Guidance framework and software for understanding and achieving system robustness

McPhail, C.; Maier, H. R.; Westra, S.; van der Linden, L.; Kwakkel, J. H.

DOI
10.1016/j.envsoft.2021.105059

Publication date
2021

Document Version
Final published version

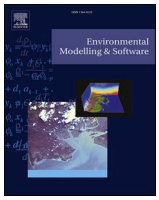
Published in
Environmental Modelling and Software

Environmental Modelling and Software 142 (2021) 105059

Contents lists available at ScienceDirect

Environmental Modelling and Software

journal homepage: www.elsevier.com/locate/envsoft



Guidance framework and software for understanding and achieving system robustness

C. McPhail [a,*], H.R. Maier [a], S. Westra [a], L. van der Linden [b], J.H. Kwakkel [c]

[a] University of Adelaide, Australia
[b] SA Water Corporation, Australia
[c] Delft University of Technology, the Netherlands

ARTICLE INFO

Keywords:
Deep uncertainty
Decision making under uncertainty
Robustness
Scenarios

ABSTRACT

To aid decision making about environmental systems under deep uncertainty, robustness metrics are commonly used to represent system performance over a number of scenarios. However, there are many robustness metrics and many ways of generating scenarios, making it difficult to know which to choose in order to quantify system robustness and to make robust decisions. To address this shortcoming, we introduce a generic guidance framework to assist with the identification of the most robust decision alternatives, as well as the RAPID (Robustness Analysis Producing Intelligent Decisions) software package which is a consistent and easy-to-use implementation of the framework. We illustrate the framework and software package on a hypothetical lake pollution problem, known as The Lake Problem, showing how the framework and software package apply to several situations where decision-makers may or may not know which scenarios or robustness metrics to use.

1. Introduction

The long-term planning of environmental systems presents major challenges, as it requires decisions to be made despite significant uncertainty about the future state of the world. Frequently, decision-makers are operating at the level of deep uncertainty, which refers to when deterministic and probabilistic approaches are insufficient for representing future states, and the consideration of multiple plausible futures (scenarios) is required (Bradfield et al., 2005; Herman et al., 2014; Kwakkel et al., 2010; Kwakkel and Haasnoot, 2019; Lempert, 2003; Little et al., 2018; Maier et al., 2016; Schwarz, 1991; van der Heijden, 1996; Varum and Melo, 2010; Walker et al., 2013; Wright and Cairns, 2011). Implicit in the deep uncertainty paradigm is that probabilities cannot be placed on these scenarios, and therefore traditional risk-based performance metrics such as reliability, vulnerability, resilience, or expected value cannot be used to quantify the overall level of system performance across all scenarios (Maier et al., 2016). Rather, deep uncertainty requires robustness metrics to be used, which aim to quantify the (relative) level or variation of system performance across all or targeted scenarios (Bartholomew and Kwakkel, 2020; Giudici et al., 2020; Herman et al., 2015; Kwakkel and Haasnoot, 2019; Lempert, 2003; Maier et al., 2016; McPhail et al., 2018). As with traditional performance metrics in deterministic and probabilistic paradigms, decision-makers aim to choose a solution that has maximal performance (robustness) or a solution that has an appropriate tradeoff between performance metrics (e.g. robustness vs. cost).

There is a multitude of approaches to quantify system robustness, generally by treating the scenarios as a distribution and making implicit probabilistic assumptions, including: (i) expected value metrics (Wald, 1951), which indicate an expected level of performance across a range of scenarios; (ii) metrics of higher-order moments, such as variance and skew (e.g. Kwakkel et al. (2016a)), which provide information on how the expected level of performance varies across multiple scenarios; (iii) regret-based metrics (Savage, 1951), where the regret of a decision alternative is defined as the difference between the performance of the selected option for a particular plausible condition and the performance of the best possible option for that condition; and (iv) satisficing metrics (Simon, 1956), which identify the range of scenarios that have acceptable performance relative to a threshold. A common conclusion from recent research is that different robustness metrics can sometimes lead to decision alternatives being ranked differently, making it difficult to determine which decision alternatives are most robust (Borgomeo et al., 2018; Drouet et al., 2015; Giuliani and Castelletti, 2016; Hall et al., 2012; Herman et al., 2015; Kwakkel et al., 2016a; Lempert and Collins,

* Corresponding author.
E-mail address: cameron.mcphail@adelaide.edu.au (C. McPhail).

https://doi.org/10.1016/j.envsoft.2021.105059
Accepted 15 April 2021
Available online 30 April 2021

2007; McPhail et al., 2018; Roach et al., 2016). For example, a case study by Kwakkel et al. (2016a) on the transition of the European energy system towards a more sustainable future concluded that "there is no clearly superior single robustness metric. Case specific consideration and system characteristics affect the merits of the various robustness measures. This implies that an analyst has to choose carefully which robustness measure is being used and assess its appropriateness."

Given that robustness metrics are calculated over a set of scenarios, the choice of scenarios that are used in this calculation can also have an impact on the robustness value obtained (in addition to the choice of robustness metric) (McPhail et al, 2018, 2020). A common categorization of scenarios is given by Börjeson et al. (2006), including the following three types:

- Predictive scenarios – where the aim is to determine "what will happen?" For example, the future state of the world could be based on some future trajectory or change in trajectory due to some event;
- Explorative scenarios – where the aim is to determine "what could happen?" Generally, this is done by framing the future in terms of the uncertainties that have the largest effects on system performance, but the future can also be unframed (Maier et al., 2016); and
- Normative scenarios – where the aim is to determine "how can a specific future be realized?" This is generally focused on interesting future outcomes or failure points for decision alternatives.

Each of these types of scenarios can be created in different ways. For example, a set of scenarios for a particular problem could be created in a largely qualitative manner through a participatory process with stakeholders with the aim of producing generalizable scenarios (e.g. Wada et al. (2019)), while a different set of scenarios for the same problem could be created through a largely quantitative process by varying the inputs to the system model of interest (e.g. using an approach such as Latin hypercube sampling (LHS)) (Culley et al, 2016, 2019; Hadka et al., 2015; Hall et al., 2012; Herman et al., 2015; Kasprzyk et al., 2013; Kwakkel, 2017; Kwakkel et al, 2015, 2016b; McPhail et al., 2018; Quinn et al, 2017, 2018; Singh et al., 2015; Trindade et al., 2017; Watson and Kasprzyk, 2017; Weaver et al., 2013; Zeff et al., 2014). Each of these approaches can lead to vastly different scenarios being produced (Shepherd et al., 2018); for example, a participatory approach will generally result in a small number of scenarios in targeted regions of the uncertain inputs space, while quantitative approaches (e.g. LHS of scenarios) would lead to a large number of scenarios with even coverage of the space. Recent studies have shown that, as is the case for the use of different robustness metrics, the use of different sets of scenarios can also result in different robustness values of decision alternatives (McPhail et al., 2020; Quinn et al., 2020; Reis and Shortridge, 2020), adding further uncertainty to the way the robustness of decision alternatives is quantified.

In order to assist analysts and decision makers in performing appropriate robustness analyses, McPhail et al. (2018) and McPhail et al. (2020) developed generalizable, quantitative approaches to assessing the sensitivity of the absolute and relative robustness of decision alternatives (e.g. designs, policies) to the selection of robustness metrics and scenarios, respectively. However, there is still a lack of a holistic procedure that provides guidance to analysts on the best way to identify which of the available decision alternatives is likely to be the most robust. Consequently, the overarching aim of this paper is to develop a generic guidance framework to help identify the robustness of decision alternatives. This paper also introduces the RAPID (Robustness Analysis Producing Intelligent Decisions) software package, implementing the proposed guidance framework in a consistent and user-friendly manner, that enables the most robust decision alternatives to be identified for a given problem. The software package complements existing software packages in this robust decision making space including the Exploratory Modelling (EM) Workbench (Kwakkel, 2017) and Rhodium (Hadjimichael et al., 2020). We illustrate the guidance framework and software package on a hypothetical lake pollution problem, known as The Lake Problem, as it is a simple and well-represented case study in the literature (Carpenter et al., 1999; Eker and Kwakkel, 2018; Hadka et al., 2015; Kwakkel, 2017; Lempert and Collins, 2007; Quinn et al., 2017; Singh et al., 2015; Ward et al., 2015).

Consequently, the specific objectives of this paper are to:

1. develop a generic guidance framework to help identify the most robust decision alternatives for a given decision context;
2. describe a software package that enables the guidance framework to be implemented in a consistent and user-friendly manner; and
3. illustrate the application of the framework and software package on the Lake Problem.

The remainder of this paper is organized as follows: Section 2 introduces the guidance framework for analyzing the robustness of a set of decision alternatives, including how to create a custom robustness metric and how to assess the impact of the selection of scenarios and choice of robustness metric; Section 3 introduces a software package that can be used to implement this guidance and quantitatively and visually assess the impact of the choice of scenarios and robustness metric on the robustness values and rankings of decision alternatives; Section 4 introduces the Lake Problem and provides a simple illustration of how the guidance and software package can be applied to an environmental model; and conclusions are presented in Section 5.

2. Guidance framework for identifying the most robust decision alternatives

At the heart of the proposed framework for assisting with the identification of the robustness of decision alternatives is the calculation of different robustness metrics. The calculation of these metrics requires scenarios, decision alternatives (i.e. plans, policies, solutions), and one or more quantitative metrics (e.g. reliability or vulnerability), which can be used to determine the level of performance of each decision alternative in each individual scenario (Herman et al., 2015; McPhail et al., 2018). Fig. 1 shows the processes through which these three inputs are used to calculate the robustness of each decision alternative (i.e. the system performance across all scenarios). Calculation of robustness consists of two main steps: (1) the use of a system model to calculate each decision alternative's performance in each scenario, followed by (2) the combination of these performance values in order to calculate a single robustness value. While these steps are identical for each robustness metric, different robustness metrics require the selection of different options at each of one of three transformations: (1) performance value transformation; (2) scenario subset selection; and (3) aggregation of performance values (McPhail et al., 2018) (Fig. 1). At the first transformation, the options are whether to use the raw values of system performance or whether to alter these values using regret or satisficing transforms. At the second transformation, the choice is which subset of the available scenarios to use in the calculation of the robustness metric. At the third transformation, the options are whether to combine the transformed performance values over the selected scenarios using a measure of the level of performance, such as the mean, or a measure of variability in performance, such as the standard deviation.

