A Non-Parametric Statistical Dominance Operator for Noisy Multiobjective Optimization

Dung H. Phan and Junichi Suzuki

Deptartment of Computer Science
University of Massachusetts, Boston, USA
{phdung, jxs}@cs.umb.edu

Abstract. This paper describes and evaluates a new noise-aware dominance operator for evolutionary algorithms to solve the multiobjective optimization problems (MOPs) that contain noise in their objective functions. This operator is designed with the Mann-Whitney U-test, which is a non-parametric (i.e., distribution-free) statistical significance test. It takes objective value samples of given two individuals, performs a U-test on the two sample sets and determines which individual is statistically superior. Experimental results show that it operates reliably in noisy MOPs and outperforms existing noise-aware dominance operators particularly when many outliers exist under asymmetric noise distributions.

Keywords: Evolutionary multiobjective optimization algorithms, noisy optimization, uncertainties in objective functions

1 Introduction

This paper focuses on noisy multiobjective optimization problems (MOPs):

$$\left.\begin{array}{r} \min\ F(\boldsymbol{x}) = [f_1(\boldsymbol{x}) + \epsilon_1, \cdots, f_m(\boldsymbol{x}) + \epsilon_m]^T \in \mathcal{O} \\ \text{subject to } \boldsymbol{x} = [x_1, x_2, \cdots, x_n]^T \in \mathcal{S} \end{array}\right\} \quad (1)$$

$\mathcal{S}$ denotes the decision variable space. $\boldsymbol{x} \in \mathcal{S}$ is a solution candidate that consists of n decision variables. It is called an *individual* in evolutionary multiobjective optimization algorithms (EMOAs). $F : \mathbb{R}^n \rightarrow \mathbb{R}^m$ consists of m real-value objective functions, which produce the objective values of $\boldsymbol{x}$ in the objective space $\mathcal{O}$. In MOPs, objective functions often conflict with each other. Thus, there rarely exists a single solution that is optimum with respect to all objectives. As a result, EMOAs often seek the optimal trade-off solutions, or *Pareto-optimal* solutions, by balancing the trade-offs among conflicting objectives. A notion of *Pareto dominance* plays an important role to seek Pareto optimality in EMOAs. An individual $\boldsymbol{x} \in \mathcal{S}$ is said to *dominate* another individual $\boldsymbol{y} \in \mathcal{S}$ (denoted by $\boldsymbol{x} \succ \boldsymbol{y}$) iif the both of the following two conditions are hold: (1) $f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{y})\ \forall\ i\ = 1, \cdots, m$ and (2) $f_i(\boldsymbol{x}) < f_i(\boldsymbol{y})\ \exists\ i\ =\ 1, \cdots, m$.

In Eq. 1, ϵ_i is a random variable that represents noise in the i-th objective function. Given noise, each objective function can yield different objective values for the same individual from time to time. Noise in objective functions often interferes with a dominance operator, which determines dominance relationships among individuals. For example, a dominance operator may mistakenly judge that an inferior individual dominates an superior one. Defects in a dominance operator significantly degrade the performance to solve MOPs [2, 11].

In order to address this issue, this paper proposes a new noise-aware dominance relationship, called *U-dominance*, which extends the classical Pareto dominance. It is designed with the Mann-Whitney U-test (U-test in short), which is a non-parametric (i.e., distribution free) statistical significance test [12]. The U-dominance operator takes objective value samples of given two individuals, performs a U-test on the two sample sets to examine whether it is statistically confident enough to judge a dominance relationship between the two individuals, and determines which individual is statistically superior/inferior. This paper evaluates the U-dominance operator by integrating it with NSGA-II [4], a well-known EMOA. Experimental results demonstrate that the U-dominance operator reliably performs dominance ranking operation in noisy MOPs and outperforms existing noise-aware dominance operators particularly when many outliers exist under asymmetric noise distributions.

2 Related Work

There exist various existing work to handle uncertainties in objective functions by modifying the classical Pareto dominance operator [2, 11]. Most of them assume particular noise distributions in advance; for example, normal distributions [1, 8–10, 13], uniform distributions [14] and Poisson distributions [6, 17]. Given a noise distribution, existing noise-aware dominance operators collect objective value samples from each individual [6, 8, 9, 14, 17], or each cluster of individuals [1], in order to determine dominance relationships among individuals. Those existing operators are susceptible to noisy MOPs in which noise follows unknown distributions. In contrast, the U-dominance operator assumes no noise distributions in advance because, in general, it is hard to predict and model them in many (particularly, real-world) MOPs. Instead of estimating each individual's objective values based on a given noise distribution, the U-dominance operator estimates the impacts of noise on objective value samples and determines whether it is statistically confident enough to compare individuals.

