An Evolutionary Approach to Active Robust Multiobjective Optimisation

Shaul Salomon[1(✉)], Robin C. Purshouse[1], Gideon Avigad[2], and Peter J. Fleming[1]

[1] Department of Automatic Control and Systems Engineering, University of Sheffield, Mappin Street, Sheffield S1 3JD, UK
{s.salomon,r.purshouse,p.fleming}@sheffield.ac.uk
[2] Department of Mechanical Engineering, ORT Braude College of Engineering, Karmiel, Israel
gideona@braude.ac.il

Abstract. An Active Robust Optimisation Problem (AROP) aims at finding robust adaptable solutions, i.e. solutions that actively gain robustness to environmental changes through adaptation. Existing AROP studies have considered only a single performance objective. This study extends the Active Robust Optimisation methodology to deal with problems with more than one objective. Once multiple objectives are considered, the optimal performance for every uncertain parameter setting is a set of configurations, offering different trade-offs between the objectives. To evaluate and compare solutions to this type of problems, we suggest a robustness indicator that uses a scalarising function combining the main aims of multi-objective optimisation: proximity, diversity and pertinence. The Active Robust Multi-objective Optimisation Problem is formulated in this study, and an evolutionary algorithm that uses the hypervolume measure as a scalarasing function is suggested in order to solve it. Proof-of-concept results are demonstrated using a simplified gearbox optimisation problem for an uncertain load demand.

Keywords: Robust optimisation · Uncertainties · Multi-objective optimisation · Adaptation · Gearbox · Design

1 Introduction

When solving real-world optimisation problems, the physical system is represented by a model to predict the future performance of candidate solutions. As a result, uncertainties become an inseparable part of the optimisation process, and solutions need to be robust in addition to having good predicted performance. A solution is considered as robust if it is less affected by the negative effects of uncertainties.

The ability of many products to adapt to environment changes provides them with active robustness to uncertain operating conditions. The active robust optimisation (ARO) methodology was suggested in order to evaluate the added value

A. Gaspar-Cunha et al. (Eds.): EMO 2015, Part II, LNCS 9019, pp. 141–155, 2015.
DOI: 10.1007/978-3-319-15892-1_10

of adaptability [1]. Till date, ARO dealt with improvement of a single performance metric through adaptation. Since the majority of real-world optimisation problems involve several, often conflicting, objectives, this study extends the ARO methodology to deal with multi-objective optimisation problems (MOPs).

1.1 Robust Multi-objective Optimisation

A MOP can be formulated as:

$$\min_{\mathbf{x}\in\mathcal{X}} \mathbf{f}(\mathbf{x},\mathbf{p}), \tag{1}$$

where $\mathbf{f}$ is a vector of performance measures, $\mathbf{x}$ is a vector of design variables, $\mathcal{X}$ is the feasible domain defined by a set of equality and inequality constraints, and $\mathbf{p}$ is a vector of parameters that cannot be determined by the designer.

Since uncertainties exist in all real-world optimisation problems, they should be accommodated within the optimisation procedure. Uncertainties might be *epistemic*, resulting from discrepancies between the model used for optimisation and the real system, or *aleatory*, where the variables within the system inherently change from unit to unit or time to time.

In their review on robust optimisation, Beyer and Sendhoff [2] classified the sources of uncertainties as follows:

Type A uncertainties occur when the environmental parameters $\mathbf{p}$ are unknown (epistemic) or may change within an expected range (aleatory).
Type B uncertainties are present when the actual values of design variables $\mathbf{x}$ differ from their nominal values, identified by the optimisation procedure. The deviation might occur upon production (manufacturing tolerances) or during operation (deterioration).
Type C uncertainties relate to model inaccuracies in predicting the performance $\mathbf{f}$ of the candidate design. This may result from an incorrect or simplified description of the relationship between variables within the model.

If the uncertainties are not addressed during the optimisation, solution identified as 'optimal' may poorly perform when implemented in real life. Over the past two decades, robust optimisation (RO) has gained increasing popularity, with many studies aiming at identifying robust solutions rather than optimal solutions. When formulating a robust optimisation problem, robustness criteria are specified to determine how candidate solutions should be evaluated with respect to the uncertainties involved.

We use upper case letters to distinguish random variates from deterministic values. Whenever uncertainties of either Type A-C are concerned, the objective vector $\mathbf{f}$ becomes a random variate $\mathbf{F}$. In a robust optimisation scheme, the aim is to optimise the robustness criterion $I[\mathbf{F}]$, that holds some information about the distribution of $\mathbf{F}$:

$$\min_{x\in\mathcal{X}} I[\mathbf{F}(\mathbf{X},\mathbf{P})]\,. \tag{2}$$

The most common robustness criteria are the worst-case scenario (e.g., [3–5]), and aggregated values such as the mean value or the variance (e.g., [6–9]). Other criteria also exist, for example, the probability for the objective functions to be better than some predefined threshold [10], a minimum confidence level in performance [5], or performing within a predefined neighbourhood of some nominal performance vector [8].

Most of the existing evolutionary algorithms for multi-objective RO consider Type C uncertainties, represented by added noise to the nominal function values. The first evolutionary algorithm for robust MOPs were suggested in 2001 by Teich [6] and Hughes [7]. Teich suggested probabilistic dominance as an alternative to the dominance relation [6]. Hughes suggested a ranking scheme based on the sum of probabilities for each solution to be dominated [7]. Since then, several evolutionary optimisers were designed to account for Type C uncertainties [11–15].