The proposed guidance framework for assisting with the identification of the most robust decision alternatives is given in Fig. 2. The framework is designed to be as generic as possible, catering to the level of knowledge of the decision-makers, including the following situations:

- where the most appropriate robustness metric for a particular problem is already fixed or pre-selected (Section 2.1),
- where a range of robustness metrics are to be considered (e.g. where decision makers are either interested in understanding multiple aspects of robustness via different robustness metrics, or cannot decide

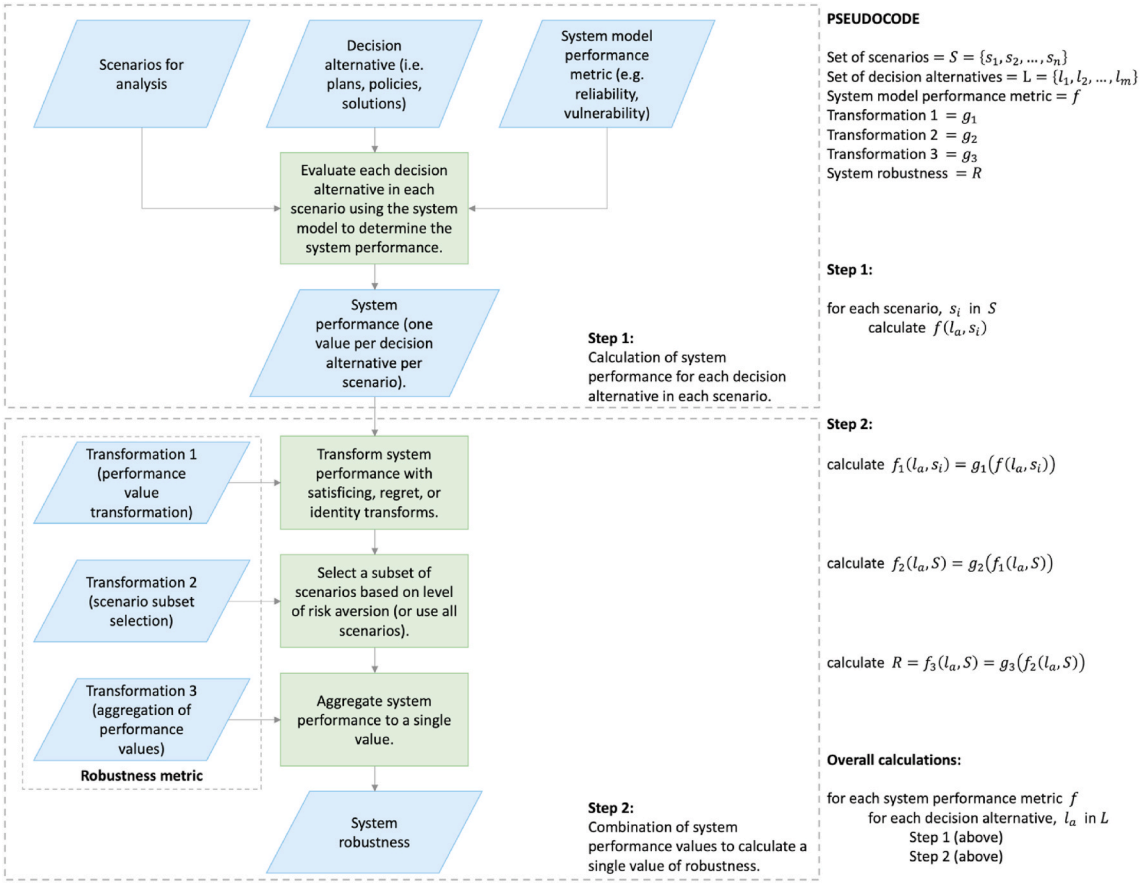


Fig. 1. Inputs and processes for calculating system performance and robustness. Transformations 1–3 are the components of the robustness metric, explained further in the main text.

on which robustness metric is most appropriate from some set of robustness metrics) (Section 2.2), or

- where the most appropriate robustness metric is yet to be determined based on the different attributes of the decision context (i.e. the properties of the problem, such as system thresholds) and the preferences of the decision-maker(s) (e.g. preferred levels of risk aversion) (Section 2.3).

The framework also caters to situations where the scenarios under which system performance is to be calculated are already selected, and situations where the influence of different sets of scenarios on the robustness of decision alternatives is to be considered (e.g. situations where one wishes to know the sensitivity of a particular decision outcome to the selection of scenarios for analysis). It should be noted that the proposed framework assumes that the decision alternatives to be considered have already been selected and that the relevant performance metrics for these decision alternatives have been calculated.

2.1. Robustness metric is already pre-selected

The process of identifying the decision alternative that has the highest relative robustness commences with the candidate set of decision alternatives for which the relative robustness is to be calculated. The first decision point in this process is whether the robustness metric to be used in the assessment has been pre-selected (Fig. 2, Box 2). If an appropriate metric has already been selected, the next decision point in determining the robustness of decision alternatives is whether the set of scenarios to be used to determine the performance of the decision alternatives under consideration is fixed/pre-selected or not (Fig. 2, Box 6). If the set of scenarios is pre-selected, the robustness of each decision alternative can be calculated by combining its performance over the selected scenarios with the aid of the selected robustness metric. Then the alternative with the highest robustness value can be selected (Fig. 2, Boxes 10 and 15).

If it is not clear which scenarios should be used for the robustness calculation, the sensitivity of the relative robustness values of the different decision alternatives can be determined for different user-defined scenario sets, using the approach of McPhail et al. (2020) (Fig. 2, Box 13). Visualizations of the relative ranking of the decision alternatives can be used to determine (using human judgement) whether the choice of candidate scenario set matters (Fig. 2, Box 14), as illustrated in McPhail et al. (2020). If the choice of candidate scenarios does not matter because the visualizations indicate that the decision alternatives are ranked similarly regardless of which scenarios are selected, then the decision alternative that is considered most robust can be easily selected (Fig. 2, Box 15). However, if the choice of scenarios does affect the relative robustness of the decision alternatives of interest, then, depending on the degree of sensitivity of the relative robustness of the different decision alternatives to the selected scenario sets, some degree of judgement will be required to determine which decision alternative is considered most robust or which decision alternatives have an acceptable level of robustness (Fig. 2, Box 16), or it might be concluded that it is not possible to identify which decision alternative is most robust. Note that in the situation where a robustness metric is known or pre-selected, it may still be useful to consider the pathways through Fig. 2 where the robustness metric is not known. This would provide extra information about the system and the impact of the selected robustness metric on the robustness and rankings, as described below.

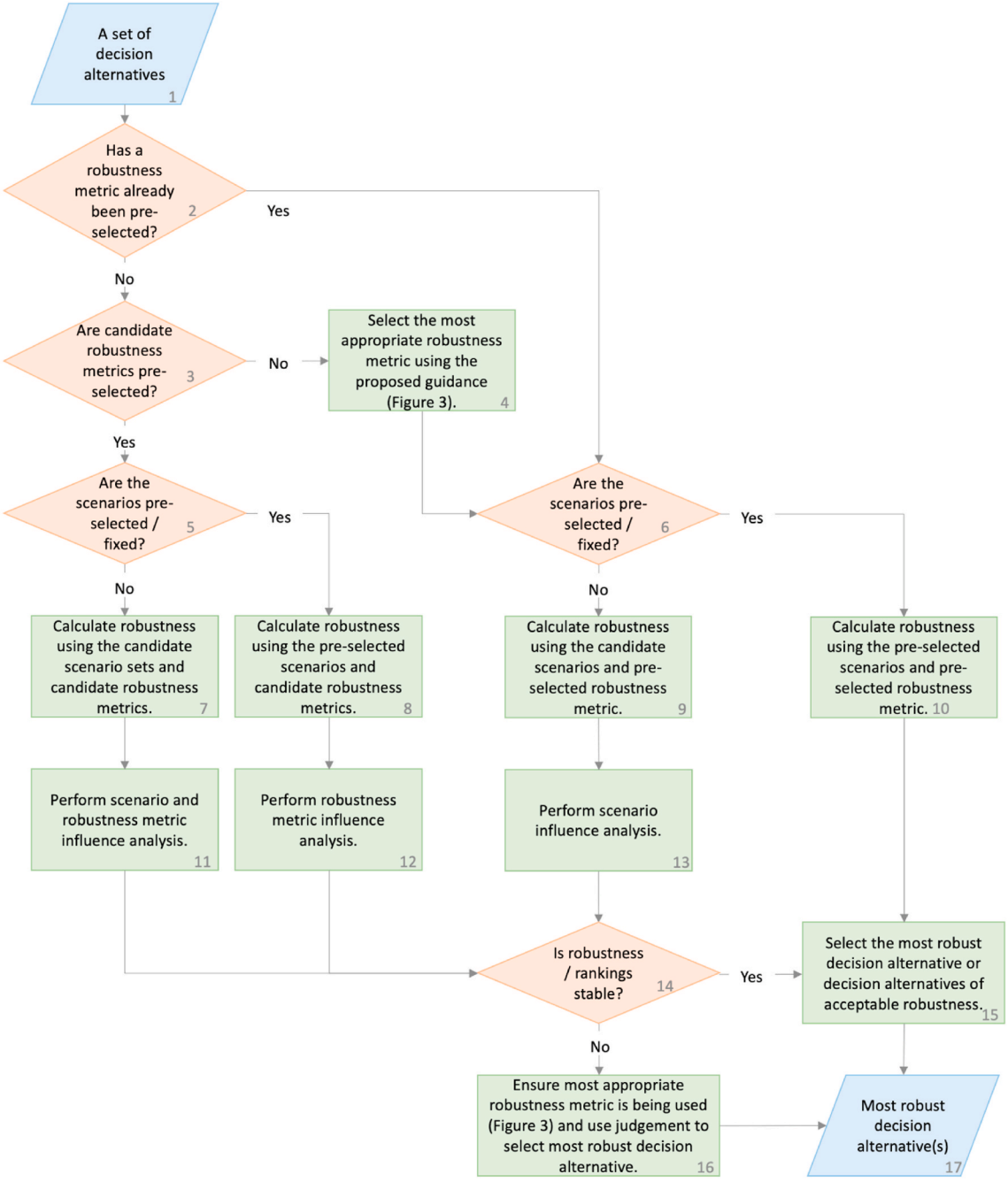


Fig. 2. Proposed generic guidance framework for assisting with the identification of the most robust decision alternative.

2.2. *There is a range of robustness metrics under consideration*

If the robustness metric to be used is not known or pre-selected, the key decision point is whether there is a known or pre-selected *set* of alternative robustness metrics to be considered in the analysis (Fig. 2, Box 3). If there is a pre-selected set of robustness metrics for consideration, the next decision point is whether there is a fixed/pre-selected set of scenarios or not (Fig. 2, Box 5). If there is a pre-selected set of scenarios, the stability of the relative robustness of the decision alternatives under consideration can be calculated for the selected robustness metrics over the selected scenarios, using the approach of McPhail et al. (2018) (Fig. 2, Box 12). Visualizations of the relative ranking of the decision alternatives can be used to determine whether the choice of candidate robustness metrics matters (Fig. 2, Box 14), as illustrated in McPhail et al. (2020) and further discussed in Sections 3 and 4. If the choice of robustness metrics does not matter because the visualizations indicate that the decision alternatives are ranked similarly regardless of which robustness metric is used, then the decision alternative that is considered most robust can be selected easily (Fig. 2, Box 15). However, if the robustness metric does affect the relative robustness of the decision alternatives of interest, then, depending on the degree to which this has an effect, some degree of judgement will be required to determine which alternative is most robust (or which decision alternatives have an acceptable level of robustness), and it is recommended that the process for identifying the most appropriate robustness metric for the decision context under consideration introduced in Fig. 3 and discussed below be applied and that the analysis be repeated for the selected robustness metric (Fig. 2, Box 16).

