tuning near the end of a run. Additionally, the strong influence from PCX can be observed in the sensitivity results in Figure 5, where PCX shows moderate first- and second-order sensitivities.

Combined with the results presented earlier, Figure 6 provides the first evidence of the beneficial effect of multiple search operators on a real-world problem.  $\epsilon$ -MOEA is limited



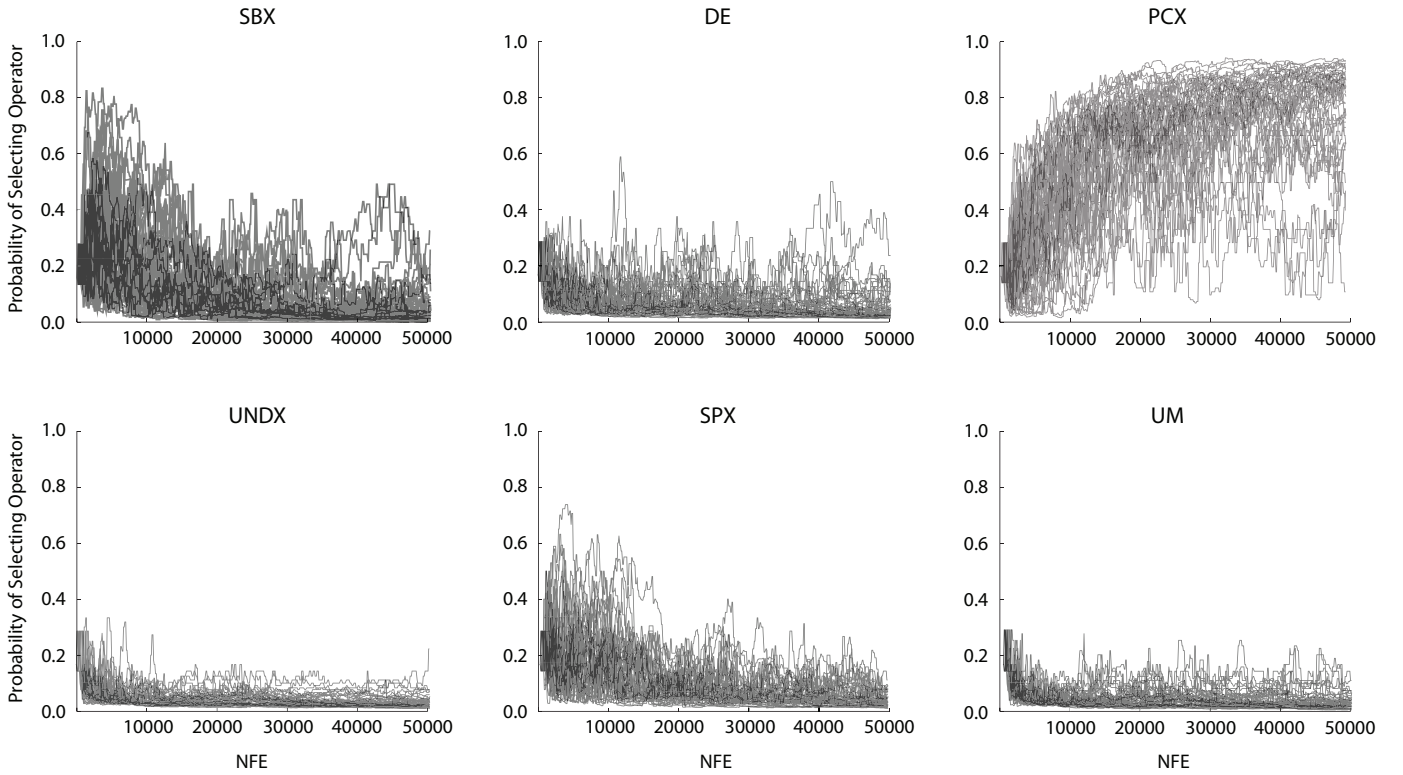


Fig. 6. Demonstration of the Borg MOEA’s auto-adaptive and cooperative multi-operator search, showing the operator probabilities from 50 seeds of the Borg MOEA using its default parameter settings (shown in Table III).

to the SBX operator, but Figure 6 shows that other search operators are more effective on this problem. Furthermore, no single operator was sufficient. It is both the individual contributions from one or more operators and their interactions that lead to the behavior seen in the Borg MOEA. Figure 5 shows that even non-dominant operators like DE, UNDX and UM have second-order interactions with NFE that influence the Borg MOEA’s final performance across its parameterizations.

