Six-Sigma Robust Design Optimization Using a Many-Objective Decomposition-Based Evolutionary Algorithm

M. Asafuddoula, Hemant K. Singh, and Tapabrata Ray

***Abstract*—Robust design optimization aims to find solutions that are competent and reliable under given uncertainties. While such uncertainties can emerge from a number of sources (imprecise variable values, errors in performance estimates, varying environmental conditions, etc.), this paper focuses on problems where uncertainties emanate from design variables. In commercial designs, being reliable is often of more practical value than being globally best (but unreliable). Robust optimization poses three key challenges: 1) appropriate formulation of the problem; 2) accurate estimation of the "robustness" measure; and 3) efficient means to identify the set of tradeoff robust solutions with an affordable computational cost. In this paper, four different problem formulations for robust optimization are presented and analyzed. The proposed formulations offer a set of tradeoff solutions with robustness from two perspectives—feasibility robustness, i.e., robustness against failure and performance robustness, i.e., robustness assuring good performance. The approach also provides means to identify critical constraints or performance functions that affect the overall robustness. The problem is posed as a many-objective optimization problem and a decomposition-based evolutionary approach is used for solving it. The performance of the proposed approach and the consequences of using different formulations are illustrated using two numerical examples and four engineering problems.**

***Index Terms*—Evolutionary algorithm, many-objective, robust optimization.**

I. Introduction

THE notion of uncertainty is omnipresent in any real-world problem. In the context of design optimization, such uncertainties emerge from varying loading conditions, material imperfections, inaccuracies in analyses/simulations, imprecise geometries, manufacturing precision, or even actual product usage. For practical implementation, designs need to be robust, i.e., less sensitive in the presence of uncertainties. While there could be different sources of uncertainties in a problem (environmental factors, design variables, performance estimates etc.), this paper focuses on solving problems with uncertainties in design variables. This class of problems is frequently encountered in various engineering applications as reported in the past [1]–[10].

It is evident that optimization based solely on the maximization of performance (i.e., ignoring the landscape of the objective/constraint functions in the vicinity of a solution) is incapable of identifying robust solutions. First, global optimal solution(s) of a performance maximization problem may lie on a constraint boundary. With marginal deviation in the variable values, such solutions could easily violate one or more constraints leading to design failure. Second, such global optimal solutions may lie on a very narrow peak of the performance function, wherein a slight variation in the value of the variables could result in significant deterioration in design performance.

The first aspect relates to feasibility robustness and is commonly dealt using additional constraints [11]–[14], wherein a set of solutions satisfying a prescribed level of feasibility robustness is identified. The second aspect relates to performance robustness, i.e., robustness in terms of performance deterioration. It is either modeled using additional constraints [15] or incorporated as an objective via variance measures [11], [12], [14], [16]. There is also a differentiation in the notion of performance used in the robust formulations. Some studies use the performance function at the given (mean) variable value [15], [16] while others use the expected value of the performance function [11]–[15]. The latter form of performance measure seems to be more practical and widely adopted.

In this paper, robustness with respect to constraints (feasibility robustness) and objectives (performance robustness) are studied using four different formulations. The key differences between the formulations are discussed using two test problems. Since the proposed formulations use additional objectives to deal with feasibility robustness and performance robustness, a many-objective optimization algorithm is used to solve the problem. Unlike previous approaches which can identify a set of tradeoff solutions satisfying a prescribed feasibility robustness criteria, the proposed approach offers the complete set of tradeoff solutions, i.e., solutions spanning various levels of feasibility robustness and performance robustness simultaneously in a single run. Such a set of tradeoff solutions is of practical value to the decision maker as one can clearly observe the performance/cost implications of delivering solutions at various robustness levels.

The rest of this paper is organized as follows. A background of previous work in the area of robust optimization is presented

Manuscript received November 12, 2013; revised February 18, 2014 and May 5, 2014; accepted June 30, 2014. Date of publication July 29, 2014; date of current version July 28, 2015. The work of M. Asafuddoula was supported by the Publication Fellowship, UNSW Canberra. The work of H. K. Singh was supported by the Early Career Research Grant, UNSW Canberra. The work of T. Ray was supported by the Future Fellowship, Australian Research Council.

The authors are with the School Engineering and Information Technology, University of New South Wales, Canberra, ACT 2612, Australia (e-mail: Md.Asaf@adfa.edu.au; h.singh@adfa.edu.au; t.ray@adfa.edu.au).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TEVC.2014.2343791

in Section II. The problem definition and robustness measures are presented in Section III, while the proposed optimization algorithm (DBEA-r) is detailed in Section IV. The working of the proposed algorithm and the consequences of using various formulations are illustrated using two simple numerical examples in Section V. The performance of the algorithm is further studied using two single objective, one multiobjective and one many objective engineering problem in Section VI. Summary of the presented work and some future directions are discussed in Section VII while the conclusion is presented in Section VIII.

II. Background

Identification of robust solutions has always been a problem of practical interest. In the past, conservative designs were generated by adding a factor of safety to constraints/variables [17]. In recent years, more involved research has focused on developing approaches to quantify and identify robust optimal solutions. The studies can be broadly classified into three areas that deal with: 1) formulation of a robust optimization problem; 2) quantification of robustness; and 3) means to deal with such problems with affordable computing resources, i.e., the search algorithms. Throughout this discussion, minimization of the objective function(s) is assumed for consistency.

A. Robust Optimization Problem Formulation

In the context of problem formulation, a number of different approaches have been proposed in the literature to deal with the aspects of feasibility robustness and performance robustness. Features of some commonly used robust formulations reported in literature are summarized in Table I.

Constraint modification is one of the simplest forms to deal with feasibility robustness [11], [12], [18]. A constraint $g < 0$, is translated to $\mu_g + n\sigma_g < 0$, where n defines the level of robustness in six-sigma terminology (discussed later in Section III). The inclusion of the term $n\sigma$, essentially translates the constraint boundary inward (i.e., reduces the feasible space), thereby ensuring that the solutions do not violate the (mean) constraint boundaries of the problem. Apart from modifying the original constraints, some other studies have suggested use of additional constraints, e.g., probability of violation of the constraints [19], [20] or reliability index [14], [21]–[23]. Sequential reliability assessment combined with deterministic optimization has also been reported [24]. It is important to highlight that all the aforementioned techniques deliver solutions for a prescribed level of feasibility robustness.

In the context of this paper, incorporating performance robustness implies that variance in the objective function(s) are considered in some form along with the objective/expected objective values. To deal with performance robustness, the approaches either use aggregation or include additional objectives or constraints in the formulation. For example, in [18], [25], and [26], a simple aggregate function i.e., $\mu_f+\sigma_f$ (or $\mu_f + k\sigma_f$) was used. A weighted composite function was also used in [12] and [27] of the form $\lambda\mu_f^2 + (1-\lambda)\sigma_f^2$ (or $\lambda\mu_f + (1-\lambda)\sigma_f$), where the factors could be varied to emphasize/deemphasize the effects of the mean and the standard deviation terms. Another different formulation was presented in [28] and [29] of the form $\alpha(\mu_f/\mu_{f^*}) + (1-\alpha)(\sigma_f/\sigma_{f^*})$. Alternate propositions of including robustness as a separate objective also appear in [11], [13], [16], [25], and [30]–[35] wherein a robustness measure/index was used as an additional objective during the course of search. Often to ensure performance robustness, expected performance measures are used in lieu of the original performance function [13], [15]. Another method of enforcing performance robustness appears in the works of [15], [36], and [37] wherein a constraint of the form $((||f^p(x) - f(x)||) / ||f(x)||) < \eta$ was added to the original problem, where f^P denotes the effective value i.e., the worst function value among chosen neighborhood solutions and η is a user defined parameter.

It can be observed from Table I that while some of the works in the past have considered performance robustness as objective, none of the works have included both feasibility and performance robustness as objectives so as to achieve solutions with varying levels of feasibility and performance robustness in a single run.

B. Quantification of Robustness

Most of the approaches for robustness quantification reported in the literature utilize the expected value and/or variance of the performance function [26]. When the analytical function is known, the expected function value can be calculated as the integral $\int f(x)w(x)dx$, and certain analytical techniques [13], [27], [40]–[46] could be used to quantify the robustness. However, for most optimization problems, such expressions may not be available, and stochastic sampling–based approaches have to be used instead for the estimates [6], [7], [47]–[52]. The major downside lies with the mechanism for estimating the expected fitness as it requires a large number of samples to compute the expected value with good accuracy. For a finite number of samples, explicit averaging [6], [7], [53]–[56] or implicit averaging [57]–[59] may be used for estimates. To save on function evaluations, some studies used metamodels [32], [60] for calculating the expected values instead of the original function.

In the context of reliability against failure (feasibility robustness), probabilistic models have been proposed, wherein the constraints of the problem are transformed to chance constraints such as

$$P\left(g_j(\mathbf{d}, \mathbf{x}) \geq 0\right) \geq R_j, j = 1, 2, \ldots\ldots, J \tag{1}$$

where R_j is the desired probability of constraint satisfaction of the jth constraint. Often such probabilistic constraints are substituted with deterministic constraints such as

$$1 - \frac{r_j}{N} \geq R_j, j = 1, 2, \ldots\ldots, J \tag{2}$$

where N is the sample size and r_j is the number of failures among N samples. Although the method is simple, the appropriate sample size required to estimate the quantity r_j often becomes computationally prohibitive. Some studies have focused on reducing the sample size via Latin hypercube sampling [61], [62], importance sampling [63], directional sampling [64], Taylor series expansion [65], [66], and polynomial chaos [49], [67]. While the above discussed

TABLE I
COMMON FORMULATIONS TO IDENTIFY ROBUST SOLUTIONS*

Reference Work	Robust formulation	Type of robustness	Robustness as an additional objective	Quantification of Robustness	Ability to generate robust solutions with varying levels of robustness in a single run	Optimization Method	Type of problems studied
K. Deb, H. Gupta [36]	Min $[f_1, f_2 ..., f_M]$ Subject to $\frac{\|\|f^p(x)-f(x)\|\|}{\|\|f(x)\|\|} \leq \eta$ Or Min $[f_1^{eff}, f_2^{eff} ..., f_M^{eff}]$	Performance robustness (as additional constraint)	No	Expected measure	No	Real Parameter GA	(MO, C)
Y. Jin and B. Sendhoff [16]	Min $[f_1 = f, f_2 = \sigma_f]$ Subject to $g_j \leq 0$	Performance robustness (as additional objective)	Yes	Variance measure	Yes, w.r.t. performance	Evolutionary Multi-objective approach	(SO, C)
W. Chen et. al. [11]	Min $[(\mu_f, \sigma_f^2)]$ Subject to $E[g_j(x,z)] + n\sigma_{g_j} \leq 0$	Feasibility and performance robustness	Yes	Expected and variance measures	Yes, w.r.t performance	Compromise DSP	(MO, C)
G. Sun et. al. [12]	Min $[(\mu_{f_1}, \sigma_{f_1}),$ $(\mu_{f_2}, \sigma_{f_2}), ...,$ $(\mu_{f_M}, \sigma_{f_M})]$ Subject to $\mu_{g_j} + n\sigma_{g_j} \leq 0$	Feasibility and performance robustness	Yes	Sigma level based measure	Yes, w.r.t performance	PSO	(MO, C)
S. Sundaresan et. al. [13]	Min $E[f]$ Subject to $E[g_j] \leq 0, E[h_j] = 0$	Feasibility robustness	No	Expected measure	No	Mathematical programming	(SO, C)
Z. Wang et. al. [18]	Min $[\mu_f + k\sigma_f]$ Subject to $\mu_{g_j} + n\sigma_{g_j} \leq 0$	Feasibility and performance robustness	No	Aggregation of expected and variance measure	No	Mathematical programming	(SO, C)
Z. P. Mourelatos and J. Liang [38]	Min $[(\mu_f, \Delta R_\sigma = \sigma_{R2} - \sigma_{R1})]$ Subject to $Prob\{g_j \leq 0\} \geq \alpha_i, i = 1, 2, ..., M$	Feasibility and performance robustness	Yes	Expected and variance measures	Yes, w.r.t performance	Mathematical programming	(MO, C)
B. D. Youn et. al. [39]	Min $[(\frac{\mu_H - h_t}{\mu_{H_0} - h_t})^2 + (\frac{\sigma_H}{\sigma_{H_0}})^2]$ or $[(sgn(\mu_H))(\frac{\mu_H}{\mu_{H_0}})^2 + (\frac{\sigma_H}{\sigma_{H_0}})^2]$ or $[(sgn(\mu_H))(\frac{\mu_1/H}{\mu_1/H_0})^2 + (\frac{\sigma_1/H}{\sigma_2/H_0})^2]$ Subject to $Prob\{g_j \leq 0\} \geq \alpha_i, i = 1, 2, ..., M$	Feasibility and performance robustness	No	Expected and variance measures	No	Mathematical programming	(MO, C)

*SO: single objective; MO: multi-objective; C: constrained. For complete details on the robust formulations, please refer to the cited publications.

methods rely on some form of sampling, there are also reports of sensitivity analysis based on the gradient information [30], [38]. Other works include computation of sensitivity via non-gradient forms, wherein, additional constraints are imposed [30] leading to the notion of acceptable performance variation region (AVPR). For such formulations, it is also necessary to compute the sensitive region (SR) before a solution violates AVPR. Since the SR could be asymmetric, i.e., a solution could be more sensitive in one direction of variation in a variable Δx, but less sensitive in others, a worst case sensitive region (WCSR) model is often required. The process of identifying WCSR is itself a complex optimization problem. While the above methods discussed so far have their origins rooted in the field of robust optimization, there are methods in the domain of reliability optimization, where the reliability of a solution is computed by determining its distance from the closest constraint boundary. The approach is commonly known as "most probable point" (MPP) of failure [68]. MPP is known to be computationally expensive as it requires several loops of optimization. A number of variants have been proposed in recent years to reduce the computational expense involved, viz., Performance measure approach (PMA), fast performance measure approach (FastPMA), reliability index approach (RIA), fast reliability index approach (FastRIA) [19]. MPP-based methods generally include the first order reliability method (FORM) and second order reliability method (SORM) [35], [68], [69]. The sensitivity analysis is achieved by simplifying the limit state function with the first order or second order Taylor expansion at the MPP or SORM. The SORM method is more accurate than FORM which requires a second order Taylor series approximation around the MPP of the limit state function.