If the set of scenarios to be used are not fixed or pre-selected, the sensitivity of the relative robustness values of the different decision

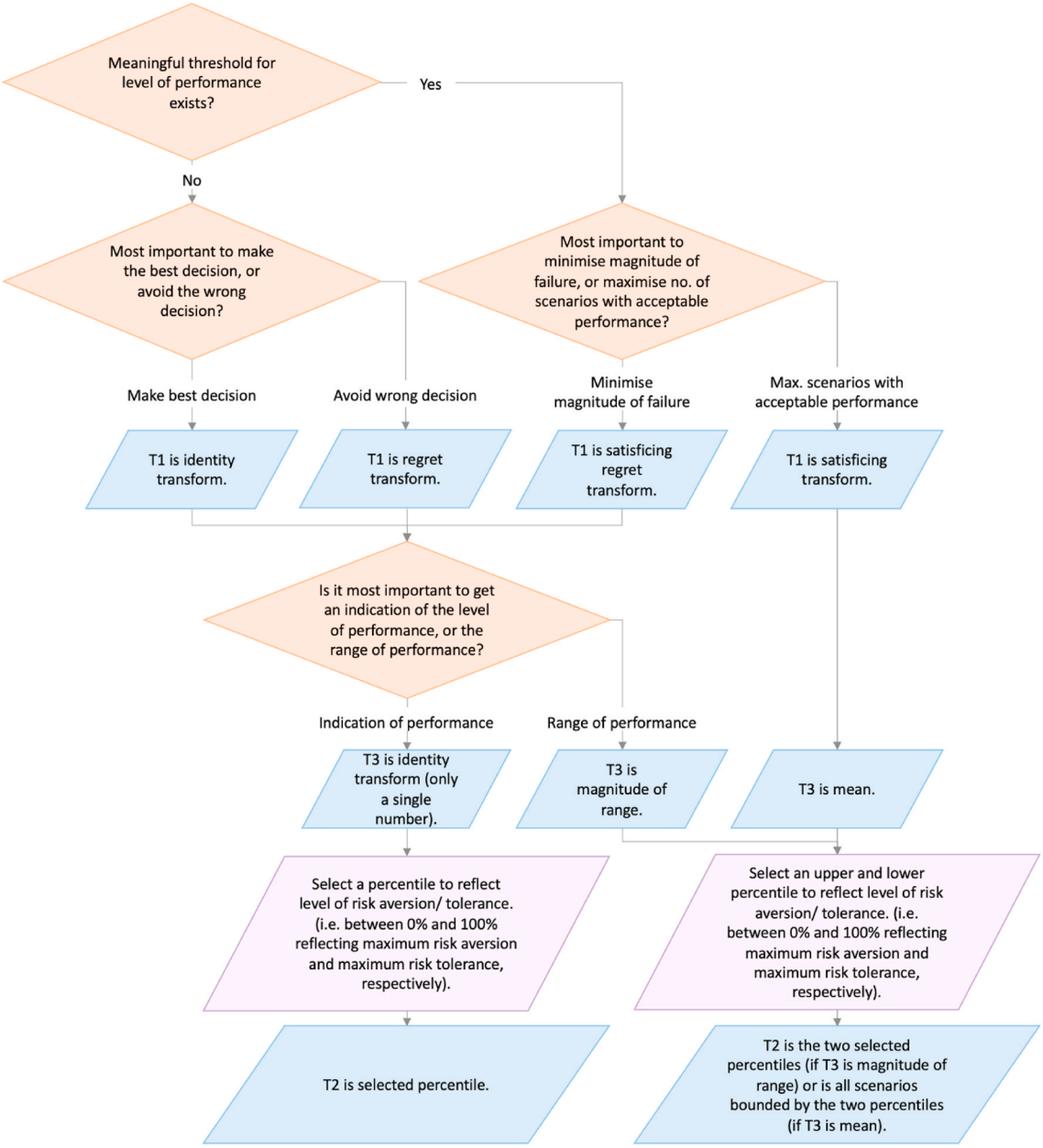


Fig. 3. Guidance for the creation of a robustness metric for each performance metric according to the problem being analyzed and the preferences of the decision-maker. (Note that the equations assume the objective here is to maximize system performance).

alternatives to the different user-defined scenario sets and robustness values can be determined using the approach of McPhail et al. (2020) (Fig. 2, Box 11). Again, the visualizations (as illustrated in McPhail et al. (2020) and further discussed in Sections 3 and 4) allow the decision-maker to see whether the candidate sets of scenarios and candidate robustness metrics have a significant effect on relative robustness (Fig. 2, Box 14). If the selection of scenarios and robustness metrics has an insignificant effect on the rankings, the most robust decision alternative can be selected easily (Fig. 2, Box 15). However, if scenario and robustness metric selection have an effect on relative robustness, then, depending on the degree to which this is the case, some degree of judgement will be required to determine which decision alternative is most robust or which decision alternatives have acceptable levels of robustness, and it is recommended that the most appropriate robustness metric is used to help determine this (Fig. 2, Box 16).

2.3. *The robustness metric(s) are yet to be determined*

If the set of alternative robustness metrics to be considered in the analysis is yet to be determined (Fig. 2, Box 3), the most appropriate robustness metric to be used for each individual performance metric can be determined by selecting the most appropriate options at each of the three transformations in Fig. 1 with the aid of the guidance in Fig. 3 and the corresponding equations in Table 1 (Fig. 2, Box 4). It should be noted that this guidance and corresponding equations can be used to derive many of the established robustness metrics, but other pathways through the guidance frameworks may lead to novel robustness metrics. The first step in this process is to determine whether there is a meaningful performance threshold in the problem under consideration. For example, in a water supply system, the sustainable yield must be greater than demand and thus the required demand becomes a constraint for the problem. In this case, the question then becomes whether solutions can be assessed using a pass or fail criterion, or whether the magnitude of the

Table 1
Equations for the robustness metric transformations (assuming the aim is to maximize performance).

T1 (performance value transformation)	
Identity transform	$f_1(l_a,s_i) = f(l_a,s_i)$(performance metric f; decision alternative a, l_a; scenario i, s_i)
Regret transform	$f_1(l_a,s_i) = \underset{j}{\text{argmax}} f(l_a,s_j) - f(l_a,s_i)$
Satisficing regret transform	$f_1(l_a,s_i) = \begin{cases} 0, & f(l_a,s_i) \geq c \\ c - f(l_a,s_i), & f(l_a,s_i) \leq c \end{cases}$(constraint of c)
Satisficing transform	$f_1(l_a,s_i) = \begin{cases} 1, & f(l_a,s_i) \geq c \\ 0, & f(l_a,s_i) \leq c \end{cases}$
T2 (scenario subset selection)	
Select a single percentile	$f_2(l_a,S) = f_1(l_a,s_p)$($p$th percentile; S is full set of scenarios)
Select bounds of range	$f_2(l_a,S) = \{f_1(l_a,s_{up}), f_1(l_a,s_{lp})\}$(where T3 is magnitude of range) (up is the upper percentile, lp is the lower percentile)
Select range of scenarios	$f_2(l_a,S) = \{f_1(l_a,s_i) \; \forall \; i : f_1(l_a,s_{lp}) \leq f_1(l_a,s_i) \leq f_1(l_a,s_{up})\}$
T3 (performance value aggregation)	
Identity transform	$f_3(l_a,S) = f_2(l_a,S)$
Magnitude of range	$f_3(l_a,S) = f_2(l_a,s_{up}) - f_2(l_a,s_{lp})$
Mean	$f_3(l_a,S) = (\sum_{i=1}^{n} f_2(l_a,s_i))/n$

failure is important. In the previous example, a water supply system would be deemed to fail if demand was greater than the sustainable yield, so all decision alternatives could be classified as passing or failing in each scenario. Alternatively, a decision-maker looking at a water supply system could choose to set a threshold as the point where supply is low enough to cause water restrictions, in which case the magnitude of failure does matter, since less water would mean greater water restrictions.

If there is no performance threshold, then the question is whether the aim is to maximize performance or avoid making the "wrong" decision. By avoiding making the "wrong" decision, we are referring to some decision-makers who may have a desire to avoid selecting decision alternatives if there is a potential that, with hindsight, the decision-maker could be criticized for having made the wrong decision, even if at the time of making the decision, it appeared to be a reasonable option with the available information. For example, many publicly owned water authorities face intense public scrutiny, and for that reason some decision-makers may want to avoid making decisions (e.g. large capital expenditure projects, such as a desalination plant for water security) that could be perceived to be "wrong" after the fact (e.g. an unnecessary expenditure because climate change or population growth eventuates to be less than expected). Decision-makers in this situation may prefer to choose a decision alternative that is not the best in any single scenario but is never far from the best decision alternative in extreme good or bad scenarios.

The next step in Fig. 3 is to determine whether it is most important to get an indication of the *level* of performance, or the *range* (or variability) of performance across multiple plausible futures. Generally, the former is of greatest importance, but the latter may also be important as an additional robustness metric given that decision makers would generally prefer to know the precise outcome of a particular decision rather than a highly uncertain outcome. Nevertheless, if the range of performance is considered important, it would generally be considered as a secondary metric to be used in addition to a robustness metric that indicates the level of performance. For example, in a water supply system, it would be most important for decision makers to have an indication of how much water each decision alternative will supply. But, as an additional metric, the decision makers may opt to choose a decision alternative with a slightly lower performance if the range of performance values is smaller across the different scenarios, since they would have greater confidence in the outcome of their decision regardless of which scenario is realized. In this case, decision makers could consider both robustness metrics in their decision-making.

In the case where an indication of the *level* of performance is chosen as being most important, this is based on the level of risk tolerance or risk aversion required for the problem or preferred by the decision-maker. Often, a high level of risk aversion is warranted when the consequences of failure are very high. For example, the design of a water supply system would require a high level of risk aversion. In contrast, the level of risk aversion associated with the design of a stormwater system for a road in a remote area would generally be considerably less. Alternatively, the level of risk aversion may also be a matter of personal preference, with some decision-makers being more tolerant of risk than others, or it may be a matter of regulation, where government set some minimum level of risk aversion. This scale of risk aversion and risk tolerance can be represented in the selection of an appropriate robustness metric by choosing a percentile between 0% and 100%, with 0% reflecting the worst-case scenario (extreme risk aversion) for each decision alternative (i.e. 0% of scenarios have worse performance) and 100% reflecting the best-case scenario (extreme risk tolerance). It must be noted that unlike a probabilistic assessment of level of performance, percentiles that are used for robustness metrics are reflective of relative (not absolute) risk. For example, the 50th percentile does not reflect the median level of performance that can be expected in future, however, it does represent a level of performance that is worse than the 90th percentile and therefore is more risk averse than selecting the 90th percentile.

Once the most appropriate "custom" robustness metric has been determined based on the attributes of the decision context (the properties of the problem) and the preferences of the decision-maker with the aid of the process in Fig. 3, the next decision point is whether the scenarios under which the performance of the decision alternatives under consideration should be evaluated are known or not (Fig. 2, Box 5). From here, the same process is followed as if the robustness metric was known in advance (as described above), leading to a scenario analysis (Fig. 2, Box 13) if the scenarios are unknown, and the selection of the most robust decision alternative or decision alternatives of acceptable levels of robustness if the scenarios are known (Fig. 2, Boxes 10 and 15). As with the selection of a performance metric in any problem (including deterministic and probabilistic problems), it is entirely possible that decision-makers will not be able to agree on which metric to use (i.e. in the case of selecting a robustness metric, which transformations are most appropriate for creating a robustness metric). Decision makers may choose to use multiple metrics to consider multiple points of view, such as using both reliability and vulnerability to measure performance in a probabilistic uncertainty problem.

3. The RAPID software package

The RAPID (Robustness Analysis Producing Intelligent Decisions) Python software package implements the generic guidance framework introduced in Fig. 2 in a user-friendly and consistent manner, including functionality to guide the user through the process of creating a custom robustness metric as described in Fig. 3. RAPID is implemented in Python, which is being used increasingly for scientific modelling because it is a high-level, general-purpose, and open source programming language with an emphasis on code readability. It also has a very large standard library, and a significant repository of third-party Python packages. The fact that the RAPID package is written in Python also makes it easier for it to interact with many other software packages, including two packages for robust decision making analysis, the Exploratory Modelling (EM) Workbench (Kwakkel, 2017) and Rhodium (Hadjimichael et al., 2020), which are also written in Python. As the EM Workbench includes functionality for the generation of decision alternatives (i.e. policy options, solutions, etc.), the generation of scenarios (i.e. states of the world, plausible futures) and vulnerability analyses

(including scenario discovery, feature scoring, and sensitivity analyses), the EM Workbench can be used for the creation of all of the inputs needed for the generic guidance framework (Fig. 2) implemented by the RAPID software package. The gap in the EM Workbench that the RAPID software package fills is to provide simple building blocks for robustness metrics that allow robustness metrics to be constructed in a consistent manner that corresponds to the guidance framework introduced in this paper. This also conforms to software best practices such as the Unix philosophy which emphasizes smaller, more modular software packages rather than one large software package.