Perturbation in design variables (Type B uncertainty) was addressed by [8,16], where each design was represented by a sampled set of designs within its neighbourhood. An algorithm aiming for reducing the amount of function evaluations for this scheme was introduced in [9].

To our knowledge, apart from previous work by the authors [1], there are no studies that explicitly treat Type A uncertainties with an evolutionary RO scheme. Instead, uncertain and dynamic environments are considered in the scope of *dynamic optimisation*, where the aim is to track a moving optimum, and remain optimal as the environment changes [17]. In dynamic optimisation the problem is deterministic, but it has to be re-solved every time the environment changes.

1.2 Active Robust Optimisation Methodology

The ARO methodology [1], is a special case of robust optimisation, where the product has some adjustable properties that can be modified by the user after the optimised design has been realised. These adjustable variables allow the product to adapt to variations of the uncontrolled parameters, so it can actively suppress their negative effect. The methodology makes a distinction between three types of variables: design variables $\mathbf{x}$, adjustable variables $\mathbf{y}$ and uncertain parameters $\mathbf{P}$, which cannot be controlled. A single realised vector of uncertain parameters from the random variate $\mathbf{P}$ is denoted as $\mathbf{p}$.

In a single-objective robust optimisation problem with Type A uncertainties, each realisation $\mathbf{p}$ is associated with a corresponding objective function value $f(\mathbf{x}, \mathbf{p})$, and a solution $\mathbf{x}$ is associated with a distribution of objective function values that correspond to the variate of the uncertain parameters $\mathbf{P}$. We denote this distribution as $F(\mathbf{x}, \mathbf{P})$. In ARO, for every realisation of the uncertain environment, the performance also depends on the value of the adjustable variables $\mathbf{y}$, i.e., $f \equiv f(\mathbf{x}, \mathbf{y}, \mathbf{p})$. Since the adjustable variables' values can be selected after $\mathbf{p}$ is realised, the solution can improve its performance by adapting its adjustable variables to the new conditions. In order to evaluate the solution's performance according to the robust optimisation methodology, it is conceivable that the $\mathbf{y}$

vector that yields the best performance for each realisation of the uncertainties will be selected. This can be expressed as the optimal configuration $\mathbf{y}^\star$:

$$\mathbf{y}^\star = \underset{\mathbf{y}\in\mathcal{Y}(\mathbf{x})}{\operatorname{argmin}} f(\mathbf{x}, \mathbf{y}, \mathbf{p}), \tag{3}$$

where $\mathcal{Y}(\mathbf{x})$ is the solution's domain of adjustable variables. it is also termed as the solution's adaptability.

Considering the entire environmental uncertainty, a one-to-one mapping between the scenarios in $\mathbf{P}$ and the optimal configurations in $\mathcal{Y}(\mathbf{x})$ can be defined as:

$$\mathbf{Y}^\star = \underset{\mathbf{y}\in\mathcal{Y}(\mathbf{x})}{\operatorname{argmin}} F(\mathbf{x}, \mathbf{y}, \mathbf{P}). \tag{4}$$

Assuming a solution will always adapt to its optimal configuration, its performance can be described by the following variate:

$$F(\mathbf{x}, \mathbf{P}) \equiv F(\mathbf{x}, \mathbf{Y}^\star, \mathbf{P}). \tag{5}$$

Following the above, the *Active Robust Opimisation Problem* is formulated:

$$\min_{\mathbf{x}\in\mathcal{X}} I\left[F(\mathbf{x}, \mathbf{Y}^\star, \mathbf{P})\right], \tag{6a}$$

$$\text{where:} \quad \mathbf{Y}^\star = \underset{\mathbf{y}\in\mathcal{Y}(\mathbf{x})}{\operatorname{argmin}} F(\mathbf{x}, \mathbf{y}, \mathbf{P}). \tag{6b}$$

It is a bi-level optimisation problem. In order to compute the objective function F in Eq. (6a), the problem in Eq. (6b) has to be solved for every solution $\mathbf{x}$, with the entire environment universe $\mathbf{P}$. To evaluate F, one may consider one or more robustness criteria $I[F]$.

2 Methodology

This study extends the single objective AROP in Eq. (6) to the following multi-objective formulation:

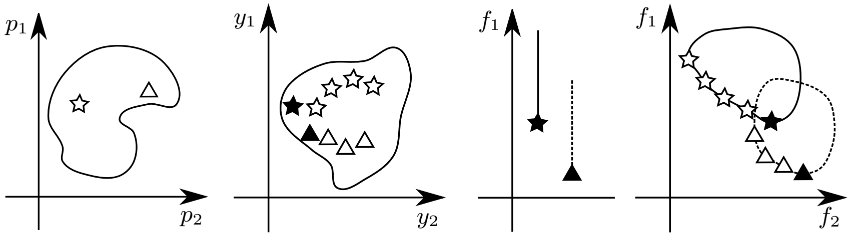
$$\min_{\mathbf{x}\in\mathcal{X}} I\left[\underline{\mathbf{F}}(\mathbf{x}, \underline{\mathbf{Y}^\star}, \mathbf{P})\right], \tag{7a}$$

$$\text{where:} \quad \underline{\mathbf{Y}^\star} = \underset{\mathbf{y}\in\mathcal{Y}(\mathbf{x})}{\operatorname{argmin}} \mathbf{F}(\mathbf{x}, \mathbf{y}, \mathbf{P}), \tag{7b}$$

where $\operatorname{argmin} \mathbf{F}$ is defined in terms of Pareto optimality, and the underscore notation is used to distinguish a set from a single point.