C. Search Algorithms

While issues concerning the formulation of the problem and the measures of robustness have been discussed above, the outcome of a robust optimization exercise also depends on the efficiency of the underlying search strategy. In most of the works discussed in the previous sub-section, population-based stochastic techniques (e.g., [70]) have been used for optimization. When additional objectives are introduced in the formulation, the number of objectives could often be more than four. It is well reported in literature that such problems (with four or more objectives) cannot be efficiently solved using nondominance-based multiobjective optimization algorithms [71]. Hence, popular multiobjective approaches such as NSGA-II [70] and SPEA2 [72] cannot be efficiently used if the formulation results in a many-objective problem. Modifying the selection pressure through the use of secondary metrics (e.g., substitute distance measures [73], [74], average rank domination [75], fuzzy dominance [76], [77], ϵ-dominance [78], [79], adaptive ϵ-ranking [80] etc.) has so far exhibited only partial success in solving such problems. In all the above approaches relying on secondary measures, there is no guarantee that the final nondominated set of solutions would span the entire Pareto surface uniformly. Methods relying on reference directions have been most promising so far for many-objective problems [81], [82] in terms of achieving convergence and diversity.

The aim of this paper is to seek enhancements in all three aspects presented above. To this effect, the following studies are presented.

1) *Formulation:* As shown in Table I, some of the formulations used in the past to solve robust optimization problems deliver solutions for a prescribed level of robustness. Some others provide solutions with varying levels of robustness in a single run, but with respect to performance only. In this paper, we intend to formulate the problem so as to deliver solutions with varying levels of robustness in a single run, both with respect to feasibility and performance. Four different formulations have been presented in order to cover some of the past formulations as well as the new one which delivers the aforementioned tradeoff set.
2) *Quantification of Robustness:* While the concept of 6σ is well recognized as quantification of robustness for feasibility robustness, the same has been introduced for the objectives in order to quantify performance robustness. The user can prescribe the acceptable deviation from the average performance specification of a design, which is then used to calculate the sigma level in terms of performance robustness. This way of quantifying robustness helps comparing the robustness of solution with respect to each objective/constraint on a common scale (even though their raw values and standard deviations values may be of different orders). For the case of multiple constraints and/or multiple objectives, minimum sigma level for a solution is considered as its overall sigma level, which helps in assuring this minimum level of feasibility/performance robustness on the solutions with respect to all objectives/constraints. The robustness measures are treated as additional criteria in the problem formulation.
3) *Search Algorithm:* The addition of robustness criteria in order to formulate the robust optimization problem results in an increase in number of objectives, and often the resulting problem could have four or more objectives. In order to solve them efficiently, the use of many-objective algorithms is introduced. A steady state, decomposition-based evolutionary algorithm for robust optimization (DBEA-r) is proposed to achieve this. The reference directions for the algorithm are generated using systematic sampling, and association of the solutions to these reference directions are based on two independent distance measures, one for convergence and the other for alignment/diversity. Constraints of the formulation are handled using an adaptive epsilon level–based scheme [83], [84].

III. Problem Definition and Robustness Measure

A generic multiobjective optimization problem can be defined as

$$\begin{aligned}
&\underset{(\mathbf{d},\mathbf{x})}{\text{Minimize}}\; f_i(\mathbf{d},\mathbf{x}),\; i=1,2,......M.\\
&\text{Subject to}\\
&\qquad g_j(\mathbf{d},\mathbf{x}) \ge 0,\; j=1,2,........................p\\
&\qquad h_j(\mathbf{d}) = 0,\; j=p+1,p+2,........p+q\\
&\qquad \mathbf{x}^{(L)} \le \mathbf{x} \le \mathbf{x}^{(U)}\\
&\qquad \mathbf{d}^{(L)} \le \mathbf{d} \le \mathbf{d}^{(U)}
\end{aligned} \tag{3}$$

where $f_1(\mathbf{d},\mathbf{x}), f_2(\mathbf{d},\mathbf{x}), f_3(\mathbf{d},\mathbf{x}), \ldots\ldots f_M(\mathbf{d},\mathbf{x})$ are the M objective functions. The functions involve a set of deterministic variables $\mathbf{d}$ and a set of uncertain variables $\mathbf{x}$. The number of inequality and equality constraints are denoted by p and q respectively. In practice, most engineering design optimization problems involve one or more variables of uncertain nature, which are often represented using their probability distributions. In this paper, the distribution of uncertain variables is assumed to be Gaussian, which reflects the behavior of majority of the variables in the context of engineering design. The variables are represented using their mean and standard deviation as $N(\mu_x, \sigma_x)$.

To convert an optimization problem to a robust optimization problem, one needs to adopt a robustness measure. In this paper, we adopt the notion of six-sigma quality measure commonly used in the industry. This measure is discussed next, followed by the problem formulations.

A. Six-Sigma Quality Measures

The notion of six-sigma refers to having six times the standard deviation (of the design performance) between the process mean/expected value and the nearest specification

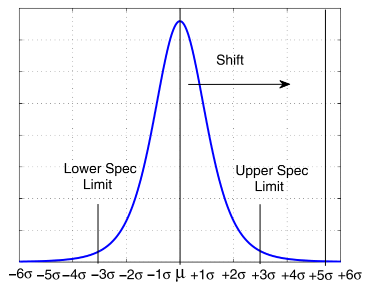


Fig. 1. Normal distribution, 3-σ design.

TABLE II
CONFIDENCE INTERVALS AND CORRESPONDING DEFECTS PER MILLION FOR VARIOUS SIGMA LEVELS

Sigma Level	Confidence Interval (CI)	Defects per million
$\pm 1\sigma$	68.26	691462
$\pm 2\sigma$	95.46	308538
$\pm 3\sigma$	99.73	66807
$\pm 4\sigma$	99.9937	6210
$\pm 5\sigma$	99.999943	233
$\pm 6\sigma$	99.9999998	2

limit as shown in Fig. 1. It is assumed that when the ratio of expected value and the standard deviation is six, practically no items (2 defects per million) will fail to meet the specifications [25]. For feasibility robustness, the specification limits are the bounds of the constraints. For performance robustness, a user defined acceptable deviation in performance from its mean value is used as the limit in this paper.

Table II shows the sigma level and the corresponding confidence interval that the performance of the solution will lie between the process mean and the specification limit.

B. Problem Formulation and Robustness Quantification

In this paper, four different formulations for robust optimization are presented, covering aspects of both feasibility and performance robustness. These cover two different cases, one using performance and other using expected performance quantification measures. The formulations (Form-1 to Form-4) are listed in Table III. Form-1 and Form-3 consider feasibility robustness only, while Form-2 and Form-4 consider both feasibility and performance robustness. Performance value is directly used as an objective in Form-1 and Form-2 which, given the Gaussian variation in design variables, is equivalent of using the performance at the mean design point ($f(\mu_{\mathbf{x}})$). Expected values (μ_f) of the performance functions are used as objectives in Form-3 and Form-4, which have been estimated by averaging over samples in the vicinity of design point with given uncertainties. Two robustness measures, namely sigma$_g$ and sigma$_f$ are introduced to quantify feasibility and performance robustness, respectively.

1) The term sigma$_g$ refers to the ratio of expected constraint value (μ_g) and standard deviation (σ_g) of constraint g. Since the constraints have been formulated as $g > 0$, the ratio sigma$_g = \mu_g/\sigma_g$ is nothing but $(\mu_g - 0)/\sigma_g$, which is a measure of how many standard deviations can be fit between the constraint boundary (0) and the given solution (see Fig. 1). This quantity is positive for feasible solutions, and needs to be maximized for achieving high robustness.
2) The term sigma$_f$ refers to the ratio of a user defined acceptable deviation σ_{f0} and the standard deviation (σ_f) of objective f. For σ_f, unlike σ_g, the boundary is not zero but the (user prescribed) specification limit on how much deviation in the objective value is acceptable from the mean. Hence, the ratio sigma$_f = \sigma_{f0}/\sigma_f$ denotes the number of standard deviations of the objective function that can be fit within the specification limit. Again, sigma$_f$ is intended to be maximized.

To ensure feasibility robustness, the quantity sigma$_g$ has to be positive and solutions with larger sigma$_g$ are more robust. If the value of sigma$_g$ is greater than a given value R_c, the value is truncated to R_c. It essentially means, the user is satisfied with the feasibility robustness level R_c and solutions having any higher robustness has the same preference as the one with R_c. A similar truncation strategy has been applied to sigma$_f$ (using R_f) to ensure performance robustness in Form-2 and Form-4. In a problem involving multiple constraints, the minimum sigma$_g$ across all constraints is considered to represent the overall sigma$_g$ of the solution. This translates to measuring the sigma-level of constraint that is most likely to be violated; which is different from traditionally used six-sigma formulation where defects caused using all constraints together are considered. Same strategy has been adopted for sigma$_f$ for the case of multiple objectives. This way of quantifying robustness helps comparing the robustness of solution with respect to each objective/constraint on a common scale (even though their raw values and standard deviations may be of different orders), and at the same time assuring this minimum level of feasibility/performance robustness on the solutions with respect to each objective/constraint. For a six-sigma design considered in this paper, both R_c and R_f are set to 6. Form-1 and Form-2 formulations are similar to those presented in [19], where objective value at mean point ($f(\mu_x)$) is considered as objective rather than the mean of the function ($\mu_{f(x)}$). These formulations have been included for completeness. However, in realistic situations, with given uncertainty in the variables, expected/average objective values for a design are more reflective of field performance compared to objective values at most likely point. It is clear from Table III that Form-4 is the most comprehensive formulation that is capable of delivering the set of tradeoff solutions spanning the entire range of feasibility robustness and performance robustness.

IV. DECOMPOSITION-BASED EVOLUTIONARY ALGORITHM FOR ROBUST OPTIMIZATION

Robust formulations presented above require solution of optimization problems involving additional objectives. The total number of such objectives can easily be more than four and hence used of a many objective optimization algorithm is proposed in this paper. The underlying algorithm is a decomposition-based evolutionary algorithm (DBEA)