As shown in Fig. 4, the processes from the guidance framework are implemented across two sub-packages, *metrics* and *analysis* (colored purple and green, respectively, in Fig. 4). The sub-package *metrics* contains functions implementing each of the three transformations required for the calculation of robustness metrics (Fig. 1) (see Table 2 for available options at each of the three transformations). This enables user-defined "custom" robustness metrics to be developed (see Table 2 for available options at each of the three transformations), including those obtained by following the process outlined in Fig. 3 (either by manually selecting the transformations and combining them using the *custom_R_metric* function, or by interacting with the guidance helper function, *guidance_to_R*, which steps through the process in Fig. 3). A number of commonly used robustness metrics have also been pre-programmed (see Table 3 for these metrics, as well as the corresponding choices at each of the three transformations). These robustness metrics can then be used to calculate the robustness values for given decision alternatives, scenarios, and performance metrics, as highlighted in Fig. 1.

The *analysis* sub-package (colored green in Fig. 4) contains the quantitative methods and visualizations for assessing the sensitivity of the relative robustness values of different decision alternatives to the choice of robustness metrics and/or scenario sets. For the assessment of the impact of scenario selection on robustness values, the software package uses the approach outlined by McPhail et al. (2020). That is, the software package calculates the difference in robustness values when the robustness is calculated using two different sets of scenarios. First, for each decision alternative, l_i, one can calculate robustness, R, using one set of scenarios, S_a, then calculate the robustness again with a second set of scenarios, S_b, and finally compare the relative difference between the two robustness values. We use the average relative difference, Δ, across all n decision alternatives:

$$\Delta = \sum_{i=1}^{n} \frac{|R(l_i, S_a) - R(l_i, S_b)|}{\left(\frac{|R(l_i,S_a)|+|R(l_i,S_b)|}{2}\right)} \Bigg/ n \times 100\%$$

Similarly, for the assessment of the impact that scenario selection has on the rankings of the decision alternatives, we follow McPhail et al. (2020), using Kendall's Tau-b ranking correlation to determine the difference in rankings when robustness is calculated using two different sets of scenarios. Kendall's Tau-b ranking has a range between -1 and $+1$ (inclusive), where -1 indicates that all decision alternatives have opposite rankings, $+1$ indicates that the rankings are exactly the same, and 0 implies that there is no correlation between the rankings. Specifically, Kendall's Tau-b metric is used to compare two sets of robustness values, one calculated using a set of scenarios, S_a, and the other calculated using a different set of scenarios, S_b:

$$\{R(l_1, S_a),\ R(l_2, S_a), \ldots, R(l_n, S_a)\}$$

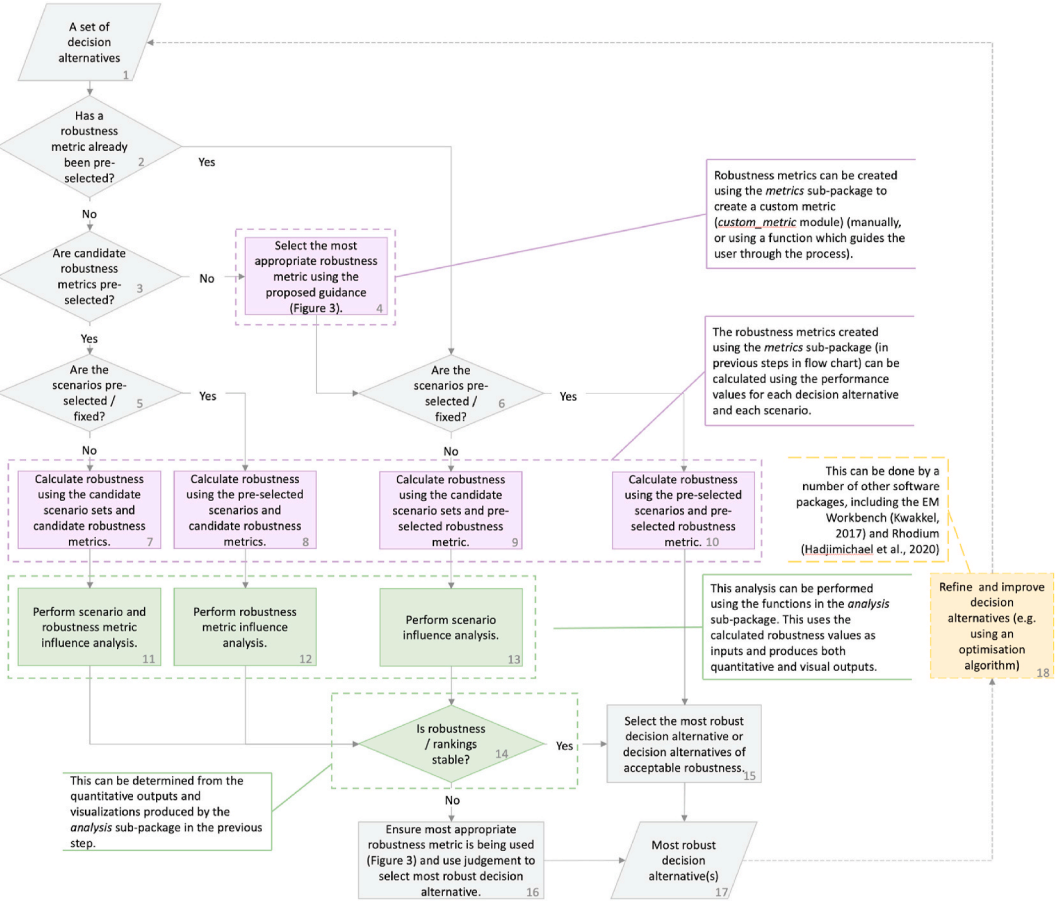


Fig. 4. The general guidance framework introduced in Fig. 2, with an explanation of how the RAPID software package assists in the implementation of this guidance and one way that it can interact with the EM Workbench package.

Table 2
Options for each of the three robustness metric calculation transformations included in the software package.

Transformation number	Transformation name	Used in traditional metrics	Used in proposed guidance	Software package function
T1 (performance value transformation)	Identity	✓	✓	*t1.identity*
	Regret	✓	✓	*t1.regret_from_best_da* (regret from best decision alternative)
	Satisficing regret	✓	✓	*t1.satisficing_regret*
	Regret from median	✓		*t1.regret_from_median*
	Regret from value	✓	✓	*t1.regret_from_value* used to calculate the other regret metrics (which are all calculating regret with respect to different values)
	Satisficing	✓	✓	*t1.satisfice*
T2 (scenario subset selection)	Select a single percentile	✓	✓	*t2.select_percentiles*, *t2.worst_case* (for 0^{th} percentile), or *t2.best_case* (for 100th percentile)
	Worst- and best-case scenarios	✓		*t2.worst_and_best_cases*
	Select bounds of range		✓	*t2.select_percentiles*
	Select range of scenarios	✓	✓	*t2.range*, *t2.worst_half*, or *t2.all_scenarios*
T3 (performance value aggregation)	Identity transform	✓	✓	*t3.f_identity*
	Magnitude of range		✓	*t3.f_range*
	Mean	✓	✓	*t3.f_mean*
	Sum	✓		*t3.f_sum*
	Weighted sum	✓		*t3.f_w_sum*
	Variance	✓		*t3.f_variance*
	Mean-variance	✓		*t3.f_mean_vairance*
	Skew	✓		*t3.f_skew*
	Kurtosis	✓		*t3.f_kurtosis*

$$\{R(l_1, S_b),\ R(l_2, S_b), \ldots, R(l_n, S_b)\}$$

Similarly, Kendall's Tau-b ranking can be used to assess the difference in rankings when robustness is calculated using two different *robustness metrics* (rather than two different *sets of scenarios*, as considered above), as recommended by McPhail et al. (2018) as a quantitative alternative to the comparison of robustness metrics using visual methods such as parallel axes plots (Giuliani and Castelletti, 2016). Specifically, Kendall's Tau-b metric is used to compare two sets of robustness values, one calculated using a robustness metric, R_1, and the other calculated using a different robustness metric, R_2:

$$\{R_1(l_1, S),\ R_1(l_2, S), \ldots, R_1(l_n, S)\}$$

$$\{R_2(l_1, S),\ R_2(l_2, S), \ldots, R_2(l_n, S)\}$$

Note that since we are comparing different robustness metrics, they can be in different scales or units. Therefore, the relative difference in robustness values cannot be calculated, unlike when assessing the impact of scenario selections on robustness values, where a single robustness metric is used and therefore the values can be compared directly.

The structure of the two sub-packages mentioned above (i.e. *metrics* and *analysis*) is as follows:

- *metrics*; a sub-package containing functions for each of the three robustness metric transformations, common metrics from the literature, functions to help build custom robustness metrics, and a helper function which asks the user the questions from the guidance provided in Section 2. This sub-package is structured as:
  - *transforms*; a sub-package, split into the three transformations (T1, T2, T3) as three separate modules (the *t1*, *t2*, and *t3* sub-packages), which implement the transformations listed in Table 2. Note that if the aim is to minimize the performance value (e.g. if cost is the measure of performance), the sign of the performance values is inverted in all T1 functions, because this ensures that the value of all robustness metrics is maximized.
  - *common_metrics*; a sub-package that calculates the following 11 commonly used robustness metrics (McPhail et al., 2018): Maximin, Maximax, Hurwicz's Optimism-Pessimism Rule, Laplace's Principle of Insufficient Reason, Minimax Regret, Percentile Minimax Regret, Mean-Variance, Undesirable Deviations, Percentile-based Skew, Percentile-based Kurtosis, and Starr's Domain Criterion, implementing the three transformations from the *transforms* sub-package (as listed in Table 3).
  - *custom_metrics*; a module that includes a function (*custom_R_metric*) for creating a custom robustness metric composed of three transformations (from the *transforms* sub-package), and also provides a helper function for stepping users through the flowchart in Fig. 3 to create a custom robustness metric that is most appropriate for the decision context under consideration (the *guidance_to_R* function). This helper function asks questions of the user and uses the responses to create the resulting custom robustness metric (using the *custom_R_metric* function).
- *analysis;* a sub-package that enables the influence of different sets of scenarios and robustness metrics on the robustness values and rankings to be determined (the *scenarios_similarity* and *robustness_similarity* functions, respectively). This module also produces plots to visualize the influence that the scenarios and robustness metrics have, including (i) the *delta_plot* function for plotting the relative difference in robustness values (i.e. the deltas) caused by different scenario selections or robustness metrics and (ii) the *tau_plot* function for plotting the ranking similarity (i.e. the Kendall's Tau-b correlation) from different robustness metrics (both functions explained in more detail above).

A number of examples using the software package are also contained within the package, including a multi-objective robust optimization of the Lake Problem (also explored in Section 4); a common, hypothetical environmental modelling problem used in the environmental systems modelling literature.

4. The Lake Problem

4.1. Background

The *examples* directory in the RAPID package includes the Lake Problem as an example of common usage of the package. The Lake Problem is a hypothetical, stylized model that is well-represented in the literature (Carpenter et al., 1999; Eker and Kwakkel, 2018; Hadka et al., 2015; Kwakkel, 2017; Lempert and Collins, 2007; McPhail et al., 2020; Quinn et al., 2017; Singh et al., 2015; Ward et al., 2015), and represents a city that must decide the amount of pollution that it releases into a lake. There are four competing objectives: (1) the average concentration of phosphorous in the lake; (2) the frequency of pollution levels exceeding a critical threshold (i.e. the reliability) (3) the economic benefit (i.e. economic utility); of polluting the lake; and (4) a penalty for if the change in level of pollution is too high from year to year (i.e. a measure of inertia of the pollution) to help achieve more realistic and appropriate solutions. Both deep and stochastic uncertainties are present for the natural inflows of pollution into the lake, the natural removal and recycling rates of pollution in the lake, and the discount rate for the economic benefits. To illustrate the generic guidance framework on the Lake Problem, we follow several different pathways through the framework (Fig. 2), including the situations where:

1. Section 4.2 – The robustness metric is unknown, and there are no candidate robustness metrics under consideration. The method for generating the scenarios is known.
2. Section 4.3 – The robustness metric is unknown and there are no candidate robustness metrics under consideration. There are multiple candidate sets of scenarios.
3. Section 4.4 – The robustness metric is unknown, however, there are multiple candidate robustness metrics. The method for generating the scenarios is known.