Voß et al. [16] and Boonma et al. [3,18] study similar dominance operators to the U-dominance operator in that they assume no noise distributions in objective value samples of each individual and statistically examine the impacts of noise on those samples. Voß et al. propose an operator that considers an uncertainty zone around the median of samples on a per-objective basis. A dominance decision is made between two individuals only when their uncertainty zones do not overlap in all objectives. The uncertainty zone can be the inter-quartile range (IQR) of samples or the bounding box based on the upper and lower quartiles (BBQ) of samples. Unlike the U-dominance operator, this operator is susceptible to high noise strength and asymmetric heavy-tailed noise distributions. Boonma et al. propose an operator that classifies objective value samples with a support vector

Algorithm 1 The U-Dominance Operator: `uDominance(`$\mathcal{A}$`, `$\mathcal{B}$`, `α`)`

Input: $\mathcal{A}$ and $\mathcal{B}$, Objective value samples of individuals $\boldsymbol{a}$ and $\boldsymbol{b}$, respectively
Input: α, The confidence level used in a U-test
Output: Dominance relationship between $\boldsymbol{a}$ and $\boldsymbol{b}$

$p_{\mathcal{A}}, p_{\mathcal{B}} = 1$
for each (the i-th) objective **do**
  Perform a U-test on $\mathcal{A}$ and $\mathcal{B}$ in the i-th objective.
  if $\mathcal{A}$ is superior than $\mathcal{B}$ with the confidence level α in the i-th objective **then**
    $p_{\mathcal{A}} = 1 * p_{\mathcal{A}}$
    $p_{\mathcal{B}} = 0$
  else if $\mathcal{B}$ is superior than $\mathcal{A}$ with the confidence level α in the i-th objective **then**
    $p_{\mathcal{A}} = 0$
    $p_{\mathcal{B}} = 1 * p_{\mathcal{B}}$
  end if
end for
if $p_{\mathcal{A}} = 1$ **and** $p_{\mathcal{B}} = 0$ **then**
  return 1 /*** $\boldsymbol{a}$ *U-dominates* $\boldsymbol{b}$. ***/
else if $p_{\mathcal{A}} = 0$ **and** $p_{\mathcal{B}} = 1$ **then**
  return -1 /*** $\boldsymbol{b}$ *U-dominates* $\boldsymbol{a}$. ***/
else
  return 0 /*** $\boldsymbol{a}$ *and* $\boldsymbol{b}$ *are non-U-dominated.* ***/
end if

machine, which can be computationally expensive. In contrast, the U-dominance operator is designed lightweight; it requires no classification and learning.

3 The U-Dominance Operator

U-dominance is a new noise-aware dominance that is designed with the Mann-Whitney U-test (U-test in short), which is a non-parametric (i.e., distribution-free) statistical significance test [12]. An individual $\boldsymbol{a}$ is said to *U-dominate* an individual $\boldsymbol{b}$ (denoted by $\boldsymbol{a} \succ_U \boldsymbol{b}$) with the confidence level α iif:

- In all objectives, $\boldsymbol{b}$ is not superior than $\boldsymbol{a}$ using a U-test with the confidence level of α.
- In at least one objective, $\boldsymbol{a}$ is superior than $\boldsymbol{b}$ using a U-test with the confidence level of α.

The U-dominance operator takes objective value samples of given two individuals, estimates the impacts of noise on the samples through a U-test, examines whether it is statistically confident enough to judge a dominance relationship between the two individuals, and determines which individual is statistically superior/inferior (Algorithm 1).

Given two sets of objective value samples, $\mathcal{A}$ and $\mathcal{B}$, which are collected from two individuals $\boldsymbol{a}$ and $\boldsymbol{b}$, a U-test is performed on $\mathcal{A}$ and $\mathcal{B}$ in each objective (Algorithm 1). First, $\mathcal{A}$ and $\mathcal{B}$ are combined into a set $\mathcal{S} = \mathcal{A} \cup \mathcal{B}$. The samples in $\mathcal{S}$ are sorted in ascending order based on their objective values in an objective in question. Then, each sample obtains its *rank*, which represents the sample's position in $\mathcal{S}$. The rank of one is given to the first sample in $\mathcal{S}$ (i.e., the best sample that has the minimum objective value). If multiple samples tie, they receive the same rank, which is equal to the mean of their positions in $\mathcal{S}$. For example, if the first two samples tie in $\mathcal{S}$, they receive the rank of 1.5 ($\frac{1+2}{2}$).