TABLE III
DIFFERENT FORMS OF ROBUST FORMULATION*

Robust form	Formulation Type	Robust formulation		Robust measure	Ability to generate tradeoff feasible robust solutions in a single run
Form-1	Original objective function(s) with feasibility robustness	$\underset{(\mathbf{d},\mathbf{x})}{\text{Minimize}} f_i(\mathbf{d},\mathbf{x}), i=1,2,......M$ $\underset{(\mathbf{d},\mathbf{x})}{\text{maximize}} f_{M+1}(\mathbf{d},\mathbf{x}) = \text{Min}(sigma_g, R_c)$ subject to $sigma_g \equiv \text{Min}(\mu_{g_j(\mathbf{d},\mathbf{x})}/\sigma_{g_j(\mathbf{d},\mathbf{x})}) \geq 0$ $\mathbf{x}^{(L)} \leq \mathbf{x} \leq \mathbf{x}^{(U)}, \mathbf{d}^{(L)} \leq \mathbf{d} \leq \mathbf{d}^{(U)}$	(4)	Sigma level based measure $(sigma_g)$	Yes
Form-2	Original objective function(s) with feasibility and performance robustness	$\underset{(\mathbf{d},\mathbf{x})}{\text{Minimize}} f_i(\mathbf{d},\mathbf{x}), i=1,2,......M$ $\underset{(\mathbf{d},\mathbf{x})}{\text{maximize}} f_{M+1}(\mathbf{d},\mathbf{x}) = \text{Min}(sigma_g, R_c)$ $\underset{(\mathbf{d},\mathbf{x})}{\text{maximize}} f_{M+2}(\mathbf{d},\mathbf{x}) = \text{Min}(sigma_f, R_f)$ subject to $sigma_g \equiv \text{Min}(\mu_{g_j(\mathbf{d},\mathbf{x})}/\sigma_{g_j(\mathbf{d},\mathbf{x})}) \geq 0$ where $sigma_f \equiv \text{Min}(\sigma_{f_{0,i}(\mathbf{d},\mathbf{x})}/\sigma_{f_i(\mathbf{d},\mathbf{x})})$ $\mathbf{x}^{(L)} \leq \mathbf{x} \leq \mathbf{x}^{(U)}, \mathbf{d}^{(L)} \leq \mathbf{d} \leq \mathbf{d}^{(U)}$	(5)	Sigma level based measure $(sigma_g, sigma_f)$	Yes
Form-3	Expected objective function(s) with feasibility robustness	$\underset{(\mathbf{d},\mathbf{x})}{\text{Minimize}}\ \mu_{f_i(\mathbf{d},\mathbf{x})}, i=1,2,......M$ $\underset{(\mathbf{d},\mathbf{x})}{\text{maximize}} f_{M+1}(\mathbf{d},\mathbf{x}) = \text{Min}(sigma_g, R_c)$ subject to $sigma_g \equiv \text{Min}(\mu_{g_j(\mathbf{d},\mathbf{x})}/\sigma_{g_j(\mathbf{d},\mathbf{x})}) \geq 0$ $\mathbf{x}^{(L)} \leq \mathbf{x} \leq \mathbf{x}^{(U)}, \mathbf{d}^{(L)} \leq \mathbf{d} \leq \mathbf{d}^{(U)}$	(6)	Sigma level based measure $(sigma_g)$	Yes
Form-4	Expected objective function(s) with feasibility and performance robustness	$\underset{(\mathbf{d},\mathbf{x})}{\text{Minimize}}\ \mu_{f_i(\mathbf{d},\mathbf{x})}, i=1,2,......M$ $\underset{(\mathbf{d},\mathbf{x})}{\text{maximize}} f_{M+1}(\mathbf{d},\mathbf{x}) = \text{Min}(sigma_g, R_c)$ $\underset{(\mathbf{d},\mathbf{x})}{\text{maximize}} f_{M+2}(\mathbf{d},\mathbf{x}) = \text{Min}(sigma_f, R_f)$ subject to $sigma_g \equiv \text{Min}(\mu_{g_j(\mathbf{d},\mathbf{x})}/\sigma_{g_j(\mathbf{d},\mathbf{x})}) \geq 0$ $\mathbf{x}^{(L)} \leq \mathbf{x} \leq \mathbf{x}^{(U)}, \mathbf{d}^{(L)} \leq \mathbf{d} \leq \mathbf{d}^{(U)}$ where $sigma_f \equiv \text{Min}(\sigma_{f_{0,i}(\mathbf{d},\mathbf{x})}/\sigma_{f_i(\mathbf{d},\mathbf{x})})$ $\mathbf{x}^{(L)} \leq \mathbf{x} \leq \mathbf{x}^{(U)}, \mathbf{d}^{(L)} \leq \mathbf{d} \leq \mathbf{d}^{(U)}$	(7)	Sigma level based measure $(sigma_g, sigma_f)$	Yes

*SO: single objective; MO: multi-objective; MaO: many-objective; C: constrained; the value of R_c and R_f is considered 6 to meet the six sigma quality.

developed by the authors. In this paper, since the algorithm has been discussed in the context of robust optimization, it is referred to as DBEA-r. DBEA-r is outlined in Algorithm 1. The details of the algorithm are discussed in following subsections.

1) *Generation of Reference Points:* A structured set of reference points γ is generated spanning a hyperplane with unit intercepts in each objective axis using normal boundary intersection method (NBI) [85]. The approach generates W points on the hyperplane with a uniform spacing of $\delta = 1/s$ for any number of objectives M with s unique sampling locations along each objective axis. The distribution of the reference points is presented in Fig. 2.

The reference directions are formed by constructing a straight line from the origin to each of these reference points. The population size of the algorithm is set to the number of reference points. The initial population consists of W individuals generated randomly within the variable bounds. Such solutions are thereafter assigned randomly to a unique reference direction during the phase of initialization. For any given reference direction, the performance of a solution can be judged using two measures d_1 and d_2 as shown in 8 and 9. The first measure d_1 is the Euclidean distance between origin and the foot of the normal drawn from the solution to the reference direction, while the second measure d_2 is the length of the normal. Mathematically, d_1 and d_2

Algorithm 1 DBEA-r

Input: $Gen_{\max}$ (maximum number of generations), W (number of reference points), p_c (probability of crossover), p_m (probability of mutation), η_c (crossover index), η_m (mutation index)

```
Generate the reference points using NBI
Initialize the population P; |P| = W and randomly assign
  each individual of P to an unique reference direction.
Evaluate the initial population using prescribed robust formulation (Table III)
  and compute the ideal point z̄_j = (f_1^min, f_2^min, ....., f_M^min),
  identify the corners and compute intercepts a_j's for j = 1 to M
Scale the individuals of the population
Assign the 2M corner solutions to a corner set S.
while (gen ≤ Gen_max) do
  for i = 1:W do
    Select P_i as the base parent
    I = Select its partner randomly from W
    Create a child via recombination as C_i
    Evaluate C_i and compute the distances (d_1 and d_2)
      using all reference directions
    Update the corner solution set S using corner-sort
    Replace the parent P_l with C_i using single-first encounter strategy,
      where l denotes the index of the first parent satisfying the condition of replacement
    Update the ideal point (z̄), the intercepts and re-scale the population
  end for
end while
```

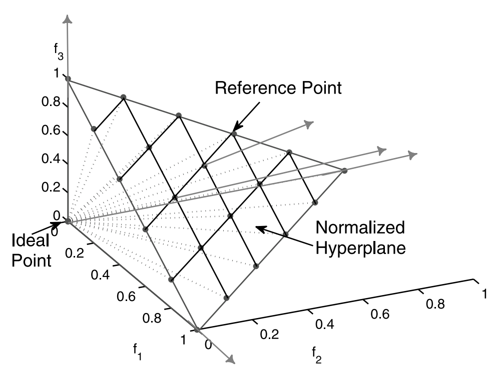


Fig. 2. Set of reference points in a normalized hyperplane for number of objectives, $M = 3$ and $p = 5$.

are computed as

$$d_1 = \mathbf{w}^T \mathbf{f}'(\mathbf{x}) \tag{8}$$

$$d_2 = \|\mathbf{f}'(\mathbf{x}) - \mathbf{w}^T \mathbf{f}'(\mathbf{x})\mathbf{w}\| \tag{9}$$

where $\mathbf{w}$ is a unit vector along any given reference direction. Evidently, a value of $d_2 = 0$ ensures the solutions are perfectly aligned along the required reference direction ensuring good diversity, while a smaller value of d_1 indicates superior convergence. These two measures are subsequently used to control diversity and convergence of the algorithm.

2) *Normalization and Computation of Distances:* Decomposition-based algorithms rely heavily on d_1 and d_2 distances and normalization is necessary in the event the objectives are in different orders of magnitude. In DBEA-Eps [81] and M-NSGA-II [82], the normalization is based on intercepts calculated using M extreme points of the non-dominated set. In DBEA-r, M solutions are identified using a corner-sort ranking [86] procedure. In corner sort, the top M solutions are the minimum in each objective, while the following M solutions are the minimum based on L_2 norm of all but one objectives. From the set of $2M$ solutions, the maximum in each objective is identified and corresponding solutions which have led to the maximum value are selected and to referred as extreme points $\mathbf{z}^e$. Such extreme points are used to create the hyperplane and compute the intercepts. In the event the number of such extreme points are less than M, the maximum value of the objective is used as the intercept value (a_j's). The ideal point of a population is denoted by $z_j = (f_1^{\min}, f_2^{\min}, \ldots\ldots, f_M^{\min})$. The intercepts of the hyperplane along the objective axes are denoted by $a_1, a_2, \ldots\ldots, a_M$. The generic equation of a plane through these points can be represented using

$$C_1 f_1 + C_2 f_2 + \ldots\ldots + C_M f_M = 1 \tag{10}$$

where, C_1, $C_2, \ldots\ldots$, C_M are the unit normal of the plane. The intercepts of the plane with the axis are given by $a_1 = 1/C_1$, $a_2 = 1/C_2, \ldots\ldots$, and $a_M = 1/C_M$.

In the event, the number of such solutions are less than M or any of the a_j's are negative, a_j's are set to $f_j^{\max}$. Every solution in the population is subsequently scaled as

$$f_j'(\mathbf{x}) = \frac{f_j(\mathbf{x}) - z_j}{a_j - z_j}, \quad \forall j = 1, 2, \ldots M. \tag{11}$$

3) *Recombination:* In the recombination process, two child solutions are generated for each pair of parents using simulated binary crossover (SBX) operator and polynomial mutation [87]. The first child is considered as an individual attempting to replace any parent in the population.

4) *Selection/Replacement:* In the steady state form, if a child solution is non-dominated with respect to the individuals in the population, it attempts to enter the population via a replacement. For the entry, the child solution has to compete with all the solutions in the population in a random order until it makes a successful replacement. If we denote the distances as $\{d_{1_r}, d_{2_r}\}$ for a r^{th} solution in the population and $\{d_{1_c}, d_{2_c}\}$ denotes the distances for the child solution along r^{th} reference direction, a child is considered winner if d_{2_c} is less than d_{2_r}. In the event the d_{2_c} is equal to d_{2_r}, the child is considered a winner if d_{1_c} is less than d_{1_r}. The simple precedence of d_2 over d_1 eliminates the need for a complex epsilon-based scheme [83].

5) *Constraint Handling:* The constraint handling approach used in this paper is based on epsilon level comparison and has been reported earlier in [83]. The feasibility ratio (FR) of a population refers to the ratio of the number of feasible solutions in the population to the number of solutions (W). The allowable violation is calculated as

$$CV = \sum_{i=1}^{p} \max(g_i, 0) + \sum_{i=1}^{q} \max(|h_i - \epsilon|, 0) \quad (12)$$

$$CV_{\text{mean}} = \frac{1}{W} \sum_{j=1}^{W} \left(CV_j\right) \quad (13)$$

$$\text{Allowable violation } (\epsilon_{CV}) = CV_{\text{mean}} FR. \quad (14)$$

An epsilon level comparison using this allowable violation measure is used to compare two solutions. If two solutions have their constraint violation value less than this epsilon level, the solutions are compared based on their objective values i.e., via d_1 and d_2 measures. Such a constraint handling scheme has performed better than feasibility first schemes on recent constrained optimization benchmarks [83].

V. Numerical Examples

In this section, we discuss the key differences between Form-1, Form-2, Form-3, and Form-4 robust formulations with respect to the final set of solutions. Two simple numerical test problems are used for illustration, and subsequently the performance of the approach is further assessed using two single-objective, one multiobjective, and a many-objective optimization problem in the next section.

Population sizes of 50, 91, 220, 330, 462, 462 have been used for 2, 3, 4, 5, 6, and 7 objective [including original(f)/expected(μ_f) and robustness (sigma$_g$, sigma$_f$) objectives) optimization problems respectively. The size of the population is set to be the same as the number of reference directions which are identified using normal boundary intersection (NBI) [85]. The probability of crossover is set to 1 and the probability of mutation is set to $p_m = 1/D$, where D is the dimensionality of the problem. The distribution index of crossover and mutation are set to $\eta_c = 30$ and $\eta_m = 20$ as in [82]. The population is evolved over 100 generations. A sample size of 100 is used to compute the expected value and the standard deviation for given solution. Latin-hypercube Sampling with Gaussian distribution (*LHS-Gaussian*) is used to generate the samples around the solution in all studies.

A. Example-1 (Robust Single Objective Optimization)

1) Test Function: Function f is a one variable problem with five unequal peaks in the range $0 \le x \le 1$. It is defined as

$$f = \begin{cases} e^{-2ln2\left(\frac{x-0.1}{0.8}\right)^2} |\sin(5\pi x)|^{0.5}: 0.4 < x \le 0.6 \\ e^{-2ln2\left(\frac{x-0.1}{0.8}\right)^2} \sin^6(5\pi x): \text{otherwise.} \end{cases} \quad (15)$$

The problem is to minimize objective f subject to a constraint $(x - 0.1) \ge 0$. It can be seen from Fig. 3 that the

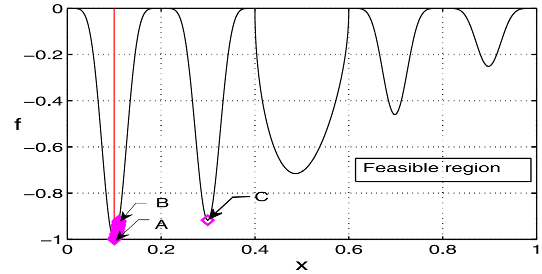


Fig. 3. Solutions obtained using Form-1 robust formulation. The solutions are labeled as (x, f, *Sigma*$_g$, *Sigma*$_f$) i.e., A = (0.1000, −1.0000, 0.0012, 0.3952), B = (0.1104, −0.9220, 0.4720, 0.3622), and C = (0.2994, −0.9172, 6, 0.4309).

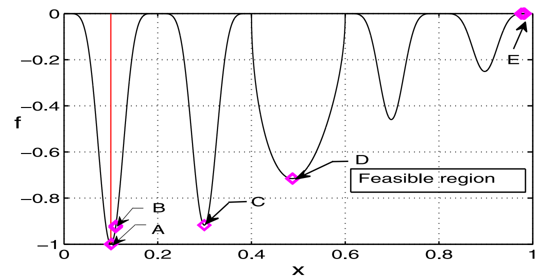


Fig. 4. Solutions obtained using Form-2 robust formulation. The solutions are labeled as (x, f, sigma$_g$, sigma$_f$) i.e., A = (0.1000, −1.000, 0.0012, 0.3952), B = (0.1106, −0.9187, 0.4834, 0.3625), C = (0.2994, −0.9172, 6, 0.4309), D = (0.4885, −0.7152, 6, 3.3308), and E = (0.9832, −5.89e-5, 6, 6).

problem has four sharp local optima and one relatively robust optimum. The global optimum is located at $x = 0.1$ with the function value of −1.0. If the uncertain variable x is assumed to follow a Gaussian distribution with a $\sigma_x^2 = 5 \times 10^{-4}$, the robust minimum is be located at $x = 0.486$ with a function value of −0.715.