4.2. No candidate robustness metrics but scenario generation method known

Following the guidance framework, we consider a situation in which we aim to use an optimization process (Fig. 4, Box 18) to determine a set of robust decision alternatives. In this situation, we also assume that the robustness metric is unknown (Fig. 4, Box 2) and that there are no candidate robustness metrics (Fig. 4, Box 3), leading to Box 4 in Fig. 4. Here, we deviate from the EM Workbench (Kwakkel, 2017) example of the Lake Problem, which used standard robustness metrics for each of the objectives. In our example, we create a custom robustness metric by following the guidelines in Fig. 3. Note that the creation of these custom robustness metrics is illustrative of how to follow the guidance and uses many assumptions about decision maker preferences that are not present in previous formulations of the Lake Problem. Also, note that we have created one robustness metric for each of the four Lake Problem performance metrics, but this need not be the case.

First, for the average concentration of the phosphorous in the lake, we decide that there is no meaningful threshold (note that some studies have created a threshold for this objective), and that we are most interested in making the best decision, which gives us the identity transform for T1. We are looking for an indication of the level of performance, leading to the identity transform for T3, and are relatively risk averse, so the 25th percentile is used for T2 (also see summary in Table 4).

For the reliability, we assume a situation where a requirement for the project is a minimum of 80% reliability (i.e. a decision alternative performs satisfactorily in an individual scenario only if pollution remains below the critical threshold for 80% of the time) and that this requirement should be met in as many scenarios as possible. Thus, the T1 transformation is the satisficing transform and the T3 transformation is the mean. It is also decided that the aim is to understand what percentage of all scenarios under consideration have acceptable performance, and so all scenarios are selected for T2.

For the economic utility, it is assumed that a level of 0.75 is required (the economic utility is dimensionless in this study, but 0.75 represents some minimum economic benefit that must be achieved), and that any level lower than this will have significant consequences. Therefore, the satisficing regret transform is used, since this can accommodate the threshold of 0.75 and penalizes decision alternatives in each scenario that fail to achieve this. The level of performance (i.e. the level of potential regret) is most important, and therefore the identity transform is used for T3. It is also assumed that the decision-maker has a moderate level of risk aversion for this objective, and T2 is the 50th percentile of performance (i.e. regret).

The inertia is a measure of how much the decision alternative options vary from year to year (it is preferred that there are no significant changes in the level of pollution from one year to the next). We are not using a specific threshold for this (although some other studies have), and the objective of the decision-maker is to make the best decision regarding the level of performance (level of inertia). Therefore, the identity transform is chosen for T1 and T3. Again, the level of risk aversion is moderate for this objective, and thus the 50th percentile is chosen for T2.

Returning to the overarching guidance framework for robustness analysis (Figs. 2 and 4), now that we have the robustness metrics (Fig. 4, Box 4) and the scenarios are known (Fig. 4, Box 6), we can calculate robustness using the selected scenarios and selected (custom) robustness metrics (Fig. 4, Box 10). To illustrate this with the RAPID software package, we build upon an example of the Lake Problem that is included in the EM Workbench (Kwakkel, 2017), with the following methodology:

Table 3
Commonly used robustness metrics included in the software package, as well as corresponding choices at each of the three transformations.

Metric name	T1 (performance value transformation)	T2 (scenario subset selection)	T3 (performance value aggregation)	Software package function
Maximin	Identity	Worst-case	Identity	*maximin*
Maximax	Identity	Best-case	Identity	*maximax*
Hurwicz's Optimism-Pessimism Rule	Identity	Worst- and best-cases	Mean	*hurwicz*
Laplace's Principle of Insufficient Reason	Identity	All scenarios	Mean	*laplace*
Minimax Regret	Regret	Worst-case	Identity	*minimax_regret*
Percentile Regret (e.g. 90th percentile regret)	Regret	Percentile	Identity	*percentile_regret*
Mean-variance	Identity	All scenarios	Mean-variance	*mean_variance*
Undesirable deviations	Regret from median	Worst-half	Sum	*undesirable_deviations*
Percentile-based skew	Identity	10th, 50th, and 90th percentiles	Skew	*percentile_skew*
Percentile-based kurtosis	Identity	10th, 25th, 75th, and 90th percentiles	Kurtosis	*percentile_kurtosis*
Starr's Domain Criterion	Satisfice	All scenarios	Mean	*starrs_domain*

1. Using the EM Workbench, we formulate the model (e.g. uncertain parameters, objectives, etc.).
2. Using the RAPID package, we create the custom robustness metrics defined above in Table 4.
3. Using the EM Workbench, we formulate an optimization problem with the formulated model (from Step 1) and custom robustness metrics (from Step 2).
4. Using the EM Workbench, we run the optimization to determine the most robust decision alternatives. Note that this means we begin with random decision alternatives in Fig. 4, Box1, but then the EM Workbench refines these decision alternatives using the feedback loop with Box 18.

For Step 1, the Lake Problem was specified in the same manner as in the EM Workbench example (i.e. the uncertain parameters, options for the decision alternatives, and the performance objectives were defined in the same way) using the EM Workbench functionality for defining a model.

```
def get_lake_model():
    """Returns a fully formulated model of the lake problem."""
    # instantiate the model
    lake_model = Model('lakeproblem', function=lake_problem)
    lake_model.time_horizon = 100

    # specify uncertainties
    lake_model.uncertainties = [RealParameter('b', 0.1, 0.45),
                                RealParameter('q', 2.0, 4.5),
                                RealParameter('mean', 0.01, 0.05),
                                RealParameter('stdev', 0.001, 0.005),
                                RealParameter('delta', 0.93, 0.99)]

    # set levers, one for each time step
    lake_model.levers = [RealParameter(str(i), 0, 0.1) for i in
                         range(lake_model.time_horizon)]

    # specify outcomes
    lake_model.outcomes = [ScalarOutcome('max_P',),
                           ScalarOutcome('utility'),
                           ScalarOutcome('inertia'),
                           ScalarOutcome('reliability')]

    # override some of the defaults of the model
    lake_model.constants = [Constant('alpha', 0.41),
                            Constant('nsamples', 150)]
    return lake_model
```

Table 4
Custom robustness metrics created for the Lake Problem.

Performance metric	T1	T2	T3
Average phosphorous	Identity	25th percentile	Identity
Reliability	Satisfice (threshold 80%)	All scenarios	Mean
Economic utility	Magnitude below threshold of 0.75	50th percentile	Identity
Inertia	Identity	50th percentile	Identity

For Step 2, the custom robustness metrics defined in Table 4 were first specified using the RAPID package and then put into the form required for the EM Workbench. Note that when defining these custom metrics, it was possible to use any combination of the three robustness

```
******
Create a custom robustness metric
******

Does a meaningful threshold for the level of performance exist? (y/n)
        E.g. supply must be greater than demand, or
             cost must be kept within a budget?
n

Is it most important to (a) make the best decision, or (b) avoid making the wrong decision? (a or b)
a

Is it most important to (a) get an indication of the level of performance or (b) the range of performance? (a or b)
a

Select a  percentile to reflect the level of risk aversion/tolerance.
(i.e. between 0% and 100% reflecting maximum risk aversion and maximum risk tolerance, respectively):
25
```

Fig. 5. Example of the dialogue provided by the metrics.guidance_to_R function in the RAPID package.

metric transformations (from the guidance for decision-makers Fig. 3, and defined in Table 1). These metrics can be defined using code as shown or can also be created using the *metrics.guidance_to_R* function. This function asks the user the questions from the flow chart in Fig. 3, guiding them to the creation of the robustness metric best suited for the problem that can then be used in subsequent analyses (as shown in Fig. 5). The output from the *metrics.guidance_to_R* function is the same as the output from the *metrics.custom_R_metric* function in the example code.

```
def get_custom_R_metrics():
    """Returns the custom robustness metrics from paper."""
    av_vulnerability_R = functools.partial(
        custom_R_metric(t1.identity, t2.select_percentiles, t3.f_identity),
        maximise=False,
        t2_kwargs={'percentiles': [0.25]})
    reliability_R = functools.partial(
        custom_R_metric(t1.satisfice, t2.all_scenarios, t3.f_mean),
        t1_kwargs={'threshold': 0.8},
        maximise=True)
    utility_R = functools.partial(
        custom_R_metric(t1.satisficing_regret, t2.select_percentiles, t3.f_identity),
        maximise=True,
        t1_kwargs={'threshold': 0.75},
        t2_kwargs={'percentiles': [0.5]})
    inertia_R = functools.partial(
        custom_R_metric(t1.identity, t2.select_percentiles, t3.f_identity),
        maximise=True,
        t2_kwargs={'percentiles': [0.5]})

    # Note that we want to minimise max_P, so we define this in the
    # robustness metrics above (maximise=False), and this changes
    # the sign of the robustness metric, so that we can always
    # make the objective to MAXIMIZE robustness.
    robustness_functions = [
        ScalarOutcome(
            'Av vulnerability R',
            kind=ScalarOutcome.MAXIMIZE,
            variable_name='max_P',
            function=av_vulnerability_R),
        ScalarOutcome(
            'Reliability R',
            kind=ScalarOutcome.MAXIMIZE,
            variable_name='reliability',
            function=reliability_R),
        ScalarOutcome(
            'Utility R',
            kind=ScalarOutcome.MAXIMIZE,
            variable_name='utility',
            function=utility_R),
        ScalarOutcome(
            'Inertia R',
            kind=ScalarOutcome.MAXIMIZE,
            variable_name='inertia',
            function=inertia_R)]
```

As per the EM Workbench example (which uses Many-Objective Robust Optimization (MORO)), once the model has been formulated and the robustness metrics have been defined, the next step is to use the EM Workbench to create a set of scenarios, formulate an optimization problem, and then run that optimization problem to find optimally robust decision alternatives. This corresponds to the loop formed by Box 18 in Fig. 4. The results found from this process are shown in Fig. 6. Note again that the robustness metric transformations from the RAPID software package ensure that a higher robustness value is always better (e.g. we seek to minimize vulnerability, but the sign for the robustness metric for vulnerability is switched so that we are aiming to maximize the robustness value). The Pareto front (Fig. 6) shows expected relationships between objectives. For example, better vulnerability also results in better reliability but a worse result for the economic utility. The relationship between the inertia and the other three objectives is weaker.

```
lake_model = get_lake_model()
robustness_functions = get_custom_R_metrics()

n_scenarios = 1000
scenarios = sample_uncertainties(lake_model, n_scenarios)

nfe = 100000  # number of function evaluations

# Run optimisation
with MultiprocessingEvaluator(lake_model) as evaluator:
    robust_results = evaluator.robust_optimize(
        robustness_functions,
        scenarios,
        nfe=nfe,
        population_size=50,
        epsilons=[0.1,] * len(robustness_functions))
```

In this example of following the guidance framework (Figs. 2 and 4), we showed that with no known robustness metric or set of candidate robustness metrics we could create a set of custom robustness metrics that were best suited to the problem (Table 4) using the guidance for creating a custom robustness metric (Fig. 3) to determine the appropriate robustness metric transformations from Table 2. We then created these robustness metrics in a systematic manner using the RAPID software package and used these newly created robustness metrics in conjunction with another software package, the EM Workbench, to run a robust optimization and develop a Pareto front of optimal decision alternatives.