Once ranks are assigned to samples, the *rank-sum* values, $R_{\mathcal{A}}$ and $R_{\mathcal{B}}$, are computed for $\mathcal{A}$ and $\mathcal{B}$, respectively. ($R_{\mathcal{A}}$ sums up the ranks for the samples in $\mathcal{A}$.) For a large sample size (> 10), the sampling distributions of $R_{\mathcal{A}}$ and $R_{\mathcal{B}}$ are approximately normal [12]. Therefore, the standardized value of a rank-sum is a standard normal deviate whose significance can be tested under the standard normal distribution. The standardized value of $R_{\mathcal{A}}$ is given as follows.

$$z_{R_{\mathcal{A}}} = \frac{R_{\mathcal{A}} - \mu_{R_{\mathcal{A}}}}{\sigma_{R_{\mathcal{A}}}} \tag{2}$$

$\mu_{R_{\mathcal{A}}}$ and $\sigma_{R_{\mathcal{A}}}$ denote the mean and standard deviation of $R_{\mathcal{A}}$, respectively.

$$\mu_{R_{\mathcal{A}}} = \frac{|\mathcal{A}| \times (|\mathcal{A}| + |\mathcal{B}| + 1)}{2} \tag{3}$$

$$\sigma_{R_{\mathcal{A}}} = \sqrt{\frac{|\mathcal{A}| \times |\mathcal{B}| \times (|\mathcal{A}| + |\mathcal{B}| + 1)}{12}} \tag{4}$$

With the confidence level of α, the U-test determines that $\mathcal{A}$ and $\mathcal{B}$ are not drawn from the same distribution if $F(z_{R_{\mathcal{A}}}) \leq (1-\alpha)$ or $F(z_{R_{\mathcal{A}}}) \geq \alpha$. ($F(z)$ is the cumulative distribution function of the standard normal distribution.) This means that $\mathcal{A}$ and $\mathcal{B}$ are significantly different with the confidence level of α. The U-test concludes that $\boldsymbol{a}$ is superior than $\boldsymbol{b}$ with respect to an objective in question if $F(z_{R_{\mathcal{A}}}) \leq (1-\alpha)$ and that $\boldsymbol{b}$ is superior than $\boldsymbol{a}$ if $F(z_{R_{\mathcal{A}}}) \geq \alpha$.

This paper integrates the U-dominance operator with NSGA-II [4], a well-known EMOA. It is integrated with a binary tournament operator and a dominance ranking operators in NSGA-II. NSGA-II uses binary tournament in its parent selection process, which selects a parent individual to be used in crossover, and uses dominance ranking in its environmental selection process, which selects the next-generation population from the union of the current population and its offspring [4]. Fig. 1 shows how to perform binary tournament with with the U-dominance operator. In Lines 1 and 2, two individuals $\boldsymbol{a}$ and $\boldsymbol{b}$ are randomly drawn from the population $\mathcal{P}$. Then, in Lines 3 and 4, their objective value samples are obtained to invoke the U-dominance operator at Line 5. Based on the U-dominance relationship between $\boldsymbol{a}$ and $\boldsymbol{b}$, one of them is returned as a parent individual (Lines 6 to 16).

Fig. 2 shows how to rank individuals with the U-dominance operator. From Line 1 to 12, U-dominance relationships are determined among N individuals in the population $\mathcal{P}$. The U-dominance operator is invoked in Line 5. Unlike the classical Pareto dominance, U-dominance relationships are not transitive. When $\boldsymbol{a} \succ_U \boldsymbol{b}$ and $\boldsymbol{b} \succ_U \boldsymbol{c}$, $\boldsymbol{a} \succ_U \boldsymbol{c}$ is not guaranteed. When objective functions contain high-level noise, $\boldsymbol{c}$ might even U-dominate $\boldsymbol{a}$. If a loop exists in U-dominance relationships (e.g., $\boldsymbol{a} \succ_U \boldsymbol{b}$, $\boldsymbol{b} \succ_U \boldsymbol{c}$ and $\boldsymbol{c} \succ_U \boldsymbol{a}$), the U-dominance operator deletes the U-dominance relationships among $\boldsymbol{a}$, $\boldsymbol{b}$ and $\boldsymbol{c}$, and concludes that they are non-U-dominated with each other (Line 13 to 15 in Algorithm 2).

Input: $\mathcal{P}$, The population of N individuals
Output: A parent individual to be used in crossover

```
a = randomSelection(P)
b = randomSelection(P)
A = samplesOf(a)
B = samplesOf(b)
r = uDominance(A, B, α)
if r = 1 then
    return a
else if r = −1 then
    return b
else
    if random() > 0.5 then
        return a
    else
        return b
    end if
end if
```