Fig. 3 shows the solutions obtained using Form-1 robust formulation. One can clearly observe solutions with varying levels of feasibility robustness, i.e., different values of sigma$_g$. For example, the solutions marked as A, B, and C have sigma$_g$'s of 0.0012, 0.4720, and 6.0, respectively. One can observe that there are no solutions between sigma$_g$ = 0.4720 and 6 since solutions in this range have been dominated by solution C (which has sigma$_g$ = 6, and whose objective value is only inferior to the solutions between A and B). As performance robustness has not been considered in the Form-1 robust formulation, the values of sigma$_f$ lie within a small band i.e., between 0.3622 and 0.4309. The above listed sigma$_f$'s were computed using a user defined objective limit $\sigma_{f_{0,1}} = 0.101$ and 100 neighboring solutions using LHS-Gaussian sampling.

For Form-2 robust formulation, the tradeoff solutions are presented in Fig. 4. Since sigma$_f$ is considered as an additional objective, two new solutions i.e., D and E emerge with sigma$_f$ = 3.3308 and 6 respectively. One can notice the increased range of sigma$_f$ using this formulation when compared to the earlier form. Solutions C, D, and E are feasibility robust solutions with sigma$_g$ = 6.

In the next two formulations i.e., Form-3 and Form-4, the expected value (μ_f) of performance function is used as objective. Once again Form-3 considers feasibility robustness only while Form-4 considers both feasibility and performance

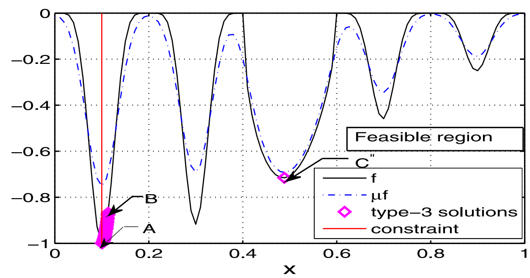


Fig. 5. Solutions obtains using Form-3 robust formulation. The solutions are labeled as (x, f, sigma$_g$, sigma$_f$) i.e., A = (0.1000, −1.0000, 0.0021, 0.3952), B = (0.1138, −0.8668, 0.0191, 0.3899), and C" = (0.4888, −0.7152, 6, 3.190).

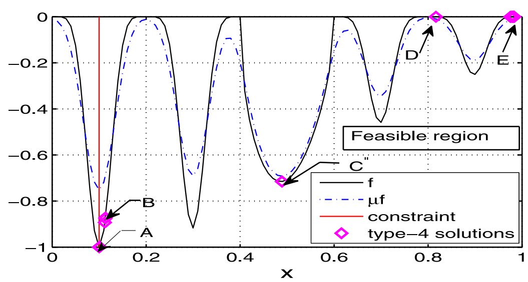


Fig. 6. Solutions obtains using Form-4 robust formulation. The solutions are labeled as (x, f, sigma$_g$, sigma$_f$) i.e., A = (0.1000, −1.000, 0.0012, 0.3952), B = (0.1136, −0.8715, 0.3948, 0.3937), C" = (0.4892, −0.7151, 6, 3.3624), D = (0.8169, −1.057e-4, 6, 5.4359), and E = (0.9832, −5.89e-5, 6, 6).

robustness. As seen from Fig. 5, the profile of the function differs from that of the expected function for the given uncertainties. The second minimum of the expected value function is located near the third minimum of the original function. The solution C" is now located near the third minimum of the original function since its expected value is lower than solutions around C in earlier plots (i.e., near the second minimum of the original function). Solutions around the second and the third minimum of the original function have sigma$_g$ = 6.

Use of Form-3 formulation results in a set of tradeoff solutions with feasibility robustness sigma$_g$ between 0.0021 (Solution A) and 6.000 (Solution C"). Once again, their corresponding sigma$_f$s can be computed, which span between 0.3899 and 3.190.

When the performance robustness is considered in addition to feasibility robustness (Form-4), two new solutions D and E are identified as shown in Fig. 6. Solution E has sigma$_f = 6$ while solution D has a marginally lower sigma$_f$. One can observe that the use of Form-4 allows identification of solutions spanning a large range of feasibility robustness (sigma$_g$'s between 0.0012 and 6) and performance robustness (sigma$_f$'s between 0.3952 and 6).

The sigma$_f$ versus sigma$_g$ plots for the solutions obtained from the four formulations are presented in Fig. 7. One can clearly observe that the solutions obtained from formulation Form-4 have a larger spread in sigma$_f$ and sigma$_g$ when compared with the solutions obtained from Form-3 . The line plots on the right convey the same information and have been included for consistency with respect to the results presented later involving multiple constraints and/or objectives.

The results obtained using the presented formulations can be summarized as follows.

1) Formulations Form-1 and Form-3 consider feasibility robustness only. Therefore the solutions obtained using these formulations span feasibility robustness range, but they may be concentrated in very small range in terms of performance robustness.
2) Formulations Form-2 and Form-4 consider both feasibility and performance robustness. The tradeoff set of solutions obtained using such formulations spans the complete possible range of feasibility robustness and performance (or expected performance) robustness.
3) In the event the landscapes of the performance function and expected performance function are the same (or very close to each other), there would be no difference in the outcome if one chooses to use Form-1 instead of Form-3 or Form-2 instead of Form-4. Since such an assertion cannot be made in general (as was the case in above example), especially for black-box functions, use of Form-4 is recommended overall. If only feasibility robustness is required, Form-3 can be used. For the objective functions where it is a priori known that function and expected function values are very close under given uncertainties, Form-1 or Form-2 (as required) should be used instead, as it will save the computational effort required in calculating the expected values at each design point.

B. *Example-2 (Robust Multiobjective Optimization)*

The second example is a multiobjective optimization problem [19] defined as

$$\begin{aligned}
&\underset{(\mathbf{x},\mathbf{y})}{\text{Minimize}}\ f_1(\mathbf{x},\mathbf{y}) = x \\
&\underset{(\mathbf{x},\mathbf{y})}{\text{minimize}}\ f_2(\mathbf{x},\mathbf{y}) = \frac{1+y}{x} \qquad (16)
\end{aligned}$$

$$\begin{aligned}
&\text{subject to} \\
&\quad y + 9x - 6 \geq 0 \\
&\quad -y + 9x - 1 \geq 0 \\
&\quad 0.1 \leq \mathbf{x} \leq 1,\ 0 \leq \mathbf{y} \leq 5. \qquad (17)
\end{aligned}$$

The problem has two variables both of which have Gaussian uncertainty with $\sigma_x^2 = 5 \times 10^{-3}$ and $\sigma_y^2 = 5 \times 10^{-3}$. Considering a generic case where the landscape of the objective function and the expected value of the objective function may not be the same, we use Form-3 and Form-4 formulations to identify robust solutions. The solutions obtained from Form-3 robust formulation have distinct non-dominated fronts corresponding to different values of sigma$_g$. In Fig. 8(a), the non-dominated fronts (in expected objective function space) corresponding to sigma$_g$ = 0, 1, 3, and 6 are presented along with the constraint boundaries (g_1 and g_2).

It can be seen that for sigma$_g$ = 0, the solutions lie near the active constraint boundary. The solutions progressively move away from the front as sigma$_g$ increases. The robust measures of the solutions are presented in Fig. 8(b). A user defined limit of $\sigma_{f_{0,1}} = 0.1012$ and $\sigma_{f_{0,2}} = 1.6959$ have been used for performance robustness. Since performance robustness is

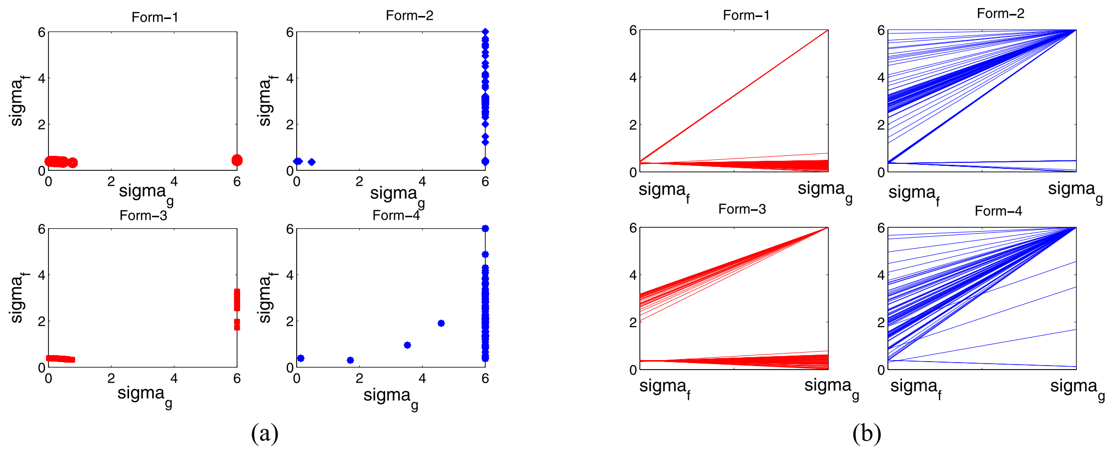


Fig. 7. Obtained solutions for example-1 (a) $sigma_f$ versus $sigma_g$ plots. (b) Line plots with values of $sigma_f$ and $sigma_g$.

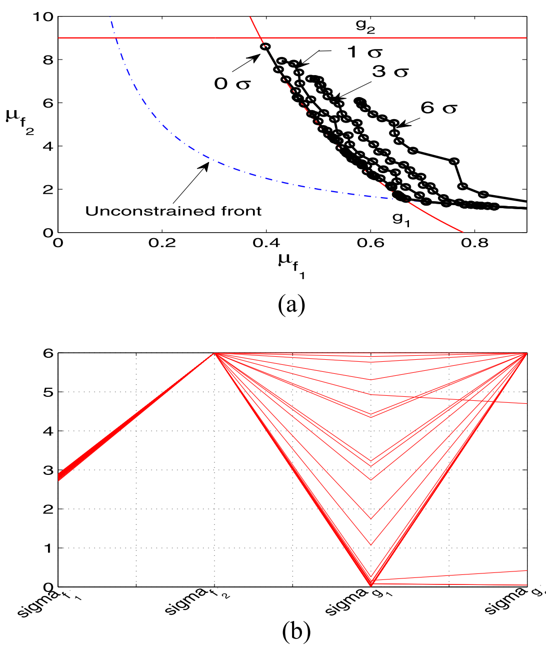


Fig. 8. Obtained (a) tradeoff frontiers of μ_{f_1} and μ_{f_2} for different values of sigma$_g$. (b) Performance of the obtained solutions with respect to robust measures sigma$_f$ and sigma$_g$ for a run using Form-3.

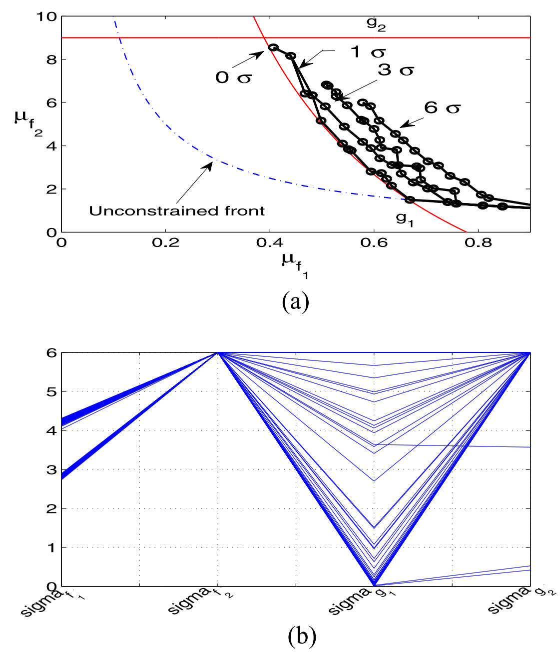


Fig. 9. Obtained (a) tradeoff frontiers of μ_{f_1} and μ_{f_2} for different values of sigma$_g$. (b) Performance of solutions with respect to robust measures sigma$_f$ and sigma$_g$ from a run using Form-4.

not considered in Form-3, all the solutions have sigma$_f$'s of around 3.0. However, since both feasibility and performance robustness are considered in Form-4, the tradeoff set as presented in Fig. 9 contains solutions with sigma$_f$ ranging from 3 to 4 and sigma$_g$ ranging from 0 to 6. One can also notice that, apart from the overall performance and/or feasibility robustness, the sigma levels of each constraint/objective (i.e., sigma$_{g_1}$ and sigma$_{g_2}$) in the figures also allow us to identify the constraint (or objective) that is most prone to be violated. For example, constraint g_1 has lower sigma$_g$ values and hence a larger probability of being violated compared to constraint g_2, for which sigma$_g$ values for all the solutions are close to 6.