4.3. No candidate robustness metrics and multiple candidate scenario sets

Again, following the guidance framework, we use the optimal decision alternatives from the previous section and assume a situation in which the robustness metric is unknown (Fig. 4, Box 2) and there are no candidate robustness metrics (Fig. 4, Box 3), leading to Box 4 in Fig. 4. Here, we create custom robustness metrics as per Section 4.2, leading to the robustness metrics in Table 4. Unlike in Section 4.2, in this section, we consider a situation where there are multiple candidate sets of scenarios (Fig. 4, Box 6) (e.g. in order to increase the diversity of considered uncertainties (Xexakis et al., 2020)). This situation could occur where the decision makers have identified different sets of scenarios that could all be appropriate for the problem, or the situation where different decision makers create different sets of scenarios for the problem. We note that while we will refer to the decision alternatives from the previous section as "optimal", they were optimal for the scenarios and robustness metrics in the previous section and for the specific formulation of this optimization problem (Maier et al., 2018). They may not be optimal in this proceeding section.

Different sets of scenarios correspond to different sets of points within the space of uncertain model inputs (McPhail et al., 2020). Because these points are inputs to the calculation of robustness (see Fig. 1), different sets of scenarios can lead to differences in robustness. As a simplified illustration of this, we create five candidate sets of 20 scenarios, where each set is sampled from the uncertain variable space using the EM Workbench package with Latin hypercube sampling. We then evaluate the optimal decision alternatives (from Section 4.2) in all 100 scenarios using the EM Workbench package and calculate the robustness for all 5 scenario sets and all decision alternatives using the

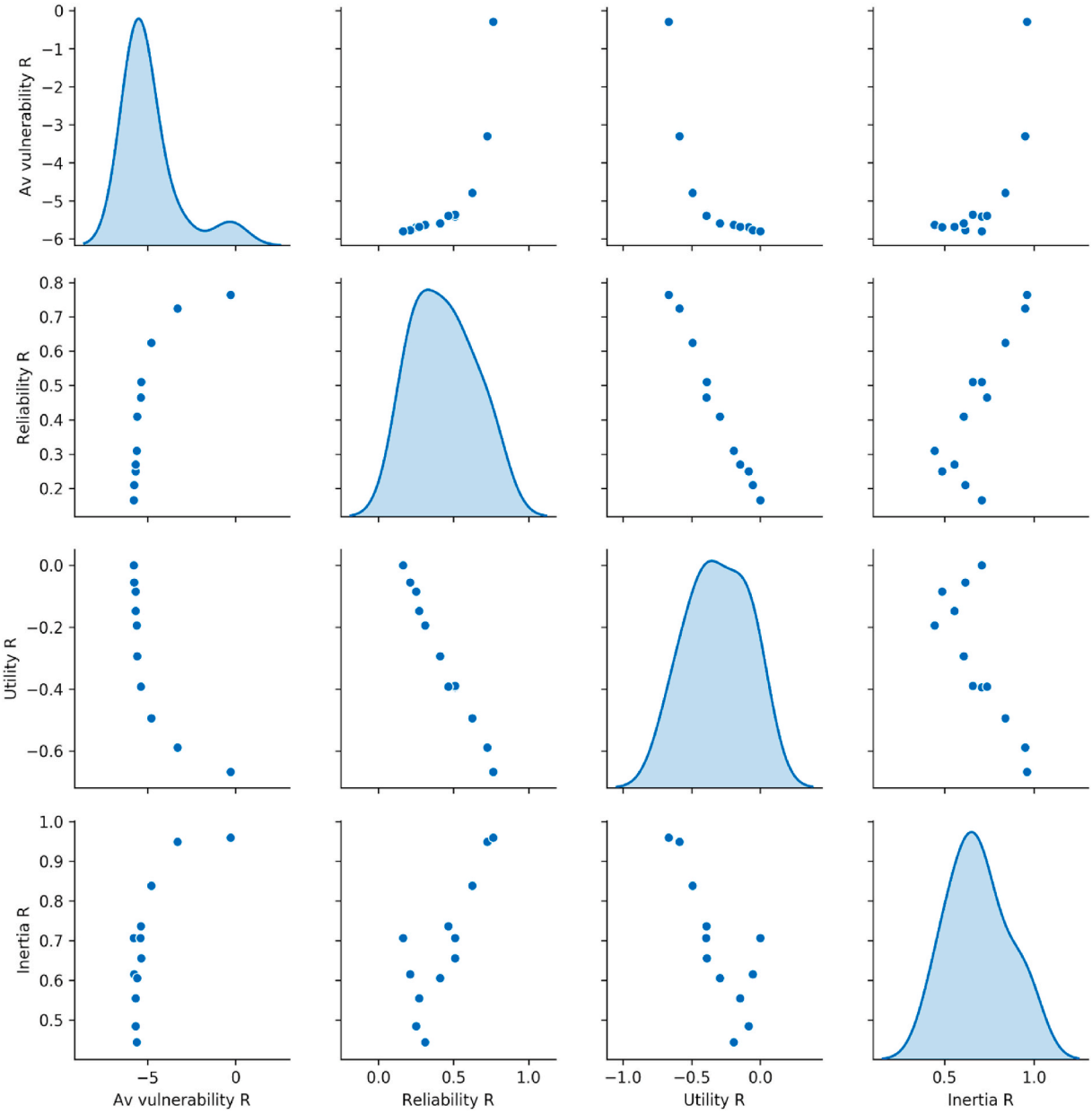


Fig. 6. Example results that can be produced using custom robustness metrics from the RAPID package and multi-objective optimization functionality from the EM Workbench package. The axes are the robustness metrics and each point represents the robustness of a single solution from the 4-dimensional Pareto front.

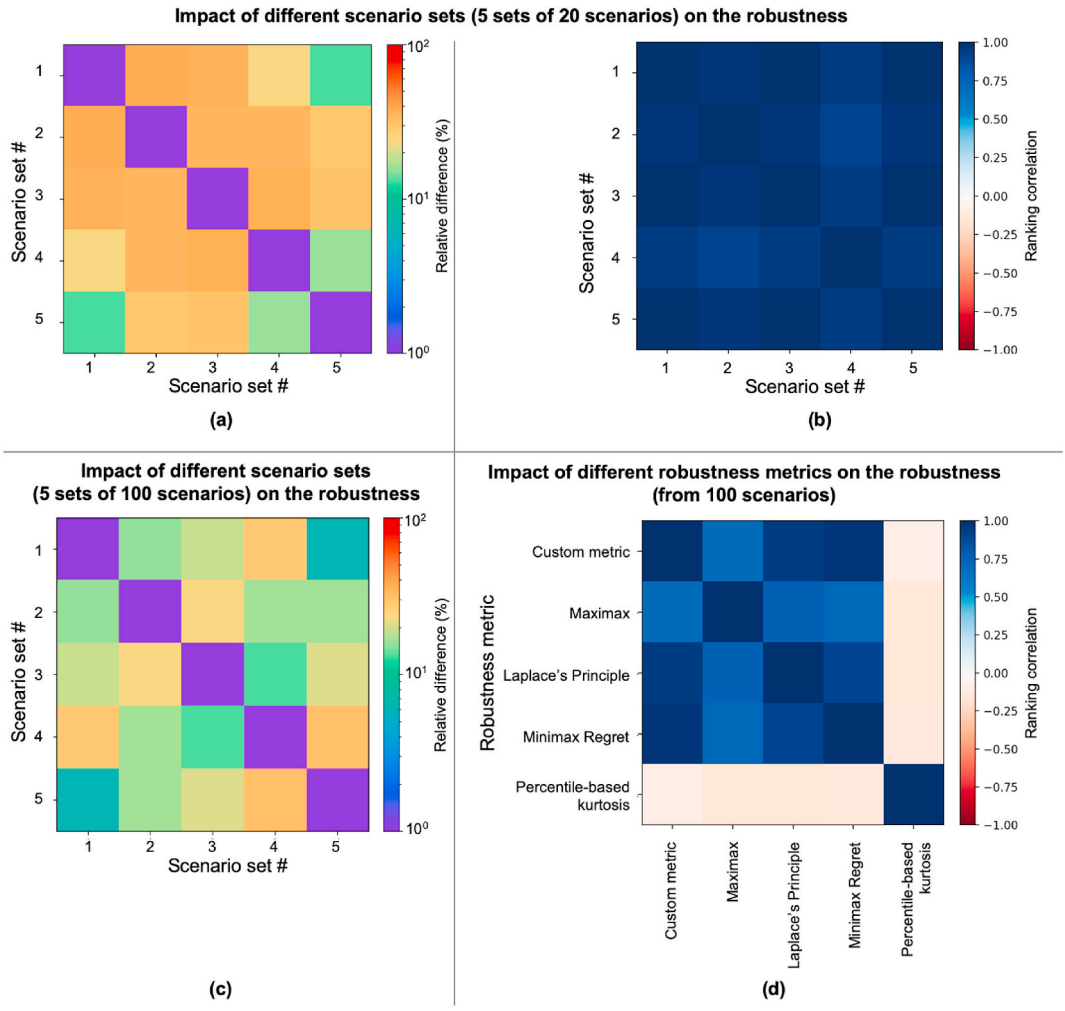


Fig. 7. Example of outputs produced by the RAPID package. For the Lake Problem analyzed as described above: (a) relative difference in robustness for pairs of scenario sets (5 sets of 20 scenarios); (b) ranking similarity for pairs of scenario sets (5 sets of 20 scenarios); (c) relative difference in robustness for pairs of scenario sets (5 sets of 100 scenarios); (d) ranking similarity for pairs of robustness metrics (one set of 100 scenarios).

custom robustness metrics created in Section 4.2 using the RAPID package (Fig. 4, Box 9). Note that for simplicity, we only focus on the vulnerability objective from here on. The same analysis could be applied to each of the four objectives.

```
# Find the influence of scenarios. Here we are creating 5
# sets of 100 scenarios each, all using the same sampling
# method.
scenarios_per_set = 20
n_sets = 5
n_scenarios = scenarios_per_set * n_sets
scenarios = sample_uncertainties(lake_model, n_scenarios)

# Simulate optimal solutions across all scenarios
with MultiprocessingEvaluator(lake_model) as evaluator:
    results = evaluator.perform_experiments(
        scenarios=scenarios, policies=decision_alternatives)
# We will just look at the vulnerability ('max_P') for this example
f = np.reshape(results[1]['max_P'], newshape=(-1, n_scenarios))
# Split the results into the different sets of scenarios
split_f = np.split(f, n_sets, axis=1)
# Calculate robustness for each set of scenarios
# Note that each split_f[set_idx] is a 2D array, with each row being
# a decision alternative, and each column a scenario
R_metric = get_custom_R_metrics()[0]
R = [R_metric(split_f[set_idx]) for set_idx in range(n_sets)]
R = np.transpose(R)

# Calculate similarity in robustness from different scenario sets
delta, tau = analysis.scenarios_similarity(R)
# Plot the deltas using a helper function
analysis.delta_plot(delta)
# Plot the Kendall's tau-b values using a helper function
analysis.tau_plot(tau)
```

Returning to the robustness analysis guidance framework, this brings us to Box 13 in Fig. 4, where we use the *analysis* module of the RAPID package to evaluate the relative difference in robustness values and the Kendall's Tau-b rank correlation (for determining the ranking similarity, as described in Section 3). The *analysis* module also enables us to visualize the influence of the scenarios by creating heatmaps that show all combinations of candidate sets of scenarios (see Fig. 7 (a) and (b)). The diagonal of the heatmaps is each candidate scenario set compared to itself, and therefore the relative difference is 0% (indicated by purple in Fig. 7 (a)) and the ranking correlation is 1 (indicated by blue in Fig. 7 (b)), as expected. From Fig. 7 (a), we can see that for the other comparisons of the scenario sets, the relative difference in robustness values is very high in general (indicated by mostly orange squares, ~30% difference in robustness values). However, there are some cases (e.g. scenario sets 1 and 5, and scenario sets 4 and 5) that are more similar than the rest (indicated by the green). Note that despite a high difference in robustness values, Fig. 7 (b) indicates that the rankings of the decision

alternatives are very stable (consistent with McPhail et al. (2020)).