The most notable difference between Eq. (6) and Eq. (7) is that the solution $\underline{\mathbf{Y}^\star}$ in Eq. (7b) is a variate of Pareto optimal sets, rather than the variate of a single optimal configuration in Eq. (6b). Instead of a one-to-one mapping between $\mathbf{P}$ and $\mathbf{Y}^\star$, Eq. (7b) consists of a one-to-many mapping. As a result, Eq. (7a) minimises the variate of Pareto optimal frontiers $\underline{\mathbf{F}}(\mathbf{x}, \underline{\mathbf{Y}^\star}, \mathbf{P})$.

The difference between $f(\mathbf{x}, \mathbf{y}^{\star}, \mathbf{p})$ and $\underline{\mathbf{f}}\big(\mathbf{x}, \underline{\mathbf{y}^{\star}}, \mathbf{p}\big)$ is illustrated in Fig. 1. A candidate solution $\mathbf{x}$ is evaluated for two scenarios of the uncertain parameter space $\mathbf{P}$ (Fig. 1(a)). The performance of the solution for every scenario $\mathbf{p}$ (star or triangle) depends on its configuration $\mathbf{y}$ (Fig. 1(b)). In Fig. 1(c), f_1 is the only objective. All possible objective values are bounded by the solid and dashed lines for the star and triangle scenarios, respectively. The black stars and triangles in Figures 1(b) and 1(c) mark the optimal configuration and objective value for each scenario ($\mathbf{y}^{\star}$ and $f(\mathbf{x}, \mathbf{y}^{\star}, \mathbf{p})$, respectively). In Fig. 1(d) an additional objective is considered. Now all possible objective values are bounded by the solid and dashed contours, and the optimal configuration for each scenario consists of a set rather than a single configuration, denoted by the additional white shapes.



(a) Uncertain parameter space (b) Configuration space (c) One objective (d) Two objectives

Fig. 1. Optimal configurations of a candidate solution $\mathbf{x}$ for two scenarios of the uncertainties, associated with the environmental parameters

The problem in Eq. (7) is termed here as an Active Robust Multi-objective Optimisation Problem (ARMOP). It introduces a very challenging question: *How can adaptable products be evaluated and compared according to their variates of Pareto frontiers* $\underline{\mathbf{F}}(\mathbf{x}, \underline{\mathbf{Y}^{\star}}, \mathbf{P})$*?* In Section 2.1 we introduce a first attempt to address this challenge, and suggest a set-based robustness indicator. In Section 4 we demonstrate how this indicator can be integrated into an evolutionary algorithm in order to solve an ARMOP.

2.1 Evaluating a Variate of Sets

In order to evaluate a candidate solution for an ARMOP, we suggest using a robustness criterion that quantifies the variate of Pareto frontiers with a single scalar value. Keeping in mind there is no way to avoid the loss of meaningful information when using a scalarising function, we strive to extract as much information as possible regarding the quality of the trade-off surfaces $\underline{\mathbf{F}}(\mathbf{x}, \underline{\mathbf{Y}^{\star}}, \mathbf{P})$. Following the motivation in evolutionary multiobjective optimisation (EMO), an approximated solution to a MOP is evaluated according to three major qualities [18]: **proximity** of the approximated front to the true Pareto front (PF), **diversity** of the solutions, and **pertinence** to the preferred region of interest.

One of the well-known quality indicators for approximation sets is the hypervolume (HV), defined as the volume of objective space enclosed by the Pareto front and a reference point [19]. The HV measure provides an integrated measure of proximity, diversity and pertinence, although it is sensitive to the choice of a reference point [20]. Despite this drawback, we use it to demonstrate the concept of the robustness indicator suggested in this study.

Hypervolume-Based Robustness Indicator. Without loss of generality, we consider the variate $\mathbf{P}$ as a finite set of sampled scenarios $\mathbf{p}$. The HV of solution $\mathbf{x}$ for scenario $\mathbf{p}$ is denoted as $hv(\mathbf{x}, \mathbf{p})$. It is calculated according to the ideal vector $\mathbf{f}^*$ and the worst objective vector $\mathbf{f}^w$, which are the vectors with minimum and maximum objective values, respectively, amongst all known solutions and scenarios. The robustness indicator I_{hv} is derived as follows:
First, the objectives of $\mathbf{f}$ are normalised in a manner that supports DM's preferences (e.g., setting $\mathbf{f}^*$ to zero and $\mathbf{f}^w$ to a vector of weights between 0-1).
Next, the hypervolume $hv(\mathbf{x}, \mathbf{p})$ is calculated for each scenario $\mathbf{p} \in \mathbf{P}$, using the worst objective vector as a reference point. The variate of the hypervolume measure that corresponds to the variate $\mathbf{P}$ is denoted as $HV(\mathbf{x}, \mathbf{P})$.
Finally, a robustness criterion is used to evaluate the variate $HV(\mathbf{x}, \mathbf{P})$:

$$I_{hv}[\mathbf{F}(\mathbf{x}, \mathbf{P})] = I[HV(\mathbf{x}, \mathbf{P})]\,. \tag{8}$$

Since the aim is to maximise $HV(\mathbf{x}, \mathbf{P})$ and its value is bounded between 0-1, in a minimisation problem, the complement can be used:

$$I_{hv}[\mathbf{F}(\mathbf{x}, \mathbf{P})] = I[1 - HV(\mathbf{x}, \mathbf{P})]\,. \tag{9}$$

Fig 2 demonstrates the above procedure for a population of two solutions. Three scenarios of $\mathbf{p}$ are considered, where the Pareto frontiers of the two solutions are depicted in stars and circles. For scenario $\mathbf{p}_3$, dashed contours show the domains in objective space that include the performances of all evaluated configurations. The worst objective vector is calculated according to the objective vectors of all configurations, including non optimal ones. The variate $HV\left(\mathbf{x}\ \ , \mathbf{P}\right)$ is shown as the collection of three HVs for $\mathbf{x}$.