VI. Robust Engineering Design Problems

In this section, we present the results of DBEA-r on two single objective, one multiobjective and a many-objective robust engineering design problem. The problem descriptions are provided in Table IV, while the results are shown in Table V.

The results obtained for welded beam design problem using Form-3 and Form-4 are presented in Figs. 10 and 11. One can observe that Form-3 is capable of delivering feasibility robust solutions. However, such solutions do not have good diversity with respect to performance robustness (notice the small spread in sigma$_f$). Form-4 on the other hand offers the tradeoff solutions spanning the complete range of feasibility and performance robustness.

To illustrate the benefits of proposed approach in improving robustness/providing additional solutions, a comparison is made using Form-4 with two other experiments reported for the problem [92]. They use two formulations to solve the problem, i.e., DFMOSS and DFSS, which are as follows: 1) DFSS: Min$[w_\mu \mu_f + w_\sigma \sigma_f^2]$ subject to $\mu_f + n\sigma_f \leq 3$

TABLE IV
PROBLEM DESCRIPTIONS

Problem	Problem description	σ_x	σ_f	Number of Objectives
Design of a welded beam (SO-1)	The welded beam design optimization problem was originally formulated in [89]. The problem is to design a welded beam for minimum cost subject to a set of constraints. The beam is designed to support a force F = 6000 lbf and the objective is to find the design with the minimum fabrication cost, considering four design variables i.e., thickness of the weld (x_1), length of the weld (x_2), thickness of the beam (x_3), and width of the beam (x_4) with the measurement unit in inches.	The uncertainty of the variables x_1, x_2, x_3, and x_4 follow a Gaussian distribution with $\sigma^2_{x_1} = 8.33 \times 10^{-4}$, $\sigma^2_{x_2} = 8.33 \times 10^{-4}$, $\sigma^2_{x_3} = 4.2 \times 10^{-3}$ and $\sigma^2_{x_4} = 4.2 \times 10^{-3}$.	The user defined limit on the variation on the objective function is prescribed as $\sigma_{f_{0,1}}$ = 1.701.	1
Design of a compression spring (SO-2)	The problem is to minimize the weight of a tension/compression spring, subject to constraints of minimum deflection, shear stress, surge frequency, and limits on outside diameter [90], [91]. The problem has three design variables: the wire diameter x_1, the mean coil diameter x_2, and the number of active coils x_3.	The uncertainty in the variables x_1, x_2 and x_3 follow a Gaussian distribution with $\sigma^2_{x_1} = 1.67 \times 10^{-5}$, $\sigma^2_{x_2} = 1.67 \times 10^{-5}$ and $\sigma^2_{x_3} = 1.67 \times 10^{-3}$.	The allowable functional variation limit set by the user is prescribed as $\sigma_{f_{0,1}}$ = 0.0101.	1
Car Side Impact problem (MO-3)	The problem aims to minimize the weight of a car and the average of three rib deflections constraints i.e., $g_5(\mathbf{x})$, $g_6(\mathbf{x})$ and $g_7(\mathbf{x})$ subject to the constraints of abdomen load, pubic force, velocity of V-Pillar, rib deflection etc [82]. The problem has eleven design variables: the thickness of V-pillar inner x_1, thickness of V-pillar reinforcement x_2, thickness of floor side inner x_3, thickness of cross members x_4, thickness of door beam x_5, thickness of door belt-line reinforcement x_6, thickness of roof rail x_7, material of V-pillar inner x_8, material of floor side inner x_9, barrier height x_{10}, barrier hitting position x_{11}.	The uncertainty in the variables x_1 to x_7 follow Gaussian distribution with given standard deviations:$\sigma^2_{x_1} = 5 \times 10^{-4}$, $\sigma^2_{x_2} = 5 \times 10^{-4}$, $\sigma^2_{x_3} = 5 \times 10^{-4}$, $\sigma^2_{x_4} = 5 \times 10^{-4}$, $\sigma^2_{x_5} = 8.33 \times 10^{-4}$, $\sigma^2_{x_6} = 5 \times 10^{-4}$ and $\sigma^2_{x_7} = 5 \times 10^{-4}$.	The allowable functional variations are set as $\sigma_{f_{0,1}}$ = 1.080 and $\sigma_{f_{0,2}}$ = 0.901.	2
Water resource management (MaO-4)	The water resource management problem was first described in [92] . The problem has three design variables: local detention storage capacity x_1, maximum treatment rate x_2 and the maximum allowable overflow rate x_3. The objective functions to be minimized are f_1 = drainage network cost, f_2 = storage facility cost, f_3 = treatment facility cost, f_4 = expected flood damage cost, and f_5 = expected economic loss due to flood.	Three of the variables, local detention storage capacity x_1, maximum treatment rate x_2 and the maximum allowable overflow rate x_3, are considered to vary as Gaussian distribution with $(\sigma^2_{x_1}, \sigma^2_{x_2}, \sigma^2_{x_3}) = (3.33 \times 10^{-4}, 1.67 \times 10^{-4}, 1.67 \times 10^{-4})$.	The acceptable functional variations are prescribed as $\sigma_{f_{0,1}}$ = 3000, $\sigma_{f_{0,2}}$ = 10, $\sigma_{f_{0,3}}$ = 35961, $\sigma_{f_{0,4}}$ = 59292 and $\sigma_{f_{0,5}}$ = 27457.	5

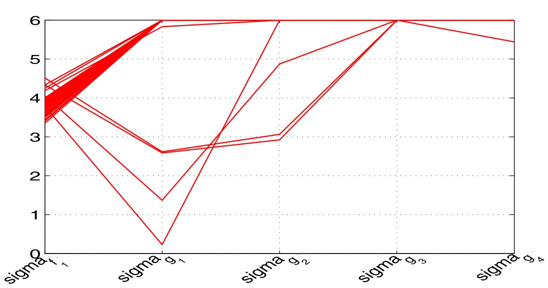


Fig. 10. Line plot with values of sigma$_{f_i}$ and sigma$_{g_j}$ for the solutions obtained using Form-3 for welded beam design problem.

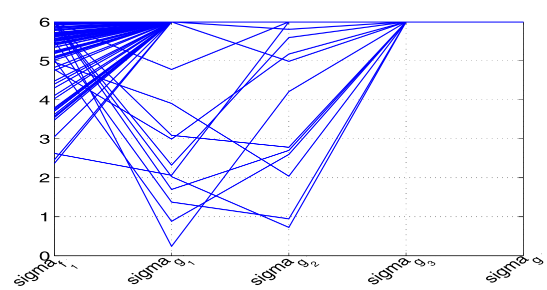


Fig. 11. Line plot with values of sigma$_{f_i}$ and sigma$_{g_j}$ for the solutions obtained using Form-4 for welded beam design problem.

and 2) DFMOSS: Min[μ_f, σ_f]. The uncertainty of all the variables follow a Gaussian distribution with its standard deviation (σ_x) of 0.01 and the acceptable performance deviation $\sigma_{f_0} = 0.4045$. As reported in [92], this paper uses NSGA-II [70] as a multiobjective evolutionary algorithm for DFMOSS and DFSS; while DBEA-r is used for Form-4 to generate the results for the presented study. Population size, sample size and maximum number of generations are set as 100, 100, and 400 respectively. For the DFSS optimization (which provides one solution in one run), seven optimization runs were performed with different weighing factors (w_μ:w_σ = 1000:1, 100:1, 10:1, 1:1, 1:10, 1:100, 1:1000) and $n = 6$. Fig. 12 shows nondominated solutions in terms of μ_f and sigma$_f$ in the final populations, whereas Fig. 13 shows all solutions in the final populations obtained using the three robust formulations.

It can be seen from Fig. 12 that all three robust formulations delivered robust solutions with sigma$_f$ level of 6; however, among these solutions (with sigma$_f = 6$), the expected performance of the robust solution using DBEA-r is better. Additionally, the proposed approach also provides solutions

TABLE V
COMPARISON BETWEEN DBEA-R, MOEA/D, AND SMS-EMOA ALGORITHMS FOR SINGLE OBJECTIVE PROBLEMS 1, 2 AND MULTIOBJECTIVE PROBLEMS 3, 4**

Prob.	Algorithm	Robust Form	HV'	HV^*	HV
SO-1	DBEA-r	1	-	-	**5.7089**
	MOEA/D		-	-	5.4787
	SMS-EMOA		-	-	5.2627
	DBEA-r	2	-	-	23.3325
	MOEA/D		-	-	16.0554
	SMS-EMOA		-	-	25.5567
	DBEA-r	3	-	-	3.7232
	MOEA/D		-	-	5.4829
	SMS-EMOA		-	-	5.8139
	DBEA-r	4	-	-	**38.0861**
	MOEA/D		-	-	32.6947
	SMS-EMOA		-	-	37.8455
SO-2	DBEA-r	1	-	-	**0.2105**
	MOEA/D		-	-	0.1742
	SMS-EMOA		-	-	0.1923
	DBEA-r	2	-	-	**0.2149**
	MOEA/D		-	-	0.1978
	SMS-EMOA		-	-	0.2078
	DBEA-r	3	-	-	**0.2823**
	MOEA/D		-	-	0.2713
	SMS-EMOA		-	-	0.2745
	DBEA-r	4	-	-	**0.1997**
	MOEA/D		-	-	0.1829
	SMS-EMOA		-	-	0.1895
MO-3	DBEA-r	1	**69.5953**	31.0741	231.8068
	MOEA/D		66.7295	32.9402	225.6509
	SMS-EMOA		67.1285	32.9354	247.7618
	DBEA-r	2	95.6037	45.9086	135.2737
	MOEA/D		102.270	47.9154	109.9465
	SMS-EMOA		104.1910	48.4418	145.1832
	DBEA-r	3	72.5755	46.4866	**290.3055**
	MOEA/D		79.6860	47.6692	259.2064
	SMS-EMOA		82.1860	48.2275	289.9788
	DBEA-r	4	91.9530	52.8294	153.2759
	MOEA/D		96.1942	53.6699	133.0404
	SMS-EMOA		98.8940	55.9082	166.1503
MaO-4	DBEA-r	1	0.9412	0.0146	0.0531
	MOEA/D		0.9373	0.0168	0.0476
	SMS-EMOA		0.9539	0.0123	0.0635
	DBEA-r	2	**1.2340**	0.0241	**0.0781**
	MOEA/D		1.1354	0.0253	0.0690
	SMS-EMOA		1.2015	0.0263	0.0655
	DBEA-r	3	**0.9923**	0.0173	0.0463
	MOEA/D		0.9813	0.0182	0.0428
	SMS-EMOA		0.9827	0.0178	0.0624
	DBEA-r	4	**1.2856**	0.0262	**0.0720**
	MOEA/D		1.2527	0.0266	0.0560
	SMS-EMOA		1.2766	0.0278	0.0556

** HV' refers the hypervolume on deterministic non-dominated objectives and HV^* refers the hypervolume on robust original function space, whereas HV refers the hypervolume on the objectives considering robustness.

which are close to six-sigma robust in both performance and feasibility, as observed from Fig. 13 which shows the line plots of sigma$_f$ and sigma$_g$ from all three formulations. It is noticeable that the robust solutions achieved from DBEA-r are much more diverse. Similar benefits can are anticipated for other engineering problems that follow, and comparison for them with past approaches is omitted for the sake of brevity.

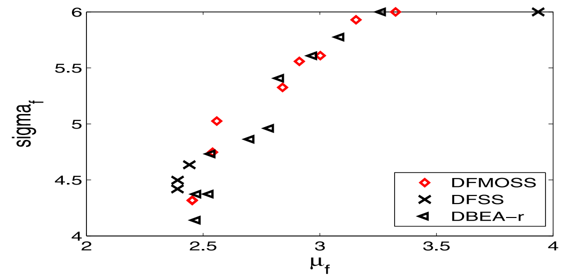


Fig. 12. Comparison of solutions obtained using robust formulations DFMOSS, DFSS [92], and Form-4 for welded beam design problem.

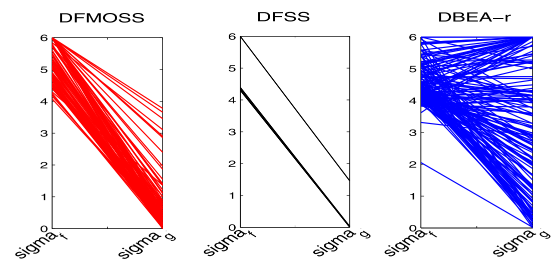


Fig. 13. Line plots with values of sigma$_{f_i}$ and sigma$_{g_j}$ for the solutions obtained using DFMOSS, DFSS, and Form-4 for welded beam design problem.

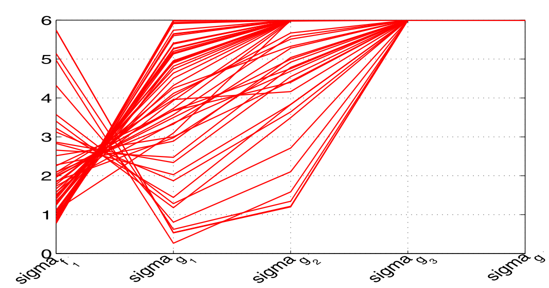


Fig. 14. Line plot with values of sigma$_{f_i}$ and sigma$_{g_j}$ for the solutions obtained using Form-3 for compression spring design problem.