Given that all five candidate sets of scenarios were sampled using Latin hypercube sampling, it is interesting that the relative difference in robustness is so high in Fig. 7 (a). If the robustness values were important for the decision-making process, it would be difficult to be sure of the actual robustness values because the values would depend on which set of scenarios is being considered (leading to Fig. 4, Box 16). There are many reasons why the relative difference could be high, including dissimilarity in the coverage of the scenario space, and discontinuities in performance space (McPhail et al., 2020). In this example, it is likely to be the former of these reasons because the number of scenarios in each set is small. Running the same code as above, but with a larger number of scenarios (100 scenarios per set, rather than 20 scenarios per set in Fig. 7 (a)), we produce the heatmap shown in Fig. 7 (c). With the larger number of scenarios, the relative difference is significantly lower in general (likely due to a more similar coverage of the scenario space), indicated by the greater number of blue and green squares and the smaller number of orange squares. In this case, we move from Box 14 to Boxes 15 and 17 in Fig. 4, being able to accurately determine the robustness of the decision alternatives. Note that a greater number of scenarios will not always allow a decision-maker to accurately determine the robustness of the decision alternatives, as indicated by the comparisons of scenario sets 4 and 5 in Fig. 7 (c). In this case, we move from Box 14 to Box 16 in which case we need to use human judgement to determine which decision alternatives are the most robust. Alternatively, if we are simply interested in the rankings of the solutions (see Fig. 7 (b)), then we would be able to move from Box 14 to Boxes 15 and 17 without increasing the number of scenarios (assuming that we judge the Kendall's Tau-b values (approximately in the range between 0.7 and 1.0) to be sufficiently high for our purposes).

In this second example of following the guidance framework (Figs. 2 and 4), we showed that, with multiple candidate sets of scenarios, we could use the RAPID software package to evaluate the influence these candidate sets of scenarios had on both the robustness values and rankings. Using the visualizations produced by the software package, we were then able to determine that the relative robustness values of different decision alternatives was not substantially affected by the different scenario sets (Fig. 7 (b)), giving confidence to decision makers and enabling the most robust decision alternative to be identified.

4.4. Multiple candidate robustness metrics and a known set of scenarios

In this situation, we assume that the robustness metric is unknown (Fig. 4, Box 2), but that there are multiple candidate robustness metrics (Fig. 4, Box 5) and that the set of scenarios is known, leading to Box 8 in Fig. 4. Note that if there were multiple candidate sets of scenarios, the analysis would be a combination of the following method and the method in Section 4.3. We create the candidate robustness metrics using the RAPID software package, retaining the original robustness metric for the vulnerability determined in Section 4.2 (Table 4) using the *metrics.custom_R_metric* module, and four traditional robustness metrics as the other candidate metrics, including the Maximax, Laplace's Principle of Insufficient Reason, Minimax Regret, and Percentile-Based Kurtosis robustness metrics (all included in the *metrics.common_metrics* module). As with the previous examples, these metrics were calculated, evaluated (this time across a known set of 100 scenarios, sampled using Latin hypercube sampling), and visualized using the RAPID package (see Fig. 7 (d)).

```
# We now want to test the effects of different robustness metrics,
# across all of the 100 scenarios. We first define a few new
# robustness metrics (in addition to our original R metric for
# the vulnerability). For this example we use some classic metrics
R_metrics = [
    R_metric,  # The original robustness metric
    functools.partial(metrics.maximax, maximise=False),
    functools.partial(metrics.laplace, maximise=False),
    functools.partial(metrics.minimax_regret, maximise=False),
    functools.partial(metrics.percentile_kurtosis, maximise=False)
]

# Calculate robustness for each robustness metric
R = np.transpose([R_metric(f) for R_metric in R_metrics])

# Calculate similarity in robustness from different robustness metrics
tau = analysis.R_metric_similarity(R)
# Plot the Kendall's tau-b values using a helper function
analysis.tau_plot(tau)
```

In the visualization of the similarity in rankings (Fig. 7 (d)), the diagonal shows full ranking similarity (a value of 1, indicated by blue) because that is where each robustness metric is being compared to itself. Most of the metrics also show high levels of ranking similarity with each other, with the exception of the percentile-based kurtosis metric, which shows a slight negative correlation with all other metrics (indicated by the slightly red squares). This potentially leads us from Box 14 to Box 16 in Fig. 4, because it is unknown which ranking is the one that we should follow: the rankings provided by the percentile-based kurtosis or the rankings provided by the rest of the metrics. Again, using our judgement, we decide that the percentile-based kurtosis does not reflect the needs of the decision-makers as much as the other robustness metrics do, because the T3 transformation does not reflect the need to get an indication of the level of performance (as explained by Fig. 3 and by McPhail et al. (2018)). Also, since all of the other candidate solutions generally agree with the custom robustness metric, it follows that we can rely on this custom metric to determine which decision alternative is most robust (Fig. 4, Box 16).

In this final illustration of using the guidance framework (Figs. 2 and 4) and RAPID software package, we showed that with multiple candidate robustness metrics, we can use the software package to evaluate the influence these robustness metrics have on the rankings of the decision alternatives. Using the visualizations produced by the software package, we were then able to determine whether or not the influence was sufficiently large to affect these rankings.

All three of the simple examples considered show that the RAPID package is easy to use and can be used in conjunction with related software packages, such as the EM Workbench. They also show that the RAPID package is a practical tool for systematically following the

guidance framework in Figs. 2 and 4, the guidance for creating robustness metrics in Fig. 3 (shown in Section 4.2), assessing the influence of candidate sets of scenarios on the robustness values and rankings (shown in Section 4.3), and assessing the influence of candidate robustness metrics on the robustness rankings of decision alternatives (shown in Section 4.4). Note that since this is a multi-objective problem, there is no single most robust decision alternative and decision-makers can only narrow down the choice of decision alternatives to those that represent the best trade-offs between the four objectives. It is likely that the decision of a final decision alternative would require further analysis by other (potentially more senior) decision-makers to determine which trade-off represents the best strategic choice.

5. Summary and conclusions

Robustness is important in the long-term planning of environmental systems. However, there is a variety of metrics that can be used to calculate the robustness of a set of decision alternatives, and recent research has shown that the choice of metric can affect the ranking of decision alternatives. Similarly, there is a variety of approaches to selecting or generating scenarios (which are an input to the calculation of robustness), and the chosen approach has also been shown to have an effect on the robustness values and rankings of decision alternatives. Despite the uncertainty associated with the selection of scenarios and robustness metrics when determining the rankings of decision alternatives under deep uncertainty, no guidance exists for decision-makers on which choices to make.

As a response to this need for guidance, this paper proposes a generic guidance framework to assist decision-makers in the identification of robust decision alternatives (Fig. 2). This framework caters to a variety of situations where the scenarios and/or robustness metrics are known or not known. The framework includes guidance on how to create a custom robustness metric for the problem at hand (Fig. 3), based on the attributes of the problem (e.g. the presence of performance thresholds/tipping points, or the objectives of the problem), as well as the preferences of the decision-maker (e.g. the level of risk-aversion). The output from the guidance for the creation of a custom robustness metric is three robustness metric transformations (Table 1), which form the robustness metric when combined (Fig. 1). The overarching guidance framework also identifies situations where quantitative analyses can be used to determine the influence that the selection of scenarios and/or the choice of robustness metric has on the rankings of decision alternatives.

This paper also introduces an open-source software package, the RAPID (Robustness Analysis Producing Intelligent Decisions) package, to assist with the consistency and ease-of-use of implementing the guidance framework (see Fig. 4). The software package includes a module for the creation of custom robustness metrics using a wide range of robustness metric transformations (Table 2), including a function that leads the user through the guidance of how to create the robustness metric most suited for the problem at hand (Fig. 3). It also includes a variety of commonly used traditional robustness metrics from the literature (Table 3). The software package also contains a module for the calculation and visualization of the impact of the selection of scenarios and choice of robustness metric on robustness values and rankings.

To illustrate the implementation of the guidance framework and RAPID software package, we consider the Lake Problem, a hypothetical lake pollution problem, commonly used in literature. We use the guidance in Fig. 3 to create custom robustness metrics for The Lake Problem, based on hypothetical problem attributes and decision-maker preferences (Table 4). In conjunction with the EM Workbench (Kwakkel, 2017), we use these robustness metrics as objectives in a robust optimization to create a set of robust decision alternatives. As an example of the utility of the guidance framework and software package, we use these optimal decision alternatives to consider a situation where there are multiple sets of scenarios under consideration. Using the RAPID software package, we visualize the impact of these different sets of scenarios, showing that the robustness values are affected (Fig. 7 (a)), but rankings of the decision alternatives are not (Fig. 7 (b)), providing confidence to decision makers that the most robust decision alternative has been identified. We also show that when using a larger set of scenarios, the impact of the set of scenarios on the robustness values is greatly decreased (Fig. 7 (c)). In another example to highlight the utility of the guidance framework and software package, we consider a situation where there is a variety of candidate robustness metrics. We use the framework and software package to visualize the impact of the choice of robustness metric (Fig. 7 (d)), showing that most of the metrics agree on the rankings of the decision alternatives, again providing confidence to decision makers that the most robust solution has been identified.

This guidance framework and software package assist decision-makers in the identification of robust decision alternatives. It does so in a systematic way, and the software package increases the consistency and ease-of-use of implementing the guidance. The guidance framework and software package are generic and cater to a wide variety of circumstances where the robustness metrics and/or scenarios may or may not be known, greatly increasing the accessibility of robustness analyses and techniques to decision-makers. After identifying the most robust decision alternatives or those decision alternatives that represent the best trade-offs across multiple objectives, decision-makers are able to then explore those decision alternatives in more detail. With these selected decision alternatives, decision-makers can better understand what makes some decision alternatives more robust, explain to other stakeholders what it is about these decisions that makes them robust, based on the information obtained from the guidance framework.

Software availability

The Lake Model is widely available on GitHub in multiple repositories, including in the EMAworkbench: https://github.com/quaquel/EMAworkbench.

The RAPID (Robustness Analysis Producing Intelligent Decisions) software package is available on GitHub (https://github.com/cameronmcphail/RAPID) and in the Python Package Index (PyPI) (https://pypi.org/project/rapidrobustness/) (archived at doi.org/10.5281/zenodo.4171495).

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

Thanks is given to SA Water Corporation (Australia) who support the research of Cameron McPhail through Water Research Australia, and thanks is also given to Water Research Australia. The authors would also like to thank Andrea Castelletti and Matteo Giuliani (both from Politecnico di Milano) for their important conceptual contributions to this research.

References

Bartholomew, E., Kwakkel, J.H., 2020. On considering robustness in the search phase of robust decision making: a comparison of many-objective robust decision making, multi-scenario many-objective robust decision making, and many objective robust optimization. Environ. Model. Software 127, 104699.

Borgomeo, E., Mortazavi-Naeini, M., Hall, J.W., Guillod, B.P., 2018. Risk, robustness and water resources planning under uncertainty. Earth's Futur 6, 468–487.