3 Case Study – Gearbox Optimisation Problem

We demonstrate our approach with a gearbox optimisation task for an uncertain load demand. A load with inertia J_L needs to be rotated at speed ω_L with a torque τ_L. All of the load parameters above may vary within known intervals. The torque is transmitted to the load from a geared motor system consisting of a DC motor and a two staged transmission with five gears. The gearbox optimisation problem, formulated as an ARMOP, searches for the number of teeth in each gear to minimise energy consumption and acceleration time. The system is evaluated at both steady state, i.e., operating at the load-speed scenarios (which are uncertain), and during transient conditions when accelerating from rest to each scenario. Three objectives are considered: power consumption in steady state (P), energy required to accelerate to steady state speed (E), and time to accelerate to steady state speed (T).

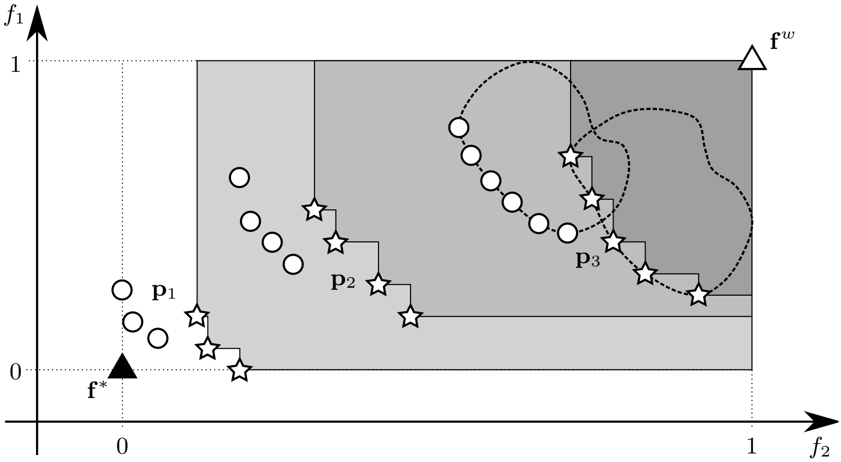


Fig. 2. Pareto frontiers $\underline{\mathbf{F}}(\mathbf{x}, \underline{\mathbf{Y}^\star}, \mathbf{P})$ of two solutions ($\mathbf{x}$ and $\mathbf{x}_\circ$) for three scenarios. The ideal vector is marked with a black triangle and the worst objective vector with a white triangle. The hypervolumes $hv(\mathbf{x}\ , \mathbf{p}_1)$, $hv(\mathbf{x}\ , \mathbf{p}_2)$ and $hv(\mathbf{x}\ , \mathbf{p}_3)$ are shown in the figure.

3.1 Mathematical Model

The variables and parameters of the motor and gear system are described in Table 1. The values are based on the Maxon A-max 32 DC motor specifications.

At steady state, the power consumption of a geared DC motor is [21]:

$$P = V * I, \tag{10a}$$

$$\text{where:} \quad I = \frac{(J_L + J_g + n_2^2 J_l + n^2 J_m)\dot{\omega} + (\nu_g + n^2 \nu_m)\omega + \tau}{n k_t}, \tag{10b}$$

$$V = RI + n k_v \omega. \tag{10c}$$

When the load is accelerated from rest, it is possible to calculate the speed trajectory, for given trajectories of input voltage and speed reduction, by solving the following differential equation:

$$\dot{\omega}(t) = \frac{n(t) k_t V(t) - n(t)^2 k_v k_t \omega(t)}{\left(J_L + J_g + n_2(t)^2 J_l + n(t)^2 J_m\right) R} - \frac{\left(\nu_g + n(t)^2 \nu_m\right)\omega(t) + \tau}{J_L + J_g + n_2(t)^2 J_l + n(t)^2 J_m}, \tag{11}$$

where $\omega(0) = 0$ is used as a starting condition.

Table 1. Variables and parameters for the gearbox ARMOP

Type	Variable/ Parameter	Symbol	Units	Lower limit	Upper limit
x	no. of teeth	z_g		19	61
y	gear no.	i		1	5
	input voltage	V	V	0	12
p	load speed	ω	s^{-1}	16.5	295
	load torque	τ	$Nm{\cdot}10^{-3}$	10	260
	load inertia	J_L	$Kg{\cdot}m^2 \cdot 10^{-3}$	5	10
	velocity constant	k_v	$V{\cdot}s{\cdot}10^{-3}$	24.3	
	torque constant	k_t	$Nm{\cdot}A^{-1} \cdot 10^{-3}$	24.3	
	armature resistance	R	Ω	2.23	
	motor damping coefficient	ν_m	$Nm{\cdot}s{\cdot}10^{-6}$	3.16	
	motor inertia	J_m	$Kg{\cdot}m^2 \cdot 10^{-6}$	4.17	
	max nominal current	I_{nom}	A	1.8	
	gear damping coefficient	ν_g	$Nm{\cdot}s{\cdot}10^{-6}$	30	
	first reduction ratio	n_1		3.21	
	transmission no. of teeth	N_t		80	
	maximum acceleration time	$t_{\max}$	s	20	
derived	armature current	I	A	0	5.39
	second reduction ratio	n_2		0.311	3.21
	total reduction ratio	n		1	10.3
	layshaft inertia	J_l	$Kg{\cdot}m^2 \cdot 10^{-6}$	15.9	64.5
	load shaft inertia	J_g	$Kg{\cdot}m^2 \cdot 10^{-6}$	5.21	53.7

The total energy required for acceleration E can be derived from Eq. (10):

$$E = \int_0^T \frac{V(t)\Big(V(t) - n(t)k_v\omega(t)\Big)}{R} \mathrm{d}t, \tag{12}$$

where T is the time ω reaches the required speed.