#### IV. CONCLUSION

In this study, we characterized the effects of the enhancements introduced in the Borg MOEA over its predecessor, the  $\epsilon$ -MOEA, on the GAA product family design problem. This study also provides the first full Sobol’ diagnostic assessment of the Borg MOEA on a severely challenging, real-world, 10-objective test problem. The results show that the enhancements proposed by the Borg MOEA significantly improve reference set coverage and increase the probability of producing high-quality results. Such gains are critical in real-world scenarios, as the decision maker can be confident that the Borg MOEA is producing high-quality solutions with a high probability with minimal sensitivities to its parameters.

Our results confirm the sensitivities first identified by Hadka and Reed [8] on a number of analytical test problems. The Borg MOEA’s performance is highly efficient and can be easily improved by increasing its runtime (i.e., the number of objective function evaluations). This implies two important conclusions. First, by not relying heavily on other parameters,

the Borg MOEA is very controllable. The user need not be concerned with parameterization and must only allocate a sufficient amount of processing time in order to produce high-quality results. Second, its dependency on runtime suggests that the Borg MOEA will benefit greatly from parallelization strategies. A simple master-slave architecture will increase the number of objective function evaluations available to the algorithm and, consequently, will improve the quality and reliability of the results. Although the Borg MOEA is dependent on runtime, the efficiency measure demonstrates that significantly fewer NFE are required relative to  $\epsilon$ -MOEA to produce near-optimal results with high likelihood. This confirms prior work in showing that the Borg MOEA is highly efficient in attaining high quality Pareto approximation sets in a limited number of evaluations.

Finally, we observed that combinations of operators were active in the Borg MOEA throughout its search. This confirms observations by Vrugt et al. [27], [28] that multiple operators benefit multiobjective optimization. While we identified SBX, PCX and SPX as the dominant search operators for the Borg MOEA, it is important to note that this does not necessarily imply that the remaining three operators are unnecessary. While DE, UNDX and UM were not selected with high probability, they do contribute to the result by periodically producing new solutions as represented in these operators second-order interactions with run duration. Overall this study demonstrates that the Borg MOEA is highly controllable in challenging

real-world applications and can, consequently, dramatically increase the size and scope of problems that can be effectively addressed. Future work entails tackling other challenging product family design problems and computationally-intensive engineering design challenges encountered in complex systems design.