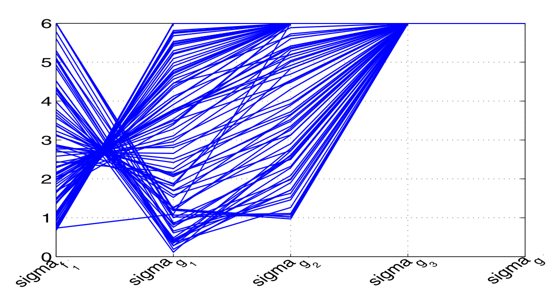
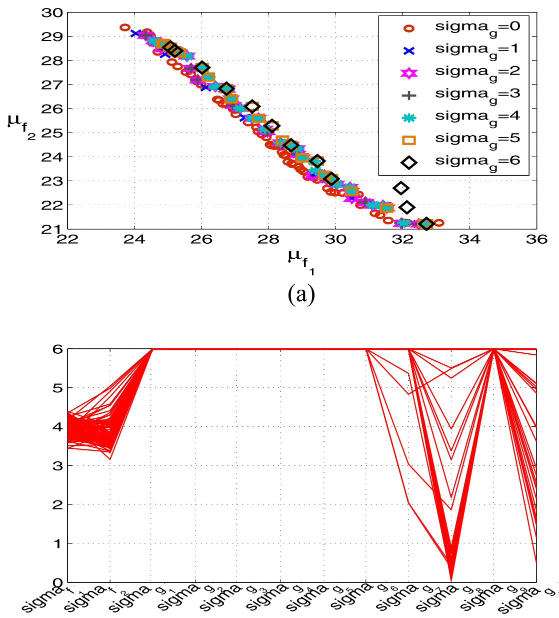


Fig. 15. Line plot with values of sigma$_{f_i}$ and sigma$_{g_j}$ for the solutions obtained using Form-4 for compression spring design problem.

The results obtained for compression design problem using Form-3 and Form-4 are presented in Figs. 14 and 15. One can again observe that Form-4 delivers solutions spanning a range of sigma$_f$.

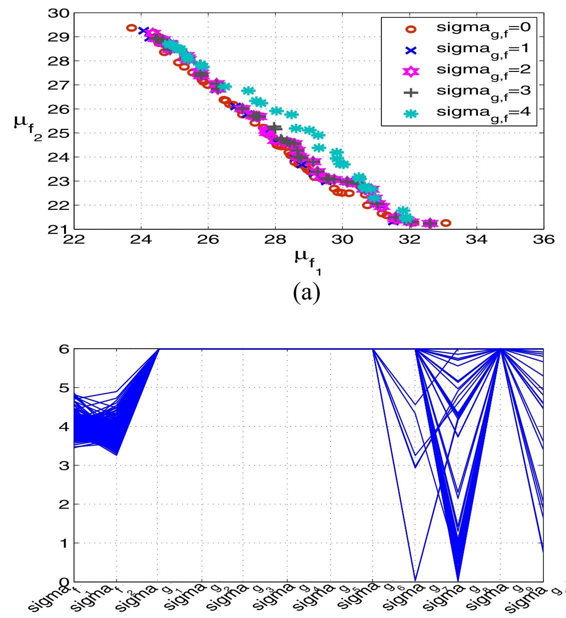
The next problem considered is a multiobjective (car side impact) optimization problem. The robust non-dominated solutions (in expected performance function space) achieved



(a)

(b)

Fig. 16. Obtained (a) nondominated robust solutions. (b) Line plot with values of sigma$_{f_i}$ and sigma$_{g_j}$ for the solutions obtained using Form-3 for car side impact problem.



(a)

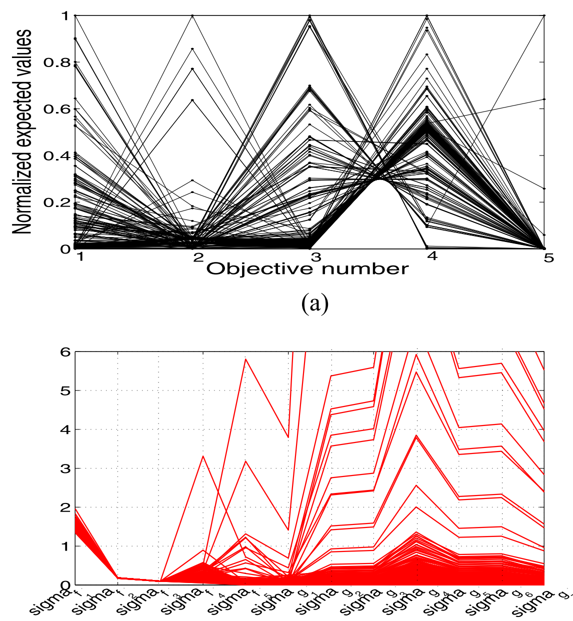
(b)

Fig. 17. Obtained (a) nondominated robust solutions. (b) Line plot with values of sigma$_{f_i}$ and sigma$_{g_j}$ for the solutions obtained using Form-4 for car side impact problem.

for the problem using Form-3 and Form-4 are shown in Figs. 16 and 17. The fronts corresponding to various levels of sigma$_g$ using Form-3 are shown in Fig. 16(a), while fronts corresponding to various levels of sigma$_g$ and sigma$_f$ is presented in Fig. 17(a). Corresponding values of sigma$_g$ and sigma$_f$ are plotted in Figs. 16(b) and 17(b). Feasibility robustness of sigma$_g$ = 6 is achieved by solutions identified using both the formulations. In the event the user requires solutions with sigma$_g$ = 4 and sigma$_f$ = 4 [i.e., both feasibility and performance robust solutions, Fig. 17(a)], the expected performance (plotted using $*$) is observed to be inferior than solutions requiring only sigma$_g$ to be greater than 4 [i.e., only feasibility robustness, Fig. 16(a)].

The performance of the approach is further analyzed using a many-objective (water resource management) optimization problem. For visualization of solutions, parallel coordinate plots, proposed in [93] and subsequently adopted in various works [94], [95] have been used. Parallel coordinate plots transform the M-dimensional objective space to two dimensions, by plotting normalized values of the objectives on M parallel axes (each of which correspond to one objective). Each solution is plotted by joining the its objective value on each axis by a straight line to the ones on neighboring axes. Evidently, two non-dominated solutions will intersect at least once in this space, and higher number of intersections will qualitatively indicate higher degree of conflict among the objectives (although depending on the objective values, the objectives may be conflicting in some segments even if they are not intersecting).

The parallel coordinate plots presented in Figs. 18 and 19 show expected objective function values with the upper limits [76191, 573.15, 2762800, 1342600 7972] and lower limits



(a)

(b)

Fig. 18. (a) Parallel coordinate plot of the non-dominated front obtained using Form-3. (b) Corresponding line plot with the values of sigma$_f$, sigma$_g$ for water resource management problem.

[63831, 295.51, 282680, 245010, 920]. These limits are computed using the non-dominated solutions obtained from the robust formulations. Figs. 18 and 19 show the robust non-dominated solutions obtained using Form-3 and Form-4 robust formulations and their corresponding robust measures. It can be seen that solutions identified using Form-3 robust formulation have a relatively uniform spread for most objectives. But

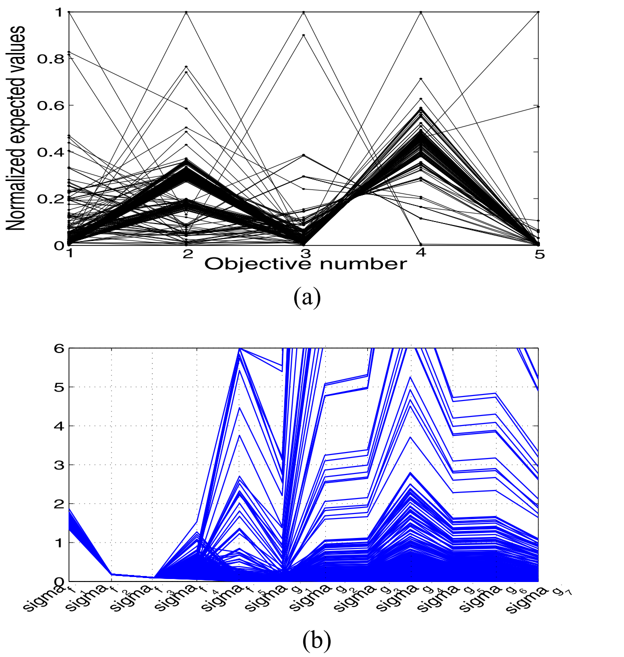


Fig. 19. (a) Parallel coordinate plot of the non-dominated front obtained using Form-4. (b) Corresponding line plot with the value of sigma$_f$, sigma$_g$ for water resource management problem.

for Form-4 robust formulation, which demands both feasibility and performance robustness, some of the solutions seem to lie closer to each other than the remaining; and also, higher (inferior) values in Objective 2 (on account of corresponding higher performance robustness) can be seen compared to that in Form-3, indicating that the adding the performance robustness objective (Form-4) has resulted in the solution set to converge in a different design space compared to Form-3. One can also note that the solutions from Form-4 formulation have a higher range of sigma$_f$s. While solutions with sigma$_g$ = 6 is achievable via both Form-3 and Form-4 formulations, sigma$_f$ limits the overall robustness of the solution. Such information assists in the identification of critical constraints/performance functions i.e., which constraints or performance function variations should be targeted to improve the overall robustness of the solution.

While the above study highlighted the effects of various formulations and the ability of DBEA-r to solve them, the next study compares the performance of DBEA-r with other many-objective optimization algorithms, SMS-EMOA [96] and MOEA/D [97]. These algorithms have been chosen for comparison as they are frequently used in the context of many-objective optimization. While these algorithms in their original form can only deal with unconstrained problems, we have included a feasibility first constraint handling scheme in both algorithms to deal with the constrained optimization problems studied in this paper. Table V presents the comparison of the results obtained using DBEA-r, SMS-EMOA, and MOEA/D on all the robust formulations for all the engineering design problems. DBEA-r offers competitive performance as compared to others based on the hypervolume measure. Three hypervolume measures have been considered here: $HV^{'}$, which refers to the hypervolume for non-dominated solutions in original function space (f); HV^{*} which refers to the hypervolume in expected objective function space (μ_f), and HV which refers to hypervolume considering expected objective values (μ_f) as well as robustness measures (sigma$_f$, $sigma_g$). Nadir point obtained from accumulation of all solutions across all runs has been used as a reference point for the calculation of hypervolumes. The performance of DBEA-r, wherever superior to other algorithms, is shown in bold. From the table, it can be observed that DBEA-r provides the best solution in terms of HV for most of the cases (9 out of 16). In most of the other cases, it is second only to SMS-EMOA. However, SMS-EMOA uses hypervolume-based measure during evolution, the computation time for which increases exponentially with number of objectives. Consequently, DBEA-r still has some advantage in terms of exhibiting much faster run-times compared to SMS-EMOA. The results highlight the effectiveness of DBEA-r in solving many-objective problems and the resulting advantage in solving robust optimization problems. The values for HV^{*} obtained using DBEA-r are lower than other two algorithms, which indicates that the solutions obtained using SMS-EMOA and MOEA/D were better in expected objective space, but relatively worse in terms of robustness measures. In terms of $HV^{'}$, DBEA-r is again better than the other two for half of the cases, but the main takeaway from these $HV^{'}$ values is that the algorithm ranking is very different for $HV^{'}$ compared to HV^{*}, emphasizing that original and expected objective space landscapes could be, in general, quite different. Therefore wherever possible, the algorithms should operate in expected space (as done in Form-3,4).

VII. Summary and Future Directions

Practical solutions of real life problems need to be robust. In this paper, we have presented and analyzed four formulations for robust optimization problems involving uncertain variable values. Robustness is quantified using six-sigma measures, wherein the uncertainties associated with the variables are assumed to follow a Gaussian distribution. Robustness has been studied from two perspectives: 1) feasibility robustness, i.e., robustness of solutions in terms of failure (violation of any constraint) and 2) performance robustness, i.e., robustness assuring good performance. The difference between setting the objective as performance function v/s expected value of the performance function has been discussed. The problems are formulated as many objective optimization problems and a DBEA has been used for the solution.

The contributions of the presented work can be summarized as follows.

1) Four different robust formulations are presented and analyzed. The differences between the formulations and their capabilities are highlighted from two perspectives, i.e., feasibility robustness and performance robustness. Form-1 and Form-3 formulations are capable of identifying feasibility robust solutions only, while Form-2 and Form-4 are both capable of identifying solutions that span the range of feasibility and performance robustness.

Form-4 is recommended over Form-2 (and overall) as it considers expected performance. Treating the robustness measures as additional objectives instead of constraints delivers a set of solutions corresponding to different levels of robustness in a single run.

2) Robustness measures have been quantified using sigma$_g$ and sigma$_f$ values for feasibility and performance respectively. This way of quantifying robustness helps comparing the robustness of solution with respect to each objective/constraint on a common scale. Furthermore, for multiple constraints/objectives case, minimum values of the feasibility and performance robustness (among all the constraints/objectives), are considered as the measures for overall robustness, which is equivalent to promising this minimum level of robustness with respect to all constraints and objectives for the solution. This also helps in identifying the critical constraint(s)/objective(s) that limit the overall robustness of the solution.