Fig. 1. Binary Tournament

Input: $\mathcal{P}$, The population of N individuals
Output: $\mathcal{F}$, Ranked and sorted N individuals

```
for each p ∈ P do
    for each q ∈ P do
        P = samplesOf(p)
        Q = samplesOf(q)
        r = uDominance(P, Q, α)
        if r = 1 then
            S_p = S_p ∪ {p}
        else if r = −1 then
            n_p = n_p + 1
        end if
    end for
end for
for each p ∈ P do
    clearDominanceRelationLoop(p)
end for
for each p ∈ P do
    if n_p = 0 then
        F_1 = F_1 ∪ {p}
    end if
end for
i = 1
while F_i ≠ ∅ do
    H = ∅
    for each p ∈ F_i do
        for each q ∈ S_p do
            n_q = n_q − 1
            if n_q = 0 then
                H = H ∪ {q}
            end if
        end for
    end for
    i = i + 1
    F_i = H
end while
return F
```

Fig. 2. U-dominance Ranking

4 Experimental Evaluation

This section evaluates the U-dominance operator by integrating it with NSGA-II. This variant of NSGA-II is called NSGA-II-U in this paper. It is compared with the following five other variants of NSGA-II.

- NSGA-II: The original NSGA-II. It takes only one sample and uses its objective values in the default Pareto dominance operator. It never considers noise in its dominance operator.
- NSGA-II-Median: takes multiple samples, obtains median values in different objectives and use them in NSGA-II's default dominance operator.
- NSGA-II-Mean: takes multiple samples, obtains mean values in different objectives and use them in NSGA-II's default dominance operator.
- NSGA-II-N: replaces NSGA-II's default dominance operator with a noise-aware dominance operator proposed in [8]. This noise-aware operator assumes Gaussian noise in objective functions in advance (c.f. Section 2).
- NSGA-II-IQR: replaces NSGA-II's default dominance operator with a noise-aware dominance operator proposed in [16] (c.f. Section 2).

All NSGA-II variants are evaluated with ZDT and DTLZ family problems (10 problems in total) [5, 19]. Experiments were configured as shown in Table 1 and conducted with jMetal [7]. The total number of generations in each experiment is 200 in ZDT problems, 500 in DTLZ3 and 250 in the other DTLZ problems. Every experimental result is obtained from 20 independent experiments.

Table 1. Experimental Configurations

Parameter	Value	Parameter	Value
Confidence level α	0.55	Population size N	100
# of samples per individual	20	Crossover rate	0.9
γ (Eq. 8)	3	Mutation rate	1/(# of decision variables)
Noise strength β (Eq. 6–9)	0.1 or 0.5	Total # of generations	200, 250 or 500

This paper uses the hypervolume ratio (HVR) metric to compare NSGA-II variants and evaluate the U-dominance operator. HVR is calculated as the ratio of the hypervolume (HV) of non-dominated individuals (D) to the hypervolume of Pareto-optimal solutions (P^*) [15].

$$HVR(D) = \frac{HV(D)}{HV(P^*)} \tag{5}$$

HV measures the union of the volumes that non-dominated individuals dominate. Thus, HVR quantifies the optimality and diversity of non-dominated individuals D. A higher HVR indicates that non-dominated individuals are closer to the Pareto-optimal front and more diverse in the objective space.

In order to turn ZDT and DTLZ family problems to be noisy problems, this paper defines four kinds of additive noise in objective functions (c.f. Eq. 1).

- **Gaussian noise:** This noise distribution is characterized with a symmetric shape and a very limited number of outliners.

$$\epsilon_i = \beta \mathcal{N}(0,1) \tag{6}$$

  $\mathcal{N}(0,1)$ is the standard normal (or Gaussian) distribution.
- **Cauchy noise:** This noise distribution is used to generate more outliers than the Gaussian distribution does.

$$\epsilon_i = \beta \frac{\mathcal{N}(0,1)}{\mathcal{N}(0,1)+e} \tag{7}$$

  e is set to be a very small value in order to prevent division by zero.
- **Chi-squared noise:** This distribution is asymmetric and heavy-tailed in contrast to Gaussian and Cauchy distributions. It contains outliers.

$$\epsilon_i = \beta \sum_{i=1}^{\gamma} \mathcal{N}_i(0,1)^2 \tag{8}$$

– **Log-normal noise:** This distribution is characterized with an asymmetric and heavy-tailed shape and outliers.

$$\epsilon_i = \beta \times exp(\mathcal{N}(0, 1)) \tag{9}$$

4.1 Experimental Results

Tables 2 to 5 show the average HVR values that six EMOAs yield at the last generation in 10 different test problems with different noise distributions. In each table, a number in parentheses indicates a standard deviation among different experiments. A bold number indicates the best result among six algorithms on a per-row basis. A double star (**) or a single star (*) is placed for an average HVR result when the result is significantly different from NSGA-II-U's result based on a single-tail t-test. A double star is placed with the confidence level of 99% while a single star is placed with the confidence level of 95%.