3.2 Problem Formulation

According to the ARO methodology, introduced in Section 1.2, the problem variables are sorted in Table 1 to three types: $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{p}$. Most of the parameters in this problem are considered as having deterministic values, but some (ω, τ and J_L) possess uncertain values. The random variates of ω, τ and J_L are denoted as Ω, $\mathcal{T}$ and $\mathcal{J}_L$, respectively. The resulting variate of $\mathbf{p}$ is denoted as $\mathbf{P}$.

A gearbox is required to perform well both in steady state and during acceleration. These two requirements can be considered as different operation modes, with different configuration spaces. The configuration space in steady state includes the choice of the gear i and the input voltage V. During acceleration, it consists of trajectories in time of $i(t)$ and $V(t)$. Therefore, the search for the optimal configuration can be separated to $\mathbf{y}_{ss}^{\star}$ that minimises P, and to $\underline{\mathbf{y}_t^{\star}}$ that minimises E and T. Since the latter is a solution to a MOP, it is expected to be a set. The variates of $\mathbf{y}_{ss}^{\star}$ and $\underline{\mathbf{y}_t^{\star}}$ that correspond to the variate $\mathbf{P}$, are denoted as $\mathbf{Y}_{ss}^{\star}$ and $\underline{\mathbf{Y}_t^{\star}}$, respectively.

Following the above, the AROP is formulated:

$$\min_{\mathbf{x}\in\mathcal{X}} \left[P(\mathbf{x},\mathbf{Y}_{ss}^{\star},\mathbf{P}), \underline{E}\Big(\mathbf{x},\underline{\mathbf{Y}_t^{\star}},\mathbf{P}\Big), \underline{T}\Big(\mathbf{x},\underline{\mathbf{Y}_t^{\star}},\mathbf{P}\Big)\right], \tag{13a}$$

$$\text{where}: \quad \mathbf{Y}_{ss}^{\star} = \underset{\mathbf{y}\in\mathcal{Y}(\mathbf{x})}{\operatorname{argmin}}\, P(\mathbf{y},\mathbf{P}), \tag{13b}$$

$$\underline{\mathbf{Y}_t^{\star}} = \underset{\mathbf{y}\in\mathcal{Y}(\mathbf{x})}{\operatorname{argmin}}\, [E(\mathbf{y},\mathbf{P}), T(\mathbf{y},\mathbf{P})], \tag{13c}$$

$$\mathbf{x} = [z_i], \quad i = 1,\ldots,5, \tag{13d}$$

$$\mathbf{y} = [i, V], \tag{13e}$$

$$\mathbf{P} = [\Omega, \mathcal{T}, \mathcal{J}_L, k_v, k_t, R, \nu_m, I_{\mathrm{nom}}, \nu_g, n_1, N_t, J_m, J_l, J_G, t_{\max}], \tag{13f}$$

$$\text{s.t.}: \quad z_{g,i} + z_{l,i} = N_t, \quad i = 1,\ldots,5, \tag{13g}$$

$$I_{ss} \le I_{\mathrm{nom}}, \tag{13h}$$

$$T \le t_{\max}. \tag{13i}$$

The steady state current constraint is evaluated according to Eq. (10b), and the objectives according to Equations (10a), (11) and (12).

Since the ARMOP consists of separable configuration spaces, it can be decoupled into two subproblems, one that searches for $\mathbf{Y}_{ss}^{\star}$ and $P(\mathbf{x},\mathbf{Y}_{ss}^{\star},\mathbf{P})$, and another that searches for $\underline{\mathbf{Y}_t^{\star}}$ and $\left[\underline{E}\Big(\mathbf{x},\underline{\mathbf{Y}_t^{\star}},\mathbf{P}\Big), \underline{T}\Big(\mathbf{x},\underline{\mathbf{Y}_t^{\star}},\mathbf{P}\Big)\right]$. The former problem is a single-objective AROP, and the latter is an ARMOP. Using robustness indicators, Eq. (13a) can be converted to the following bi-objective problem that simultaneously minimises the steady-state AROP and the transient ARMOP:

$$\min_{\mathbf{x}\in\mathcal{X}} \left[I\Big[P(\mathbf{x},\mathbf{Y}_{ss}^{\star},\mathbf{P})\Big], I_{hv}\Big[\underline{E}\Big(\mathbf{x},\underline{\mathbf{Y}_t^{\star}},\mathbf{P}\Big), \underline{T}\Big(\mathbf{x},\underline{\mathbf{Y}_t^{\star}},\mathbf{P}\Big)\Big]\right]. \tag{14}$$

4 Optimiser Design

The problem was solved by a bi-level EMOA whose structure is described in Algorithm 1.

First, the uncertain domain is sampled N_p times. These samples serve as the same representation of uncertainties to evaluate all solutions.