3) A DBEA-r is used to solve the robust optimization problem. To the authors' knowledge, many-objective algorithms have not been previously used to deal with robust optimization problems. The performance of DBEA-r is compared to two other widely used algorithms, SMS-EMOA and MOEA/D. DBEA-r is able to deliver best results for most cases, and competitive results for others; thus highlighting its potential to solve single/multi/many-objective robust optimization problems. The MATLAB code for DBEA-r is available for download at http://seit.unsw.adfa.edu.au/research/sites/mdo/Ray/Research-Data/Matlab-DBEAr.rar.

The efficiency of the approach can be further improved through the use of more efficient sampling schemes. In our approach, the expected value of every solution is estimated using explicit mean of the neighboring sample points generated using latin hypercube sampling (LHS) with Gaussian distribution. Such an approach requires evaluation of a number of neighborhood solutions and could be computationally prohibitive. Use of the Polynomial chaos (PC)-based estimation [49], [67] is particularly attractive as it is able to estimate the fitness using far fewer points as compared to sampling only (even with efficient schemes like LHS). The results of Form-3 robust formulation of the multiobjective problem discussed in Example 2 is presented in Table VI for comparison. For this illustration, a sample of 15 LHS points with explicit mean is compared with PC-based estimate using six neighborhood samples. While both the approaches achieve nearly same level of hypervolume in original performance (objective) space (3.7046 and 3.7039) and expected performance function (objective) space (510.148 and 510.011), the PC-based approach uses less than half the number of function evaluations (39 600 versus 99 000).

TABLE VI
PERFORMANCE OF LHS AND PC **

Sampling	Estimated fitness	Sample points	Num. of FEs	HV^*	HV
LHS	Sample Mean	15	99,000	3.7046	510.148
LHS	Polynomial chaos	6	39,600	3.7039	510.011

** HV^* refers the hypervolume on expected objective function (μ_f) space, whereas HV refers to hypervolume considering expected objectives as well as robustness ($sigma_g$).

The reference direction–based scheme also has some inherent limitations. Although the approach has been quite promising in dealing with many-objective optimization problems, it requires appropriate means of scaling the objectives for creating the directions uniformly. In addition, the number of directions also increase rapidly with the number of objectives (for a given spacing). This in turn requires correspondingly large population size to cover all directions (as done in the presented work). Alternatively, smarter means of managing the solutions via archive need to be devised. This is a limitation with most other algorithms too as the number of solutions required to cover the Pareto front grows exponentially with number of objectives. Use of reference direction–based methods may also not be particularly attractive for problems with disconnected Pareto fronts, as there may not be any Pareto solutions along certain directions. Use of hypervolume as an indicator of convergence/diversity for driving an EA has also been used in the past with success; however, the hypervolume computation may be computationally expensive for high number of objectives.

Improvements in above mentioned directions are currently being explored by the authors.

VIII. CONCLUSION

The studies in robust optimization can broadly be classified in to three areas, namely, formulation of robust optimization problem, quantification of robustness, and effective search techniques. In this paper, improvements have been sought in each of these three areas. In terms of formulating the robust optimization problem, four different strategies are presented and analyzed. The proposed formulations offer a set of tradeoff robust solutions from two perspectives—feasibility robustness (robustness against failure) and performance robustness (robustness assuring good performance). The approach also provides means to identify critical constraints or performance functions that affect the overall robustness. For quantification of robustness, the notion of six-sigma quality measure commonly used in the industry is employed, both in terms of feasibility and performance robustness. The problem is posed as a multi/many-objective optimization problem and a decomposition-based evolutionary approach is proposed for solving it efficiently. The proposed many-objective optimization algorithm (DBEA-r) utilizes systematically generated reference directions to guide the search. In order to deal with constraints, an epsilon level comparison is used which is known to be more effective than methods employing feasibility first principles. The performance of the proposed approach and the implications of using various formulations are illustrated using two numerical examples and four engineering problems. The performance of DBEA-r is compared with two other state of the art many objective optimization algorithms SMS-EMOA [96] and MOEA/D [97] to establish its credibility. The approach is able to identify solutions with varying degrees of robustness in terms of feasibility and performance simultaneously.

REFERENCES

[1] D. A. DeLaurentis and D. N. Mavris, "Uncertainty modeling and management in multidisciplinary analysis and synthesis," in *Proc. 38th AIAA Aerosp. Sci. Meeting Exhibit.*, Reno, NV, USA, Jan. 2000.

[2] D. N. Mavris, D. A. DeLaurentis, O. Bandte, and M. A. Hale, "A stochastic approach to multi-disciplinary aircraft analysis and design," in *Proc. 36th Aerosp. Sci. Meeting Exhibit.*, Reno, NV, USA, Jan. 1998.

[3] O. Bandte, D. N. Mavris, and D. A. DeLaurentis, "Viable designs through a joint probabilistic estimation technique," in *Proc. 4th World Aviat. Congr. Expo.*, San Francisco, CA, USA, Oct. 1999.

[4] X. Gu, J. Renaud, S. Batill, R. Brach, and A. Budhiraja, "Worst case propagated uncertainty of multidisciplinary systems in robust design optimization," *Struct. Multidiscip. Optim.*, vol. 20, no. 3, pp. 190–213, 2000.

[5] D. Anthony and A. Keane, "Robust-optimal design of a lightweight space structure using a genetic algorithm," *AIAA J.*, vol. 41, no. 8, pp. 1601–1604, 2003.

[6] H. Greiner, "Robust optical coating design with evolution strategies," *Appl. Opt.*, vol. 35, no. 28, pp. 5477–5483, 1996.

[7] D. Wiesmann, U. Hammel, and T. Back, "Robust design of multilayer optical coatings by means of evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 2, no. 4, pp. 162–167, Nov. 1998.

[8] H.-G. Beyer and B. Sendhoff, "Robust optimization—A comprehensive survey," *Comput. Methods Appl. Mech. Eng.*, vol. 196, nos. 33–34, pp. 3190–3218, 2007.

[9] D. V. Arnold and H.-G. Beyer, "A general noise model and its effects on evolution strategy performance," *IEEE Trans. Evol. Comput.*, vol. 10, no. 4, pp. 380–391, Aug. 2006.

[10] D. V. Arnold and H.-G. Beyer, "Evolution strategies with cumulative step length adaptation on the noisy parabolic ridge," *Nat. Comput.*, vol. 7, no. 4, pp. 555–587, 2008.

[11] W. Chen, J. K. Allen, K.-L. Tsui, and F. Mistree, "A procedure for robust design: Minimizing variations caused by noise factors and control factors," *ASME J. Mech. Des.*, vol. 118, no. 4, pp. 478–485, 1996.

[12] G. Sun *et al.*, "Crashworthiness design of vehicle by using multiobjective robust optimization," *Struct. Multidiscipl. Optim.*, vol. 44, no. 1, pp. 99–110, 2011.

[13] S. Sundaresan, K. Ishii, and D. R. Houser, "A robust optimization procedure with variations on design variables and constraints," *Eng. Optim.*, vol. 24, no. 2, pp. 101–117, 1995.

[14] L. Wang, R. V. Grandhi, and D. A. Hopkins, "Structural reliability optimization using an efficient safety index calculation procedure," *Int. J. Numer. Methods Eng.*, vol. 38, no. 10, pp. 1721–1738, May 1995.

[15] K. Deb and H. Gupta, "Searching for robust Pareto-optimal solutions in multi-objective optimization," in *Evolutionary Multi-Criterion Optimization* (Lecture Notes in Computer Science), vol. 3410. Berlin, Germany: Springer, 2005, pp. 150–164.

[16] Y. Jin and B. Sendhoff, "Trade-off between performance and robustness: An evolutionary multiobjective approach," in *Evolutionary Multi-Criterion Optimization* (Lecture Notes in Computer Science), vol. 2632, C. M. Fonseca, P. J. Fleming, E. Zitzler, L. Thiele, and K. Deb, Eds. Berlin, Germany: Springer, 2003, pp. 237–251.

[17] J. A. Bennett and R. V. Lust, "Conservative methods for structural optimization," *AIAA J.*, vol. 28, no. 8, pp. 1491–1496, 1990.

[18] Z. Wang, H.-Z. Huang, and Y. Liu, "A unified framework for integrated optimization under uncertainty," *ASME J. Mech. Des.*, vol. 132, no. 5, pp. 1–8, 2010.

[19] K. Deb *et al.*, "Reliability-based optimization using evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 13, no. 5, pp. 154–174, Oct. 2009.

[20] D. H. Jung and B. C. Lee, "Development of a simple and efficient method for robust optimization," *Int. J. Numer. Methods Eng.*, vol. 53, no. 9, pp. 2201–2215, Mar. 2002.

[21] D. Frangopol and F. Moses, *Reliability-Based Structural Optimization*. London, U.K.: Chapman and Hall, 1994, ch. 13, pp. 492–570.

[22] A. S. Al-Harthy and D. M. Frangopol, "Reliability-based design of prestressed concrete beams," *J. Struct. Eng.*, vol. 120, no. 11, pp. 3156–3177, Nov. 1994.

[23] M. Kleiber, A. Siemaszko, and R. Stocki, "Interactive stability-oriented reliability-based design optimization," *Comput. Methods Appl. Mech. Eng.*, vol. 168, nos. 1–4, pp. 243–253, 1999.

[24] X. Du and W. Chen, "Sequential optimization and reliability assessment method for efficient probabilistic design," *ASME J. Mech. Des.*, vol. 126, no. 2, pp. 225–233, 2004.

[25] P. Koch, R.-J. Yang, and L. Gu, "Design for six sigma through robust optimization," *Struct. Multidiscipl. Optim.*, vol. 26, nos. 3–4, pp. 235–248, 2004.

[26] G. Lei *et al.*, "Robust design optimization of PM-SMC motors for six sigma quality manufacturing," *IEEE Trans. Magn.*, vol. 49, no. 7, pp. 3953–3956, Jul. 2013.

[27] K.-H. Lee, I.-S. Eom, G.-J. Park, and W.-I. Lee, "Robust design for unconstrained optimization problems using the Taguchi method," *AIAA J.*, vol. 34, no. 5, pp. 1059–1063, 1996.

[28] K.-H. Lee and G.-J. Park, "Robust optimization considering tolerances of design variables," *Comput. Struct.*, vol. 79, no. 1, pp. 77–86, 2001.

[29] K. Zaman and S. Mahadevan, "Robustness-based design optimization of multidisciplinary system under epistemic uncertainty," *AIAA J.*, vol. 51, no. 5, pp. 1021–1031, May 2013.

[30] M. Li, S. Azarm, and V. Aute, "A multi-objective genetic algorithm for robust design optimization," in *Proc. Conf. Genet. Evol. Comput. (GECCO)*, 2005, pp. 771–778.

[31] I. Das, "Robustness optimization for constrained nonlinear programming problems?" *Eng. Optim.*, vol. 32, no. 5, pp. 585–618, 2000.

[32] I. Paenke, J. Branke, and Y. Jin, "Efficient search for robust solutions by means of evolutionary algorithm and fitness approximation," *IEEE Trans. Evol. Comput.*, vol. 10, no. 4, pp. 405–420, Aug. 2006.

[33] Y. Jin and J. Branke, "Evolutionary optimization in uncertain environments—A survey," *IEEE Trans. Evol. Comput.*, vol. 9, no. 3, pp. 303–317, Jun. 2005.

[34] R. Farmani, G. A. Walters, and D. A. Savic, "Trade-off between total cost and reliability for anytown water distribution network," *J. Water Resour. Plan. Manag.*, vol. 131, no. 3, pp. 161–171, 2005.

[35] X. Du and A. Sudjianto, "First-order saddlepoint approximation for reliability analysis," *AIAA J.*, vol. 42, no. 6, pp. 1199–1207, 2004.

[36] K. Deb and H. Gupta, "Introducing robustness in multi-objective optimization," *Evol. Comput.*, vol. 14, no. 4, pp. 463–494, 2006.

[37] H. Gupta and K. Deb, "Handling constraints in robust multiobjective optimization," in *Proc. IEEE Congr. Evol. Comput.*, Piscataway, NJ, USA, 2005, pp. 450–457.

[38] Z. P. Mourelatos and J. Liang, "A methodology for trading-off performance and robustness under uncertainty," *ASME J. Mech. Des.*, vol. 128, no. 4, pp. 856–863, 2005.

[39] B. D. Youn, K. K. Choi, and K. Yi, "Performance moment integration (PMI) method for quality assessment in reliability-based robust design optimization," *Mech. Based Des. Struct. Mach.*, vol. 33, no. 2, pp. 185–213, 2005.

[40] K.-H. Lee, I.-S. Eom, G.-J. Park, and W.-I. Lee, "A study on the robust design for unconstrained optimization problems," *KSME J. Mech. Des.*, vol. 18, no. 11, pp. 2825–2836, 1994.

[41] G.-J. Park, W.-J. Hwang, and W.-I. Lee, "Structural optimization postprocess using Taguchi method," *JSME Int. J. Ser. A Mech. Mater. Eng.*, vol. 37, no. 2, pp. 166–172, 1994.

[42] R. J. Balling, J. C. Free, and A. R. Parkinson, "Consideration of worst-case manufacturing tolerances in design optimization," *J. Mech. Trans. Autom.*, vol. 108, no. 4, pp. 438–441, 1986.

[43] S. Sundaresan, K. Ishii, and D. R. Houser, "Design optimization for robustness using performance simulation programs," *Eng. Optim.*, vol. 20, no. 3, pp. 163–178, 1992.