Table 2 shows the experimental results under Gaussian noise. NSGA-II-U clearly outperforms NSGA-II, NSGAII-Median and NSGA-II-IQR in all problems except ZDT2. NSGA-II-Mean and NSGA-II-N are more competitive against NSGA-II-U because noise follows a normal distribution and the distribution is symmetric. NSGA-II-U significantly outperforms NSGA-II-Mean and NSGA-II-N in DTLZ1 and DTLZ3 with the confidence level of 99% while the three EMOAs perform similarly in the other problems.

Table 3 shows the results under Cauchy noise. NSGA-II-U clearly outperforms NSGA-II, NSGA-II-Mean, NSGA-II-N and NSGA-II-IQR. In contrast to Table 2, NSGA-II-Median outperforms NSGA-II-Mean because Cauchy noise contains a lot of outliers. NSGA-II-U significantly outperforms NSGA-II-Median in DTLZ1 and DTLZ3 with the confidence level of 99% while NSGA-II-Median yields similar or better performance than NSGA-II-U in the other problems.

Under chi-squared noise (Table 4) and lognormal noise (Table 5), NSGA-II-U significantly outperforms five other EMOAs in almost all problems except ZDT2. In ZDT2, NSGA-II-Median performs better than NSGA-II-U when noise strength is 0.5. However, there is no significant difference between the two algorithms when noise strength is 0.1. Tables 4 and 5 demonstrate that the U-dominance operator reliably operates when many outliers exist in objective value samples under asymmetric noise distributions.

5 Conclusions

This paper proposes and evaluates a new noise-aware dominance operator, called the U-dominance operator, which never assumes noise distributions in advance by leveraging the Mann-Whitney U-test. Experimental results show that it operates reliably in noisy MOPs and outperforms existing noise-aware dominance operators particularly when many outliers exist in objective value samples under asymmetric noise distributions.

Table 2. HVR Results under Gaussian Noise

	β	NSGA-II	NSGA-II-Median	NSGA-II-Mean	NSGA-II-N	NSGA-II-IQR	NSGA-II-U
ZDT1	0.1	0.732(0.050)**	0.922(0.007)**	0.931(0.007)	0.923(0.006)**	0(0)**	**0.932**(0.006)
	0.5	0.004(0.015)**	0.604(0.066)**	**0.690**(0.045)	0.652(0.050)	0(0)**	0.684(0.056)
ZDT2	0.1	0.086(0.177)	0.716(0.264)**	**0.813**(0.150)**	0.347(0.404)	0(0)**	0.256(0.386)
	0.5	0(0)	0.010(0.044)	**0.024**(0.059)	0(0)	0(0)	0(0)
ZDT3	0.1	0.814(0.045)**	0.948(0.006)	0.951(0.018)	0.945(0.022)	0.003(0.004)**	**0.952**(0.006)
	0.5	0.066(0.066)**	0.679(0.108)	**0.757**(0.081)	0.731(0.077)	0(0)**	0.716(0.087)
ZDT4	0.1	0.764(0.139)**	0.908(0.105)	**0.938**(0.016)	0.839(0.134)*	0(0)**	0.917(0.053)
	0.5	0.004(0.019)**	0.558(0.197)*	0.679(0.163)	0.672(0.191)	0(0)**	**0.685**(0.181)
ZDT6	0.1	0.233(0.077)**	0.698(0.028)**	0.721(0.031)	0.691(0.033)**	0(0)**	**0.732**(0.023)
	0.5	0(0)**	0.067(0.055)*	**0.181**(0.094)*	0.071(0.054)*	0(0)**	0.112(0.073)
DTLZ1	0.1	0.022(0.056)**	0.397(0.390)**	0.477(0.369)**	0.546(0.353)**	0(0)**	**0.895**(0.010)
	0.5	0(0)**	0(0)**	0.007(0.032)**	0.014(0.055)**	0(0)**	**0.519**(0.138)
DTLZ2	0.1	0.319(0.101)	0.756(0.0141)	0.775(0.011)	0.736(0.020)	0(0)**	**0.780**(0.014)
	0.5	0(0)	0.006(0.015)	0.070(0.091)**	**0.204**(0.132)**	0(0)	0.002(0.008)
DTLZ3	0.1	0.015(0.069)**	0.007(0.024)**	0.059(0.178)**	0(0)**	0(0)**	**0.725**(0.151)
	0.5	0(0)**	0.002(0.011)**	0.009(0.030)**	0(0)**	0(0)**	**0.417**(0.151)
DTLZ4	0.1	0.469(0.122)**	0.801(0.131)	**0.862**(0.010)	0.810(0.091)	0(0)**	0.819(0.134)
	0.5	0.046(0.058)**	0.354(0.106)	**0.407**(0.093)	0.474(0.088)**	0(0)**	0.373(0.137)
DTLZ7	0.1	0.234(0.050)**	0.733(0.037)**	0.764(0.032)	0.728(0.035)**	0(0)**	**0.770**(0.030)
	0.5	0(0)**	0.153(0.063)**	**0.229**(0.059)**	0.182(0.059)**	0(0)**	0.069(0.084)