Next, Eq. (13b) is solved for the entire design space, and $\mathbf{Y}_{ss}^{\star}$ and $P(\mathbf{x},\mathbf{Y}_{ss}^{\star},\mathbf{P})$ are stored in an archive for every feasible solution. It is possible to find the optimal steady-state configuration of every solution for all sampled load scenarios because the design space is discrete and the objective and constraints are simple expressions. The search space consists of 962,598 different combinations of gears (choice of 5 gears from 43 possibilities). The constraints and objective functions depend on the number of teeth z, so they only have to be evaluated 43 times for each of the sampled scenarios. A feasible solution is a gearbox that has at least one gear that does not violate the constraints for each of the scenarios (i.e., $I \leq I_{\mathrm{nom}}$ and $V \leq V_{\mathrm{max}}$).

Next, a multi-objective search is conducted amongst the feasible solutions to solve Eq. (14). The solutions to Eq. (13c) for every sampled scenario are obtained by the evolutionary algorithm described in Section 4.1. The solutions to Eq. (13b) are already stored in an archive.

Algorithm 1.. Pseudo algorithm for solving the ARMOP

```
sample the uncertain domain
evaluate all possible solutions for steady state (s.s)
initialise nadir and ideal points for transient objectives (limits)
generate an initial population
while stopping criterion not satisfied do
   for every scenario do
      for every new solution do
         optimise for time–energy and store PF
      end for
   end for
   if limits have changed then
      update limits
      calculate HV of entire population
   else
      calculate HV of new feasible solutions
   end if
   assign scalar indicator values for s.s and transient
   evolve new population (selection, cross-over and mutation)
   re-mutate solutions that were already evaluated / infeasible for s.s
end while
```

4.1 EMOA for Identifying Optimal Gearing Sequences

For every load scenario, a multi-objective optimisation is conducted for each candidate solution to identify the optimal shift sequence that minimises energy and acceleration time. Early experiments revealed that maximum voltage results in better values for both objectives, regardless of the candidate solution or the load scenario. Therefore, the input voltage was considered as constant V_{max}, and the only search variable is $i(t)$, the selected gear at time t. A certain trajectory $i(t)$ results in a gearing ratio trajectory $n(t)$ that depends on the gearbox $\mathbf{x}$ that is being evaluated.

The trajectory $i(t)$ is coded as a vector of time intervals $\mathbf{dt} = [\mathrm{d}t_1, \ldots, \mathrm{d}t_N]$ defining the duration of each gear in the sequence from first gear to the N^{th}, with N being the optimal gear at steady state for the load scenario under consideration. The sum of all time intervals is equal to $t_{\max}$, and this relation is enforced whenever a new solution is created by setting:

$$\mathbf{dt} \leftarrow \frac{\mathbf{dt}}{\|\mathbf{dt}\|_1} t_{\max}. \tag{15}$$

Plugging $n(t)$ into Eq. (11) results in a trajectory $\omega(t)$, which can be used to calculate E, T or whether the gearbox failed to reach the desired speed before $t_{\max}$. A multi-objective evolutionary algorithm was used to estimate:

$$\underline{\mathbf{y}_t^\star} = \underset{n(t)}{\operatorname{argmin}} \left[E(\mathbf{x}, n(t), \mathbf{p}), T(\mathbf{x}, n(t), \mathbf{p})\right], \tag{16}$$

where both $\mathbf{x}$ and $\mathbf{p}$ are fixed during the entire optimisation run.

Solving the differential equation (11) repeatedly to obtain $\underline{\mathbf{y}_t^\star}$ is the most expensive part of the algorithm in terms of computational resources. Therefore, all of the solutions to (16) are stored in an archive to avoid repeated computations.

4.2 Calculating the Set-Based Robustness Indicator

The ARMOP's indicator I_{hv} uses a dynamic reference point. At every generation, after the approximated Pareto frontiers $\underline{\mathbf{F}}(\mathbf{x}, \underline{\mathbf{Y}^\star}, \mathbf{P})$ are identified for all evaluated solutions, the ideal and worst objective vectors are re-evaluated to include the objective vectors of the new solutions. If neither the ideal nor the worst objective vectors have changed, I_{hv} is calculated only for the recently evaluated solutions according to the procedure described in Section 2.1. Otherwise, the indicator values of the entire current population are recalculated as well, in order to allow for fair comparisons between new and old candidate solutions. No preferences were considered in this case study, hence, the objectives were normalised by setting $\mathbf{f}^w$ to one.

5 Simulation Results

5.1 Parameter Setting

The ARMOP described in Section 3 was solved with the proposed evolutionary algorithm. Two robustness criteria were considered: I^w considers the worst case scenario, meaning the upper limits of the uncertain load parameters, as given in Table 1. I^m considers the mean value over a set of sampled load scenarios. For both cases the same criterion was used for the steady state and transient indicators of Eq. 14, i.e., either I^w and I^w_{hv} or I^m and I^m_{hv}.

A standard elitist MOEA [22] with a fixed number of generations was used for both stages of the problem (referred to as outer and inner).

Parameter setting of the outer algorithm: population size $N = 100$, 50 generations, integer coded, One-point crossover with crossover rate $p_c = 1$, polynomial mutation with mutation rate $p_m = 1/n_x = 0.2$ and distribution index $\eta_m = 20$.
Parameter setting of the inner algorithm: population size $N = 50$, 30 generations, real coded, SBX crossover with crossover rate $p_c = 1$ and distribution index $\eta_c = 15$, polynomial mutation with mutation rate $p_m = 1/n_y = 0.2$ and distribution index $\eta_m = 20$.

Both stages used sequential tournament selection, considering constraint violation, non-dominance rank and niche count, and had an elite population size of $N_E = 0.4N$. The uncertain load domain was sampled 25 times using Latin hypercube sampling.