Börjeson, L., Höjer, M., Dreborg, K.H., Ekvall, T., Finnveden, G., 2006. Scenario types and techniques: towards a user's guide. Futures 38, 723–739. https://doi.org/10.1016/j.futures.2005.12.002.

Bradfield, R., Wright, G., Burt, G., Cairns, G., Van Der Heijden, K., 2005. The origins and evolution of scenario techniques in long range business planning. Futures 37, 795–812. https://doi.org/10.1016/j.futures.2005.01.003.

Carpenter, S.R., Ludwig, D., Brock, W.A., 1999. Management of eutrophication for lakes subject to potentially irreversible change. Ecol. Appl. 9, 751–771.
Culley, S., Bennett, B., Westra, S., Maier, H.R., 2019. Generating realistic perturbed hydrometeorological time series to inform scenario-neutral climate impact assessments. J. Hydrol. 576, 111–122.
Culley, S., Noble, S., Yates, A., Timbs, M., Westra, S., Maier, H.R., Giuliani, M., Castelletti, A., 2016. A bottom-up approach to identifying the maximum operational adaptive capacity of water resource systems to a changing climate. Water Resour. Res. 52, 6751–6768. https://doi.org/10.1002/2015WR018253.
Drouet, L., Bosetti, V., Tavoni, M., 2015. Selection of climate policies under the uncertainties in the fifth assessment report of the IPCC. Nat. Clim. Change 5, 937–940.
Eker, S., Kwakkel, J.H., 2018. Including robustness considerations in the search phase of many-objective robust decision making. Environ. Model. Software 105, 201–216. https://doi.org/10.1016/j.envsoft.2018.03.029.
Giudici, F., Castelletti, A., Giuliani, M., Maier, H.R., 2020. An active learning approach for identifying the smallest subset of informative scenarios for robust planning under deep uncertainty. Environ. Model. Software 127, 104681.
Giuliani, M., Castelletti, A., 2016. Is robustness really robust? How different definitions of robustness impact decision-making under climate change. Climatic Change 135, 409–424. https://doi.org/10.1007/s10584-015-1586-9.
Hadjimichael, A., Gold, D., Hadka, D., Reed, P., 2020. Rhodium: Python library for many-objective robust decision making and exploratory modeling. J. Open Res. Software 8.
Hadka, D., Herman, J., Reed, P., Keller, K., 2015. An open source framework for many-objective robust decision making. Environ. Model. Software 74, 114–129. https://doi.org/10.1016/j.envsoft.2015.07.014.
Hall, J.W., Lempert, R.J., Keller, K., Hackbarth, A., Mijere, C., Mcinerney, D.J., 2012. Robust climate policies under uncertainty: a comparison of robust decision making and info-gap methods. Risk Anal. 32, 1657–1672. https://doi.org/10.1111/j.1539-6924.2012.01802.x.
Herman, J.D., Reed, P.M., Zeff, H.B., Characklis, G.W., 2015. How should robustness Be defined for water systems planning under change? J. Water Resour. Plann. Manag. 141, 04015012 https://doi.org/10.1061/(ASCE)WR.1943-5452.0000509.
Herman, J.D., Zeff, H.B., Reed, P.M., Characklis, G.W., 2014. Beyond optimality: multistakeholder robustness tradeoffs for regional water portfolio planning under deep uncertainty. Water Resour. Res. 50, 7692–7713.
Kasprzyk, J.R., Nataraj, S., Reed, P.M., Lempert, R.J., 2013. Many objective robust decision making for complex environmental systems undergoing change. Environ. Model. Software 42, 55–71. https://doi.org/10.1016/j.envsoft.2012.12.007.
Kwakkel, J.H., 2017. The Exploratory Modeling Workbench: an open source toolkit for exploratory modeling, scenario discovery, and (multi-objective) robust decision making. Environ. Model. Software 96, 239–250. https://doi.org/10.1016/j.envsoft.2017.06.054.
Kwakkel, J.H., Eker, S., Pruyt, E., 2016a. How robust is a robust policy? Comparing alternative robustness metrics for robust decision-making. In: International Series in Operations Research and Management Science. Springer, pp. 221–237. https://doi.org/10.1007/978-3-319-33121-8_10.
Kwakkel, J.H., Haasnoot, M., 2019. Supporting DMDU: a taxonomy of approaches and tools. In: Decision Making under Deep Uncertainty. Springer, pp. 355–374.
Kwakkel, J.H., Haasnoot, M., Walker, W.E., 2015. Developing dynamic adaptive policy pathways: a computer-assisted approach for developing adaptive strategies for a deeply uncertain world. Climatic Change 132, 373–386. https://doi.org/10.1007/s10584-014-1210-4.
Kwakkel, J.H., Walker, W.E., Haasnoot, M., 2016b. Coping with the wickedness of public policy problems: approaches for decision making under deep uncertainty. J. Water Resour. Plann. Manag. 142, 01816001 https://doi.org/10.1061/(ASCE)WR.1943-5452.0000626.
Kwakkel, J.H., Walker, W.E., Marchau, V.A.W.J., 2010. Classifying and communicating uncertainties in model-based policy analysis. Int. J. Technol. Pol. Manag. 10, 299. https://doi.org/10.1504/IJTPM.2010.036918.
Lempert, R.J., 2003. Shaping the Next One Hundred Years: New Methods for Quantitative, Long-Term Policy Analysis. Rand Corporation. https://doi.org/10.1016/j.techfore.2003.09.006.
Lempert, R.J., Collins, M.T., 2007. Managing the risk of uncertain threshold responses: comparison of robust, optimum, and precautionary approaches. Risk Anal. 27, 1009–1026. https://doi.org/10.1111/j.1539-6924.2007.00940.x.
Little, J.C., Hester, E.T., Elsawah, S., Filz, G.M., Sandu, A., Carey, C.C., Iwanaga, T., Jakeman, A.J., 2018. A tiered, system-of-systems modeling framework for resolving complex socio-environmental policy issues. Environ. Model. Software 112, 82–94.
Maier, H.R., Guillaume, J.H.A., van Delden, H., Riddell, G.A., Haasnoot, M., Kwakkel, J. H., 2016. An uncertain future, deep uncertainty, scenarios, robustness and adaptation: how do they fit together? Environ. Model. Software 81, 154–164. https://doi.org/10.1016/j.envsoft.2016.03.014.
Maier, H.R., Razavi, S., Kapelan, Z., Matott, L.S., Kasprzyk, J., Tolson, B.A., 2018. Introductory overview: optimization using evolutionary algorithms and other metaheuristics. Environ. Model. Software 114, 195–213.
McPhail, C., Maier, H.R., Kwakkel, J.H., Giuliani, M., Castelletti, A., Westra, S., 2018. Robustness metrics: how are they calculated, when should they Be used and why do they give different results? Earth's Futur 6, 169–191. https://doi.org/10.1002/2017EF000649.
McPhail, C., Maier, H.R., Westra, S., Kwakkel, J.H., van der Linden, L., 2020. Impact of scenario selection on robustness. Water Resour. Res. 56 (9).
Quinn, J.D., Hadjimichael, A., Reed, P.M., Steinschneider, S., 2020. Can exploratory modeling of water scarcity vulnerabilities and robustness Be scenario neutral? Earth's Futur 8, e2020EF001650.
Quinn, J.D., Reed, P.M., Giuliani, M., Castelletti, A., Oyler, J.W., Nicholas, R.E., 2018. Exploring how changing monsoonal dynamics and human pressures challenge multi reservoir management for flood protection, hydropower production, and agricultural water supply. Water Resour. Res. 54, 4638–4662.
Quinn, J.D., Reed, P.M., Keller, K., 2017. Direct policy search for robust multi-objective management of deeply uncertain socio-ecological tipping points. Environ. Model. Software 92, 125–141.
Reis, J., Shortridge, J., 2020. Impact of uncertainty parameter distribution on Robust Decision Making outcomes for climate change adaptation under deep uncertainty. Risk Anal. 40, 494–511.
Roach, T., Kapelan, Z., Ledbetter, R., Ledbetter, M., 2016. Comparison of robust optimization and info-gap methods for water resource management under deep uncertainty. J. Water Resour. Plann. Manag. 142, 04016028 https://doi.org/10.1061/(ASCE)WR.1943-5452.0000660.
Savage, L.J., 1951. The theory of statistical decision. J. Am. Stat. Assoc. 46, 55–67. https://doi.org/10.1080/01621459.1951.10500768.
Schwarz, P., 1991. The Art of the Long View: Planning for the Future in an Uncertain World. John Wiley & Sons, Chichester, England.
Shepherd, T.G., Boyd, E., Calel, R.A., Chapman, S.C., Dessai, S., Dima-West, I.M., Fowler, H.J., James, R., Maraun, D., Martius, O., 2018. Storylines: an alternative approach to representing uncertainty in physical aspects of climate change. Climatic Change 1–17.
Simon, H.A., 1956. Rational choice and the structure of the environment. Psychol. Rev. 63, 129–138. https://doi.org/10.1037/h0042769.
Singh, R., Reed, P.M., Keller, K., 2015. Many-objective robust decision making for managing an ecosystem with a deeply uncertain threshold response. Ecol. Soc. 20.
Trindade, B.C., Reed, P.M., Herman, J.D., Zeff, H.B., Characklis, G.W., 2017. Reducing regional drought vulnerabilities and multi-city robustness conflicts using many-objective optimization under deep uncertainty. Adv. Water Resour. 104, 195–209.
van der Heijden, K., 1996. Scenarios: the Art of Strategic Conversation. John Wiley & Sons.
Varum, C.A., Melo, C., 2010. Directions in scenario planning literature - a review of the past decades. Futures 42, 355–369. https://doi.org/10.1016/j.futures.2009.11.021.
Wada, Y., Vinca, A., Parkinson, S., Willaarts, B.A., Magnuszewski, P., Mochizuki, J., Mayor, B., Wang, Y., Burek, P., Byers, E., 2019. Co-designing indus water-energy-land futures. One Earth 1, 185–194.
Wald, A., 1951. Statistical Decision Functions, Nature. New York. Chapman & Hall, London. https://doi.org/10.1038/1671044b0.
Walker, W.E., Lempert, R., Kwakkel, J., 2013. Deep uncertainty. In: Encyclopedia of Operations Research and Management Science. Springer, pp. 395–402. https://doi.org/10.1007/978-1-4419-1153-7_1140.
Ward, V.L., Singh, R., Reed, P.M., Keller, K., 2015. Confronting tipping points: can multi-objective evolutionary algorithms discover pollution control tradeoffs given environmental thresholds? Environ. Model. Software 73, 27–43.
Watson, A.A., Kasprzyk, J.R., 2017. Incorporating deeply uncertain factors into the many objective search process. Environ. Model. Software 89, 159–171. https://doi.org/10.1016/j.envsoft.2016.12.001.
Weaver, C.P., Lempert, R.J., Brown, C., Hall, J.A., Revell, D., Sarewitz, D., 2013. Improving the contribution of climate model information to decision making: the value and demands of robust decision frameworks. Wiley Interdiscip. Rev. Clim. Chang. 4, 39–60.
Wright, G., Cairns, G., 2011. Scenario Thinking: Practical Approaches to the Future. Springer.
Xexakis, G., Hansmann, R., Volken, S.P., Trutnevyte, E., 2020. Models on the wrong track: model-based electricity supply scenarios in Switzerland are not aligned with the perspectives of energy experts and the public. Renew. Sustain. Energy Rev. 134, 110297.
Zeff, H.B., Kasprzyk, J.R., Herman, J.D., Reed, P.M., Characklis, G.W., 2014. Navigating financial and supply reliability tradeoffs in regional drought management portfolios. Water Resour. Res. 50, 4906–4923. https://doi.org/10.1002/2013WR015126.