References

1. M. Babbar, A. Lakshmikantha, and D. Goldberg. A modified NSGA-II to solve noisy multiobjective problems. In *Proc. ACM Genet. Evol. Computat. Conf.*, 2003.
2. L. Bianchi, M. Dorigo, L. Gambardella, and W. J. Gutjahr. A survey on metaheuristics for stochastic combinatorial optimization. *Nat. Comput.*, 8(2), 2009.
3. P. Boonma and J. Suzuki. A confidence-based dominance operator in evolutionary algorithms for noisy multiobjective optimization problems. In *Proc. IEEE Int'l Conference on Tools with Artificial Intelligence*, 2009.
4. K. Deb, S. Agrawal, A. Pratab, and T. Meyarivan. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In *Proc. Int'l Conference on Parallel Problem Solving from Nature*, 2000.
5. K. Deb, L. Thiele, M. Laumanns, and E. Zitzler. Scalable test problems for evolutionary multiobjective optimization. In A. Abraham, R. Jain, and R. Goldberg, editors, *Evolutionary Multiobjective Optimization*. Springer, 2005.
6. K. Delibrasis, P. Undrill, and G. Cameron. Genetic algorithm implementation of stack filter design for image restoration. In *Proc. Vis., Image, Sign. Proc.*, 1996.
7. J. Durillo, A. Nebro, and E. Alba. The jMetal framework for multi-objective optimization: Design and architecture. In *Proc. IEEE Congress on Evol. Computat.*, 2010.
8. H. Eskandari, C. Geiger, and R. Bird. Handling uncertainty in evolutionary multiobjective optimization: SPGA. In *Proc. IEEE Congress Evol. Computat.*, 2007.
9. C. K. Goh and K. C. Tan. Noise handling in evolutionary multi-objective optimization. In *Proc. of IEEE Congress on Evolutionary Computation*, 2006.
10. E. Hughes. Evolutionary multi-objective ranking with uncertainty and noise. In *Proc. Int'l Conf. on Evolutionary Multi-Criterion Optimization*, 2001.
11. Y. Jin and J. Branke. Evolutionary optimization in uncertain environments-a survey. *IEEE Trans. Evol. Computat.*, 9(3), 2005.
12. H. Mann and D. Whitney. On a test of whether one of two random variables is stochastically larger than the other. *Annals of Math. Stat.*, 18(1), 1947.

Table 3. HVR Results under Cauchy Noise

	β	NSGA-II	NSGA-II-Median	NSGA-II-Mean	NSGA-II-N	NSGA-II-IQR	NSGA-II-U
ZDT1	0.1	0.682(0.054)**	**0.924**(0.005)**	0.700(0.040)**	0.719(0.039)**	0.0(0.001)**	0.917(0.009)
	0.5	0.160(0.086)**	**0.727**(0.031)**	0.174(0.068)**	0.195(0.085)**	0(0)**	0.676(0.043)
ZDT2	0.1	0.307(0.197)	**0.819**(0.193)**	0.214(0.228)*	0.039(0.122)**	0(0)**	0.459(0.426)
	0.5	0(0)	**0.300**(0.255)**	0(0)	0(0)	0(0)**	0.015(0.069)
ZDT3	0.1	0.675(0.054)**	**0.936**(0.009)*	0.647(0.062)**	0.629(0.060)**	0.007(0.017)**	0.929(0.007)
	0.5	0.041(0.025)**	**0.672**(0.045)**	0.042(0.030)**	0.035(0.038)**	0(0)**	0.569(0.094)
ZDT4	0.1	0.686(0.134)**	**0.931**(0.012)	0.670(0.152)**	0.658(0.199)**	0(0)**	0.907(0.061)
	0.5	0.245(0.188)**	0.645(0.146)**	0.226(0.158)**	0.230(0.203)**	0(0)**	**0.777**(0.091)
ZDT6	0.1	0.311(0.058)**	**0.740**(0.027)	0.308(0.070)**	0.349(0.046)**	0(0)**	0.736(0.025)
	0.5	0.001(0.005)**	**0.406**(0.048)**	0.002(0.006)**	0(0)**	0(0)**	0.300(0.068)
DTLZ1	0.1	0(0)**	0.444(0.358)**	0(0)**	0.364(0.156)**	0(0)**	**0.873**(0.019)
	0.5	0.011(0.050)**	0.102(0.174)**	0(0)**	0.048(0.114)**	0(0)**	**0.492**(0.120)
DTLZ2	0.1	0.026(0.049)**	**0.780**(0.012)	0.007(0.021)**	0.019(0.039)**	0(0)**	0.773(0.009)
	0.5	0.000(0.001)	**0.346**(0.074)**	0.003(0.011)	0.010(0.025)	0(0)**	0.014(0.031)
DTLZ3	0.1	0(0)**	0.027(0.110)**	0(0)**	0.006(0.027)**	0(0)**	**0.714**(0.056)
	0.5	0(0)**	0.004(0.020)**	0(0)**	0(0)**	0(0)**	**0.008**(0.038)
DTLZ4	0.1	0.370(0.097)**	**0.825**(0.135)	0.370(0.117)**	0.297(0.101)**	0(0)**	0.795(0.161)
	0.5	0.087(0.060)**	**0.496**(0.162)	0.092(0.075)**	0.014(0.030)**	0(0)**	0.416(0.134)
DTLZ7	0.1	0.490(0.068)**	**0.830**(0.019)	0.500(0.050)**	0.552(0.037)**	0(0)**	0.820(0.016)
	0.5	0.129(0.033)**	**0.636**(0.034)	0.123(0.056)**	0.037(0.045)**	0(0)**	0.612(0.044)