[44] J. Zhu and K.-L. Ting, "Performance distribution analysis and robust design," *ASME J. Mech. Des.*, vol. 123, no. 1, pp. 11–17, 2000.

[45] J. Su and J. E. Renaud, "Automatic differentiation in robust optimization," *AIAA J.*, vol. 35, no. 6, pp. 1072–179, 1997.

[46] A. Messac and A. I. Yahaya, "Multiobjective robust design using physical programming," *Struct. Multidiscipl. Optim.*, vol. 23, no. 5, pp. 357–371, 2002.

[47] M. S. Phadke, *Quality Engineering Using Robust Design*. Englewood Cliffs, NJ, USA: Prentice Hall, 1989.

[48] L. P. Chao, M. V. Gandhi, and B. S. Thompson, "A design-for-manufacture methodology for incorporating manufacturing uncertainties in the robust design of fibrous laminated composite structures," *J. Compos. Mater.*, vol. 27, no. 2, pp. 175–194, 1993.

[49] S. Poles and A. Lovison, "A polynomial chaos approach to robust multiobjective optimization," in *Hybrid and Robust Approaches to Multiobjective Optimization* (Dagstuhl Seminar Proceedings), K. Deb, S. Greco, K. Miettinen, and E. Zitzler, Eds. Dagstuhl, Germany: Schloss Dagstuhl—Leibniz-Zentrum fuer Informatik, 2009, pp. 1862–4405.

[50] S. Tsutsui, A. Ghosh, and Y. Fujimoto, "A robust solution searching scheme in genetic search," in *Parallel Problem Solving from Nature—PPSN IV*. Berlin, Germany: Springer, 1996, pp. 543–552.

[51] J. Branke, "Creating robust solutions by means of evolutionary algorithms," in *Parallel Problem Solving from Nature—PPSN V* (Lecture Notes in Computer Science), vol. 1498, A. E. Eiben, T. Bäck, M. Schoenauer, and H.-P. Schwefel, Eds. Berlin, Germany: Springer, 1998, pp. 119–128.

[52] B. Forouraghi, "A genetic algorithm for multiobjective robust design," *Appl. Intell.*, vol. 12, no. 3, pp. 151–161, 2000.

[53] S. Tsutsui and A. Ghosh, "Genetic algorithms with a robust solution searching scheme," *IEEE Trans. Evol. Comput.*, vol. 1, no. 3, pp. 201–208, Sep. 1997.

[54] J. Branke, "Reducing the sampling variance when searching for robust solutions," in *Proc. Genet. Evol. Comput. Conf. (GECCO)*, 2001, pp. 235–242.

[55] A. Sebald and D. Fogel, "Design of fault tolerant neural networks for pattern classification," in *Proc. 1st Annu. Conf. Evol. Program.*, 1992, pp. 90–99.

[56] A. Thompson, "On the automatic design of robust electronics through artificial evolution," in *Proc. Int. Conf. Evol. Syst.*, Lausanne, Switzerland, Sep. 1998, pp. 13–24.

[57] M. Rattray and J. Shapiro, "Noisy fitness evaluation in genetic algorithms and the dynamics of learning," *Found. Genet. Algorithm.*, vol. 4, no. 1, pp. 117–139, 1997.

[58] B. L. Miller and D. E. Goldberg, "Genetic algorithms, selection schemes, and the varying effects of noise," *Evol. Comput.*, vol. 4, no. 2, pp. 113–131, 1996.

[59] J. M. Fitzpatrick and J. J. Grefenstette, "Genetic algorithms in noisy environments," *Mach. Learn.*, vol. 3, nos. 2–3, pp. 101–120, 1988.

[60] C. Kim and K. K. Choi, "Reliability-based design optimization using response surface method with prediction interval estimation," *ASME J. Mech. Des.*, vol. 130, no. 12, pp. 1–12, 2007.

[61] M. D. McKay, R. J. Beckmkan, and W. J. Conover, "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, vol. 21, no. 2, pp. 239–245, 1979.

[62] D. H. Loughlin and S. R. Ranjithan, "Chance-constrained genetic algorithms," in *Genetic Evolutionary Computation*. San Francisco, CA, USA: Morgan Kaufmann, 1999, pp. 369–376.

[63] A. Harbitz, "An efficient sampling method for probability of failure calculation," *Struct. Safety*, vol. 3, no. 2, pp. 109–115, 1986.

[64] O. Ditlevsen and P. Bjerager, "Plastic reliability analysis by directional simulation," *ASCE J. Eng. Mech.*, vol. 115, no. 6, pp. 1347–1362, 1989.

[65] W. Chen, M. M. Wiecek, and J. Zhang, "Quality utility—A compromise programming approach to robust design," *ASME J. Mech. Des.*, vol. 121, no. 2, pp. 179–187, 2007.

[66] M. Putko, P. Newman, A. Taylor, III, and L. Green, "Approach for uncertainty propagation and robust design in CFD using sensitivity derivatives," in *Proc. AIAA 15th Comput. Fluid Dyn. Conf.*, 2001, pp. 2001–2528.

[67] N. Wiener, "The homogeneous chaos," *Amer. J. Math.*, vol. 60, no. 4, pp. 897–936, 1938.

[68] A. M. Hasofer and N. C. Lind, "Exact and invariant second-moment code format," *ASCE J. Eng. Mech. Div.*, vol. 100, no. 1, pp. 111–121, Jan./Feb. 1974.

[69] M. Hohenbichler, S. Gollwitzer, W. Kruse, and R. Rackwitz, "New light on first- and second-order reliability methods," *Struct. Safety*, vol. 4, no. 4, pp. 267–284, 1987.

[70] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

[71] H. Ishibuchi, N. Tsukamoto, and Y. Nojima, "Evolutionary many-objective optimization: A short review," in *Proc. IEEE Congr. Evol. Comput.*, Hong Kong, Jun. 2008, pp. 2419–2426.

[72] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm for multi-objective optimisation," in *Evolutionary Methods for Design, Optimisation, and Control*. Barcelona, Spain: CIMNE, 2002, pp. 95–100.

[73] M. Koppen and K. Yoshida, "Substitute distance assignment in NSGA-II for handling many-objective optimization problems," in *Evolutionary Multi-Criterion Optimization* (Lecture Notes in Computer Science), S. Obayashi, Ed., vol. 4403. Berlin, Germany: Springer, 2007, pp. 727–741.

[74] H. K. Singh, A. Isaacs, T. Ray, and W. Smith, "A study on the performance of substitute distance based approaches for evolutionary many objective optimization," in *Simulated Evolution and Learning* (Lecture Notes in Computer Science), X. Li, Ed., vol. 5361. Berlin, Germany: Springer, 2008, pp. 401–410.

[75] S. Kachroudi and M. Grossard, "Average rank domination relation for NSGA-II and SMPSO algorithms for many-objective optimization," in *Proc. 2nd World Congr. Nat. Biol. Inspired Comput. (NaBIC)*, Fukuoka, Japan, Dec. 2010, pp. 19–24.

[76] G. Wang and J. Wu, "A new fuzzy dominance GA applied to solve many-objective optimization problem," in *Proc. 2nd Int. Conf. Innov. Comput. Inf. Control (ICICIC)*, Kumamoto, Japan, Sep. 2007, pp. 617–621.

[77] H. Sato, H. Aguirre, and K. Tanaka, "Controlling dominance area of solutions and its impact on the performance of MOEAs," in *Evolutionary Multi-Criterion Optimization* (Lecture Notes in Computer Science), S. Obayashi, K. Deb, C. Poloni, T. Hiroyasu, and T. Murata, Eds., vol. 4403. Berlin, Germany: Springer, 2007, pp. 5–20.

[78] X. Zou, Y. Chen, M. Liu, and L. Kang, "A new evolutionary algorithm for solving many-objective optimization problems," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 38, no. 5, pp. 1402–1412, Oct. 2008.

[79] D. Hadka, P. M. Reed, and T. W. Simpson, "Diagnostic assessment of the Borg MOEA for many-objective product family design problems," in *Proc. IEEE World Congr. Comput. Intell.*, Brisbane, QLD, Australia, Jun. 2012, pp. 10–15.

[80] H. Aguirre and K. Tanaka, "Adaptive ϵ-ranking on MNK-landscapes," in *Proc. IEEE Symp. Comput. Intell. Multi Crit. Decis. Making (MCDM)*, 2009, pp. 104–111.

[81] M. Asafuddoula, T. Ray, and R. Sarker, "A decomposition based evolutionary algorithm for many objective optimization with systematic sampling and adaptive epsilon control," in *Evolutionary Multi-Criterion Optimization* (Lecture Notes in Computer Science), R. Purshouse, Ed., vol. 7811. Berlin, Germany: Springer, Mar. 2013, pp. 413–427.

[82] H. Jain and K. Deb, "An evolutionary many-objective optimization algorithm using reference-point based non-dominated sorting approach, part II: Handling constraints and extending to an adaptive approach," *IEEE Trans. Evol. Comput.*, vol. 18, no. 4, pp. 602–622, Aug. 2013.

[83] M. Asafuddoula, T. Ray, R. Sarker, and K. Alam, "An adaptive constraint handling approach embedded MOEA/D," in *Proc. IEEE World Congr. Comput. Intell.*, Brisbane, QLD, Australia, Jun. 2012, pp. 1–8.

[84] T. Takahama and S. Sakai, "Constrained optimization by applying the α constrained method to the nonlinear simplex method with mutations," *IEEE Trans. Evol. Comput.*, vol. 9, no. 5, pp. 437–451, Oct. 2005.

[85] I. Das and J. E. Dennis, "Normal-bounday intersection: A new method for generating Pareto optimal points in multicriteria optimization problems," *SIAM J. Optim.*, vol. 8, no. 3, pp. 631–657, Aug. 1998.

[86] H. K. Singh, A. Isaacs, and T. Ray, "A Pareto corner search evolutionary algorithm and dimensionality reduction in many-objective optimization problems," *IEEE Trans. Evol. Comput.*, vol. 15, no. 4, pp. 539–556, Aug. 2011.

[87] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. New York, NY, USA: Wiley, 2001.

[88] K. Ragsdell and D. Phillips, "Optimal design of a class of welded structures using geometric programming," *J. Eng. Ind.*, vol. 98, no. 3, pp. 1021–1025, 1976.

[89] A. Belegundu, "A study of mathematical programming methods for structural optimization," Ph.D. dissertation, Dept. Civil Environ. Eng., Univ. Iowa, Iowa City, IA, USA, 1982.

[90] J. Arora, *Introduction to Optimum Design*. New York, NY, USA: McGraw-Hill, 1989.

[91] K. Musselman and J. Talavage, "A trade-off cut approach lo multiple objective optimization," *Operat. Res.*, vol. 28, no. 6, pp. 1424–1435, 1980.

[92] K. Shimoyama, A. Oyama, and K. Fujii, "Development of multi-objective six-sigma approach for robust design optimization," *J. Aerosp. Comput. Inf. Commun.*, vol. 5, pp. 215–233, Aug. 2008.

[93] A. Inselberg, "The plane with parallel coordinates," *Vis. Comput.*, vol. 1, no. 2, pp. 69–91, 1985.

[94] R. C. Purshouse and P. J. Fleming, "Conflict, harmony, and independence: Relationships in evolutionary multi-criterion optimisation," in *Evolutionary Multi-Criterion Optimization*. Berlin, Germany: Springer, 2003, pp. 16–30.

[95] C. M. Fonseca *et al.*, “Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization.” in *Proc. 5th Int. Conf. Genet. Algorithm. (ICGA)*, vol. 93. 1993, pp. 416–423.
[96] N. Beume, B. Naujoks, and M. Emmerich, “SMS-EMOA: Multiobjective selection based on dominated hypervolume,” *Eur. J. Operat. Res.*, vol. 181, no. 3, pp. 1653–1669, Sep. 2007.
[97] Q. Zhang and H. Li, “MOEA/D: A multiobjective evolutionary algorithm based on decomposition,” *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007.

M. Asafuddoula received the B.Sc. degree in computer science and engineering from Chittagong University of Engineering and Technology, Chittagong, Bangladesh, in 2008 and the Ph.D. degree in computer science from University of New South Wales at Canberra, Canberra, ACT, Australia, in 2014.

He is a Research Associate with the School of Engineering and Information Technology, University of New South Wales at Canberra. His research interests include evolutionary optimization, constraint handling, many-objective optimization, robust design optimization, and multifidelity-based optimization.

Hemant K. Singh received the B.Tech. degree in mechanical engineering from IIT Kanpur, Kanpur, India, in 2007 and the Ph.D. degree in mechanical engineering from University of New South Wales (UNSW) Canberra, Canberra, ACT, Australia, in 2011.

He is a Lecturer with the School of Engineering and Information Technology, UNSW Canberra. His research interests include the development of evolutionary optimization methods for engineering design.

Tapabrata Ray received the B.Tech, M.Tech, and Ph.D. degree from the Indian Institute of Technology Kharagpur, Kharagpur, India.

He is an Associate Professor with the School of Engineering and Information Technology, University of New South Wales, Canberra, ACT, Australia. He leads the Multidisciplinary Design Optimization Group at UNSW Canberra. His research interests include evolutionary optimization, surrogate-assisted optimization, robust design optimization, and shape and topology optimization.

Dr. Ray is a Future Fellow of the Australian Research Council.