## REFERENCES

- [1] S. Venkataraman and R. T. Haftka, "Structural optimization complexity: What has moore's law done for us?" *Structural and Multidisciplinary Optimization*, vol. 28, no. 6, pp. 375–387, 2004.
- [2] R. J. Balling, J. T. Taber, M. R. Brown, and K. Day, "Multiobjective urban planning using genetic algorithms," *Journal of Urban Planning and Development*, vol. 125, no. 2, pp. 86–99, 1999.
- [3] J. B. Kollat and P. M. Reed, "A computational scaling analysis of multi-objective evolutionary algorithms in long-term groundwater monitoring applications," *Advances in Water Resources*, vol. 30, no. 3, pp. 408–419, 2007.
- [4] T. W. Simpson and J. R. R. A. Martins, "The future of multidisciplinary design optimization (mdo): Advancing the design of complex engineered systems," National Science Foundation, Arlington, VA, Tech. Rep., 2010.
- [5] C. L. Bloebaum and A. M. R. McGowan, "Design of complex engineered systems," *Journal of Mechanical Design*, vol. 132, no. 12, pp. 1–2, 2010.
- [6] C. A. Coello Coello, G. B. Lamont, and D. A. Van Veldhuizen, *Evolutionary Algorithms for Solving Multi-Objective Problems*. New York, NY, USA: Springer Science+Business Media, LLC, 2007.
- [7] R. C. Purshouse and P. J. Fleming, "On the evolutionary optimization of many conflicting objectives," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 770–784, 2007.
- [8] D. Hadka and P. Reed, "Diagnostic assessment of search controls and failure modes in many-objective evolutionary optimization," *Evolutionary Computation*, 2012.
- [9] P. M. Reed, D. M. Hadka, J. D. Herman, J. R. Kasprzyk, and J. B. Kollat, "Evolutionary multiobjective optimization in water resources: The past, present, and future," *Advances in Water Resources*, 2012.
- [10] D. Hadka and P. Reed, "Borg: An auto-adaptive many-objective evolutionary computing framework," *Evolutionary Computation (In Review)*, 2012.
- [11] D. E. Goldberg, *Design of Innovation: Lessons From and For Competent Genetic Algorithms*. Norwell, MA: Kluwer Academic Publishers, 2002.
- [12] I. M. Sobol', "Global sensitivity indices for nonlinear mathematical models and their monte carlo estimates," *Mathematics and Computers in Simulation*, vol. 55, pp. 271–280, 2001.
- [13] A. Saltelli, "Making best use of model evaluations to compute sensitivity indices," *Computer Physics Communications*, vol. 145, pp. 280–297, 2002.
- [14] I. M. Sobol' and S. S. Kucherenko, "Global sensitivity indices for nonlinear mathematical models," *Wilmott magazine*, vol. 2, pp. 1–6, 2005.
- [15] T. W. Simpson, W. Chen, J. K. Allen, and F. Mistree, "Conceptual design of a family of products through the use of the robust concept exploration method," in *6th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, vol. 2, 1996, pp. 1535–1545.
- [16] T. W. Simpson, "Methods for optimizing product platforms and product families: Overview and classification," in *Product Platform and Product Family Design: Methods and Applications*. New York: Springer, 2005, pp. 133–156.
- [17] B. D'Souza and T. W. Simpson, "A genetic algorithm based method for product family design optimization," *Engineering Optimization*, vol. 35, no. 1, pp. 1–18, 2003.
- [18] T. W. Simpson and B. D'Souza, "Assessing variable levels of platform commonality within a product family using a multiobjective genetic algorithm," *Concurrent Engineering: Research and Applications*, vol. 12, no. 2, pp. 119–130, 2004.
- [19] R. Shah, P. M. Reed, and T. Simpson, "Many-objective evolutionary optimization and visual analytics for product family design," *Multi-objective Evolutionary Optimisation for Product Design and Manufacturing*, 2011.
- [20] J. B. Kollat and P. M. Reed, "Comparison of multi-objective evolutionary algorithms for long-term monitoring design," *Advances in Water Resources*, vol. 29, no. 6, pp. 792–807, 2006.
- [21] K. Deb, M. Mohan, and S. Mishra, "A fast multi-objective evolutionary algorithm for finding well-spread Pareto-optimal solutions," Kanpur Genetic Algorithms Laboratory (KanGAL), KanGAL Report No. 2003002, 2003.
- [22] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler, "Combining convergence and diversity in evolutionary multi-objective optimization," *Evolutionary Computation*, vol. 10, no. 3, 2002.
- [23] J. Horn, "The nature of niching: Genetic algorithms and the evolution of optimal, cooperative populations," Ph.D. dissertation, University of Illinois, 1995.
- [24] S. W. Mahfoud, "Niching methods for genetic algorithms," Ph.D. dissertation, University of Illinois, 1995.
- [25] Y. Tang, P. Reed, and T. Wagener, "How effective and efficient are multiobjective evolutionary algorithms at hydrologic model calibration?" *Hydrology and Earth System Science*, vol. 10, pp. 289–307, 2006.
- [26] R. J. Moral, D. Sahoo, and G. S. Dulikravich, "Multi-objective hybrid evolutionary optimization with automatic switching," in *11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Portsmouth, VA, Sept. 2006.
- [27] J. A. Vrugt and B. A. Robinson, "Improved evolutionary optimization from genetically adaptive multimethod search," *Proceedings of the National Academy of Sciences*, vol. 104, no. 3, pp. 708–711, 2007.
- [28] J. A. Vrugt, B. A. Robinson, and J. M. Hyman, "Self-adaptive multimethod search for global optimization in real-parameter spaces," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 2, pp. 243–259, April 2009.
- [29] K. Deb and R. B. Agrawal, "Simulated binary crossover for continuous search space," Indian Institute of Technology, Kanpur, UP, India, Tech. Rep. IITK/ME/SMD-94027, Nov. 1994.
- [30] R. Storn and K. Price, "Differential evolution — a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.
- [31] K. Deb, D. Joshi, and A. Anand, "Real-coded evolutionary algorithms with parent-centric recombination," *Computational Intelligence, Proceedings of the World on Congress on*, vol. 1, pp. 61–66, 2002.
- [32] S. Tsutsui, M. Yamamura, and T. Higuchi, "Multi-parent recombination with simplex crossover in real coded genetic algorithms," in *Genetic and Evolutionary Computation Conference (GECCO 1999)*, 1999.
- [33] H. Kita, I. Ono, and S. Kobayashi, "Multi-parental extension of the unimodal normal distribution crossover for real-coded genetic algorithms," in *Congress on Evolutionary Computation*, 1999, pp. 1581–1588.
- [34] N. Srinivas and K. Deb, "Multiobjective optimization using nondominated sorting in genetic algorithms," *Evolutionary Computation*, vol. 2, no. 3, pp. 221–248, 1994.
- [35] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, and S. Tarantola, *Global Sensitivity Analysis: The Primer*. Wiley, Jan. 2008.
- [36] C. M. Fonseca and P. J. Fleming, "On the performance assessment and comparison of stochastic multiobjective optimizers," in *Parallel Problem Solving from Nature (PPSN IV)*. London, UK: Springer-Verlag, 1996, pp. 584–593.