13. T. Park and K. Ryu. Accumulative sampling for noisy evolutionary multi-objective optimization. In *Proc. of ACM Genetic and Evol. Computat. Conference*, 2011.
14. J. Teich. Pareto-front exploration with uncertain objectives. In *Proc. of Int'l Conf. on Evol. Multi-Criterion Optimization*, 2001.
15. D. A. V. Veldhuizen and G. B. Lamont. Multiobjective evolutionary algorithm test suites. In *Proc. ACM Symposium on Applied Computing*, 1999.
16. T. Voß, H. Trautmann, and C. Igel. New uncertainty handling strategies in multi-objective evolutionary optimization. In *Proc. Int'l Conf. on Parallel Problem Solving from Nature*, 2010.
17. M. Wormington, C. Panaccione, K. M. Matney, and D. K. Bowen. Characterization of structures from x-ray scattering data using genetic algorithms. *Phil. Trans. R. Soc. Lond. A*, 357(1761), 1999.
18. B. Zhu, J. Suzuki, and P. Boonma. Solving the probabilistic traveling salesperson problem with profits (pTSPP) with a noise-aware evolutionary multiobjective optimization algorithm. In *Proc. IEEE Congress on Evol. Computat.*, 2011.
19. E. Zitzler, K. Deb, and L. Thiele. Comparison of multiobjective evolutionary algorithms: Empirical results. *Evol. Computat.*, 8(2), 2000.

Table 4. HVR Results under Chi-squared Noise

	β	NSGA-II	NSGA-II-Median	NSGA-II-Mean	NSGA-II-N	NSGA-II-IQR	NSGA-II-U
ZDT1	0.1	0.689(0.045)**	0.893(0.009)**	0.898(0.012)**	0.897(0.009)**	0.001(0.002)**	**0.922**(0.010)
	0.5	0.173(0.059)**	0.532(0.066)**	0.587(0.053)**	0.615(0.061)**	0(0)**	**0.713**(0.037)
ZDT2	0.1	0.46(0.169)	**0.727**(0.256)	0.733(0.258)	0.279(0.374)	0(0)**	0.603(0.370)
	0.5	0(0)	**0.208**(0.085)**	0.120(0.149)	0(0)	0(0)	0.027(0.087)
ZDT3	0.1	0.675(0.059)**	0.915(0.010)**	0.920(0.008)**	0.914(0.009)**	0.002(0.003)**	**0.930**(0.005)
	0.5	0.071(0.091)	0.490(0.081)	0.521(0.085)	0.550(0.081)	0(0)**	**0.556**(0.066)
ZDT4	0.1	0.610(0.204)**	0.891(0.086)	0.884(0.081)	0.866(0.056)**	0(0)**	**0.926**(0.027)
	0.5	0.218(0.150)**	0.685(0.115)	0.569(0.205)	0.623(0.154)	0(0)**	**0.732**(0.156)
ZDT6	0.1	0.333(0.070)**	0.687(0.032)**	0.684(0.033)**	0.657(0.047)**	0(0)**	**0.740**(0.022)
	0.5	0(0)**	0.239(0.060)**	0.228(0.098)**	0.205(0.063)**	0(0)**	**0.337**(0.066)
DTLZ1	0.1	0(0)**	0.541(0.292)**	0.452(0.390)**	0.265(0.230)**	0(0)**	**0.878**(0.013)
	0.5	0(0)**	0(0)**	0.014(0.047)**	0.009(0.019)**	0(0)**	**0.513**(0.134)
DTLZ2	0.1	0.451(0.058)**	0.745(0.015)**	0.749(0.011)**	0.736(0.013)**	0(0)**	**0.787**(0.004)
	0.5	0.007(0.024)	0.002(0.004)	**0.012**(0.020)	0.026(0.024)	0(0)**	0.007(0.013)
DTLZ3	0.1	0(0)**	0(0)**	0(0)**	0(0)**	0(0)**	**0.742**(0.063)
	0.5	0(0)**	0(0)**	0(0)**	0(0)**	0(0)**	**0.011**(0.041)
DTLZ4	0.1	0.578(0.094)**	0.789(0.129)	0.840(0.012)**	0.827(0.014)**	0(0)**	**0.873**(0.011)
	0.5	0.120(0.093)**	0.285(0.099)**	0.323(0.091)**	0.326(0.079)**	0(0)**	**0.491**(0.100)
DTLZ7	0.1	0.658(0.036)**	0.798(0.013)**	0.787(0.018)**	0.782(0.026)**	0(0)**	**0.828**(0.008)
	0.5	0.138(0.042)**	0.535(0.051)**	0.553(0.053)*	0.558(0.045)*	0(0)**	**0.607**(0.037)