5.2 Results

The approximated Pareto frontiers for both worst-case and mean-value criteria are depicted in Fig. 3. For the worst-case criterion, the PS consists of only two, almost identical, solutions. In a close-up view on the approximated PF for mean performance, the extreme solutions are marked as A and B. Mean performances of the approximated Pareto set for the worst-case problem are also shown.

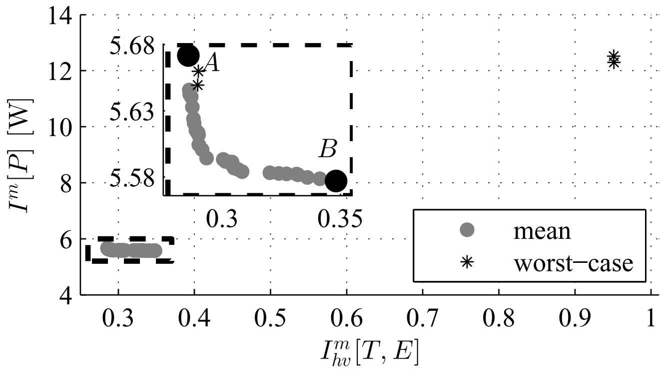


Fig. 3. Approximated Pareto frontiers for the worst-case and mean-value criteria. A close-up of the robust mean Pareto front is shown with the extreme solutions marked as A and B, and the mean performance of the approximated set according to I^w.

Details on the solutions for both robustness criteria are summarised in Table 2. Note the similarity in both design and objective spaces between the two solutions of the worst-case problem, and the difference between Solutions A and B. Also note that the best solutions found for a certain robustness criterion, are dominated for another. Solution B performs well in most steady state scenarios, since it has a large variety of high gears (small reduction ratio), but its ability to efficiently accelerate the load is limited from the same reason. Solution B becomes infeasible when the worst-case is considered. This was not detected

Table 2. Optimisation Results

Goal	**Solution**	**Reduction Ratios** 1^{st}	2^{nd}	3^{rd}	4^{th}	5^{th}	$I^m\ [P]$	$I^m_{hv}\ [T,E]$	$I^w\ [P]$	$I^w_{hv}\ [T,E]$
I^m	A	9.02	4.34	2.62	1.93	1.30	5.672	0.2857	13.10	0.9631
	B	2.76	2.25	1.92	1.73	1.64	5.577	0.3481	infeasible	
I^w		7.06	3.38	2.14	1.55	1.14	5.649	0.2896	12.30	0.9511
		7.49	3.38	2.03	1.46	1.14	5.660	0.2899	12.52	0.9510

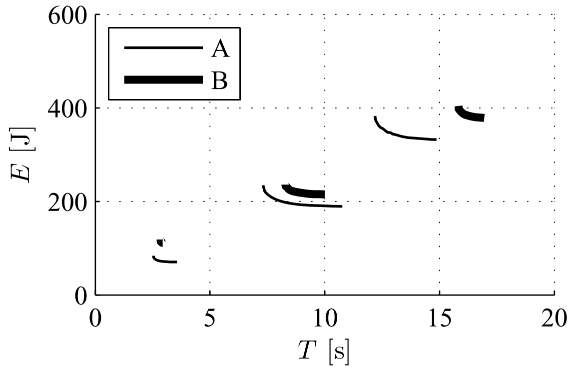


Fig. 4. Approximated Pareto frontiers $\underline{\mathbf{F}}(\mathbf{x}, \underline{\mathbf{Y}}^{\star}, \mathbf{P})$ of two solutions (A and B) for three scenarios (of 25). Solution A dominates Solution B in all evaluated scenarios.

while optimising for the mean value since the worst-case scenario was not sampled. This result highlights the impact of the choice of robustness criterion, and the challenge in optimising for the worst-case (see [3]).

The dynamic performances of Solutions A and B for three load scenarios are depicted in Fig 4. Solution A's superiority for both dynamic objectives is well captured by the I_{hv} indicator values.

6 Discussion and Future Work

This study introduced a new optimisation problem, the Active Robust Multiobjective Optimisation Problem. It enables a designer to examine the effectiveness of design adaptability to improve performance in an uncertain environment. The ARMOP introduces several challenges, some of which were addressed in this study, and others which need to be further explored.

The approach taken in this study to solve an ARMOP is to use a scalarising function to represent the variate of Pareto frontiers of every candidate solution. This approach was found useful for the gearbox case study – solutions with better Pareto frontiers were assigned with a better indicator value. However, whenever a set is represented by a scalar value, some of its information must be lost. As a result, setting a robustness criterion for the utility indicator value does not automatically imply that the individual objectives will also be robust.

Being a bi-level optimisation problem, an AROP requires many function evaluations. An ARMOP is even harder to solve, because the inner problem is a MOP. The strategy for obtaining robust solutions taken in this study was based on Monte Carlo simulations to represent the uncertain variables. This representation requires a large set of samples to adequately capture the nature of the uncertainties involve, and to gain confidence in the robustness of the obtained solutions. Due to limited computational resources, the approach was demonstrated in this study with a small set of sampled scenarios, only to provide a proof of concept. Even for these minimal optimiser settings, almost 70 million function evaluations were conducted. It took approximately three days to compute on a 3.40GHz Intel® Core™ i7-4930K CPU, running Matlab® on 12 cores.

Future research should explore other representations of the uncertainties that involve more efficient sampling approaches and use of a-priori knowledge; as well as optimisation algorithms for expensive function evaluations. Alternative scalarising functions, and their effects on the optimisation results, should also be explored.