Table 5. HVR Results under Log-normal Noise

	β	NSGA-II	NSGA-II-Median	NSGA-II-Mean	NSGA-II-N	NSGA-II-IQR	NSGA-II-U
ZDT1	0.1	0.822(0.019)**	0.937(0.007)**	0.917(0.006)**	0.917(0.009)**	0.048(0.04)**	**0.951**(0.005)
	0.5	0.323(0.103)**	0.791(0.036)**	0.667(0.052)**	0.683(0.057)**	0.0(0.0)**	**0.857**(0.013)
ZDT2	0.1	0.689(0.164)	**0.882**(0.011)	0.776(0.195)	0.428(0.394)**	0.0(0.0)**	0.753(0.321)
	0.5	0.046(0.074)**	**0.561**(0.196)**	0.278(0.206)	0.043(0.134)**	0.0(0.0)**	0.311(0.353)
ZDT3	0.1	0.873(0.019)**	0.949(0.008)	0.934(0.007)**	0.931(0.008)**	0.086(0.056)**	**0.954**(0.016)
	0.5	0.27(0.118)**	0.806(0.032)**	0.626(0.068)**	0.63(0.094)**	0.0(0.0)**	**0.874**(0.014)
ZDT4	0.1	0.739(0.145)**	0.905(0.069)*	0.898(0.067)*	0.86(0.096)**	0.0(0.0)**	**0.940**(0.016)
	0.5	0.178(0.15)**	0.808(0.089)	0.673(0.199)*	0.596(0.194)**	0.0(0.0)**	**0.816**(0.130)
ZDT6	0.1	0.531(0.052)**	0.784(0.029)**	0.709(0.024)**	0.694(0.026)**	0.0(0.0)**	**0.808**(0.020)
	0.5	0.062(0.034)**	0.471(0.059)**	0.301(0.075)**	0.315(0.062)**	0.0(0.0)**	**0.589**(0.032)
DTLZ1	0.1	0.036(0.158)**	0.561(0.334)**	0.426(0.373)**	0.556(0.265)**	0.0(0.0)**	**0.906**(0.013)
	0.5	0.0(0.0)**	0.146(0.223)**	0.063(0.129)**	0.429(0.167)**	0.0(0.0)**	**0.795**(0.027)
DTLZ2	0.1	0.675(0.016)**	0.779(0.014)**	0.781(0.007)**	0.747(0.013)**	0.0(0.0)**	**0.792**(0.011)
	0.5	0.004(0.012)**	0.589(0.045)**	0.27(0.101)**	0.313(0.112)**	0.0(0.0)**	**0.676**(0.024)
DTLZ3	0.1	0.0(0.0)**	0.0(0.0)**	0.0(0.0)**	0.0(0.0)**	0.0(0.0)**	**0.763**(0.035)
	0.5	0.0(0.0)**	0.008(0.036)**	0.0(0.0)**	0.0(0.0)**	0.0(0.0)**	**0.547**(0.107)
DTLZ4	0.1	0.657(0.178)**	**0.844**(0.094)	0.802(0.161)	0.71(0.192)	0.0(0.0)**	0.816(0.156)
	0.5	0.255(0.094)**	0.653(0.138)	0.516(0.089)**	0.509(0.109)**	0.0(0.0)**	**0.703**(0.206)
DTLZ7	0.1	0.763(0.014)**	**0.840**(0.012)	0.83(0.015)	0.799(0.024)**	0.001(0.003)**	**0.840**(0.030)
	0.5	0.341(0.065)**	0.690(0.030)**	0.631(0.033)**	0.62(0.034)**	0.0(0.0)**	**0.742**(0.028)