Acknowledgments. This research was supported by a Marie Curie International Research Staff Exchange Scheme Fellowship within the 7th European Community Framework Programme. The first author acknowledges the support of the Anglo-Israel Association.

References

1. Salomon, S., Avigad, G., Fleming, P.J., Purshouse, R.C.: Active Robust Optimization - Enhancing Robustness to Uncertain Environments. IEEE Transactions on Cybernetics **44**(11), 2221–2231 (2014)
2. Beyer, H.G., Sendhoff, B.: Robust Optimization - A Comprehensive Survey. Computer Methods in Applied Mechanics and Engineering **196**(33–34), 3190–3218 (2007)
3. Branke, J., Rosenbusch, J.: New approaches to coevolutionary worst-case optimization. In: Rudolph, G., Jansen, T., Lucas, S., Poloni, C., Beume, N. (eds.) PPSN X. LNCS, vol. 5199, pp. 144–153. Springer, Heidelberg (2008)
4. Avigad, G., Coello, C.A.: Highly Reliable Optimal Solutions to Multi-Objective Problems and Their Evolution by Means of Worst-Case Analysis. Engineering Optimization **42**(12), 1095–1117 (2010)
5. Alicino, S., Vasile, M.: An evolutionary approach to the solution of multi-objective min-max problems in evidence-based robust optimization. In: 2014 IEEE Congress on Evolutionary Computation (CEC), pp. 1179–1186 (2014)
6. Teich, J.: Pareto-front exploration with uncertain objectives. In: Zitzler, E., Deb, K., Thiele, L., Coello Coello, C.A., Corne, D. (eds.) EMO 2001. LNCS, vol. 1993, pp. 314–328. Springer, Heidelberg (2001)
7. Hughes, E.J.: Evolutionary multi-objective ranking with uncertainty and noise. In: Zitzler, E., Thiele, L., Deb, K., Coello Coello, C.A., Corne, D. (eds.) EMO 2001. LNCS, vol. 1993, pp. 329–343. Springer, Heidelberg (2001)
8. Deb, K., Gupta, H.: Introducing Robustness in Multi-Objective Optimization. Evolutionary Computation **14**(4), 463–494 (2006)

9. Saha, A., Ray, T.: Practical Robust Design Optimization Using Evolutionary Algorithms. Journal of Mechanical Design **133**(10), 101012 (2011)
10. Beyer, H.G., Sendhoff, B.: Functions with noise-induced multimodality: a test for evolutionary robust Optimization-properties and performance analysis. IEEE Transactions on Evolutionary Computation **10**(5), 507–526 (2006)
11. Fieldsend, J.E., Everson, R.M.: Multi-objective optimisation in the presence of uncertainty. In: The 2005 IEEE Congress on Evolutionary Computation, vol. 1, pp. 243–250 (2005)
12. Bui, L.T., Abbass, H.A., Essam, D.: Fitness inheritance for noisy evolutionary multi-objective optimization. In: Proceedings of the 7th Annual Conference on Genetic and Evolutionary Computation, GECCO 2005, pp. 779–785, New York. ACM (2005)
13. Goh, C.K., Tan, K.C.: An Investigation on Noisy Environments in Evolutionary Multiobjective Optimization. IEEE Transactions on Evolutionary Computation **11**(3), 354–381 (2007)
14. Knowles, J., Corne, D., Reynolds, A.: Noisy multiobjective optimization on a budget of 250 evaluations. In: Ehrgott, M., Fonseca, C.M., Gandibleux, X., Hao, J.-K., Sevaux, M. (eds.) EMO 2009. LNCS, vol. 5467, pp. 36–50. Springer, Heidelberg (2009)
15. Fieldsend, J.E., Everson, R.M.: The Rolling Tide Evolutionary Algorithm: A Multi-Objective Optimiser for Noisy Optimisation Problems. IEEE Transactions on Evolutionary Computation **PP**(99), 1 (2014)
16. Paenke, I., Branke, J., Jin, Y.: Efficient Search for Robust Solutions by Means of Evolutionary Algorithms and Fitness Approximation. IEEE Transactions on Evolutionary Computation **10**(4), 405–420 (2006)
17. Cruz, C., González, J.R., Pelta, D.A.: Optimization in Dynamic Environments: A Survey on Problems, Methods and Measures. Soft Computing **15**(7), 1427–1448 (2011)
18. Fleming, P.J., Purshouse, R.C., Lygoe, R.J.: Many-objective optimization: an engineering design perspective. In: Coello Coello, C.A., Hernández Aguirre, A., Zitzler, E. (eds.) EMO 2005. LNCS, vol. 3410, pp. 14–32. Springer, Heidelberg (2005)
19. Zitzler, E.: Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications. Phd dissertation, Swiss Federal Institute of Technology Zurich (1999)
20. Knowles, J., Corne, D.: On metrics for comparing nondominated sets. In: Proceedings of the 2002 Congress on Evolutionary Computation, CEC 2002, pp. 711–716. IEEE (2002)
21. Krishnan, R.: Electric Motor Drives - Modeling, Analysis, And Control. Prentice Hall (2001)
22. Salomon, S., Avigad, G., Goldvard, A., Schütze, O.: PSA – a new scalable space partition based selection algorithm for MOEAs. In: Schütze, O., Coello Coello, C.A., Tantar, A.-A., Tantar, E., Bouvry, P., Del Moral, P., Legrand, P. (eds.) EVOLVE - A Bridge between Probability. AISC, vol. 175, pp. 137–151. Springer, Heidelberg (2012)