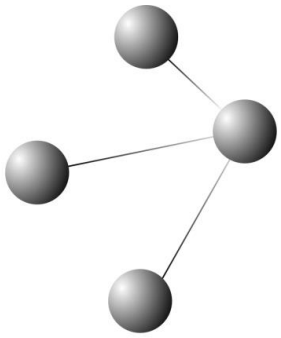
Australian Government
Department of Agriculture

cebra
Centre of Excellence for Biosecurity Risk Analysis

CEBRA Report Cover Page				
Project Title, ID & Output	Uncertainty in the RRRA model, CEBRA Project 1304B, Recommendation Report			
Project Type	Final Report			
DAFF Project Sponsor	Greg Williamson	**DAFF Project Leader/s**	Jean Chesson / Meredith Walton	
CEBRA Project Leader	Mark Burgman	**NZ MPI Collaborator**	Nil	
Project Objectives	The objective of this project is to develop a practical method for including measures of uncertainty with the RRRA system output so that decision-makers can gauge how much confidence they can place in the results. This objective contributes to the RRRA project 'to develop a decision support system that will analyse the cost and effectiveness of biosecurity controls, thereby supporting DAFF Executive to make risk based decisions on biosecurity expenditure'. The RRRA project contributes to the DAFF priority of 'continuing the transition to a fully integrated risk-based approach to managing biosecurity risk offshore, at the border and onshore'.			
Outputs	1. List of potential approaches for evaluation 2. Evaluation criteria 3. Draft report describing aim, methods, results and recommendations 4. Final report			
Original Budget	**Year 2013-14**	**Year 2014-15**	**Year 2015-16**	**Year 2016-17**
	$57,500	NA	NA	NA
Project Changes	There were no changes to scope, timing or budget.			
Research Outcomes	The report looks at an array of techniques for making decisions based on models subject to severe uncertainty. It identifies a sensitivity analysis technique that is well-suited to the RRRA model, and proposes an approach based on this technique that allows the decision maker to directly compare the uncertainty around model predictions under different investment scenarios. The approach has been developed with practicality strongly in mind, and can be directly embedded within the decision support system being developed for the RRRA project, substantially improving DAFF's ability to assess, understand and manage biosecurity risk.			
Recommendations	After surveying existing approaches to uncertainty analysis, the following recommendations were developed for incorporating and reporting on uncertainty in the RRRA decision support system: • Employ an approach to investigating uncertainty in the RRRA model based on the technique known as variance-based sensitivity analysis • Provide confidence assessments in the form of uncertainty distributions for input parameters of the model • Implement a model review process to identify parts of the model that can be treated as certain • Group parameters and take advantage of local structure wherever possible to reduce the effort (both manual and computational) needed for confidence assessments and sensitivity analyses • Treat the RRRA model as a hierarchy of models, and analyse each part of the hierarchy separately (with a summary of the analysis from lower levels feeding into higher level metamodels) • Use the results of the sensitivity analysis to characterise uncertainty around model outputs in ways that can be clearly communicated to decision makers			
Related Documents	RRRA Project Technical Manual			

CEBRA Use only	Received by:	Date:
	CEBRA / SRC Approval	Date:
	DAFF Endorsement ( ) Yes ( ) No	Date:
	Report published	Date:



BAYESIAN
INTELLIGENCE

Making Robust Decisions with a Model Subject to Severe Uncertainty

July 11, 2014

Steven Mascaro
steven.mascaro@bayesian-intelligence.com

Developed for the Department of Agriculture in conjunction with CEBRA

Bayesian Intelligence
ABN: 33 128 355 386
2/21 The Parade, Clarinda, 3168 Victoria Australia
Tel: 0425 801 277 Email: bi@bayesian-intelligence.com

Contents

1 Introduction

The Risk-Return Resource Allocation (RRRA) project provides a framework for the department to make resource allocation decisions that account for biosecurity risk. The project team has developed a model for estimating risk and cost given specified biosecurity investment scenarios. The model has been applied to over 60 organisms arriving by over 50 different pathways. The RRRA project contributes to the DAFF priorities of 'continuing the transition to a fully integrated risk-based approach to managing biosecurity risk offshore, at the border and onshore' and 'to develop a decision support system that will analyze the cost and effectiveness of biosecurity controls, thereby supporting DAFF Executive to make risk based decisions on biosecurity expenditure'.

The basic structure of the model is complete. It allows users to compare output from alternative scenarios to determine whether a change in investment would be desirable. Currently, the model calculates expected (mean) values, in accordance with classical decision theory. Sub-models are often represented as Bayesian networks (BNs), and take advantage of the built-in decision network features available in off-the-shelf BN software packages. However, sub-models need not be BNs and can be represented in any appropriate form.[1] The model estimates the number of organisms of quarantine concern that are expected to arrive on each pathway. Risk is defined as the probability that an organism enters, establishes and spreads in Australia multiplied by the consequences (or impact) of its spread. Each pathway application of the RRRA model includes many parameters ($\gg$100). As in all risk analyses, model structures and the values of individual parameters are uncertain.

Risk return resource allocation implies that decision makers consider both the cost of investment and the change in risk, when deciding where to invest in biosecurity. The project is building a system to explore the relationship between investment and risk.

The impact of uncertainty on decisions depends on the context of the decision and the decision maker's attitude to the risk. When deciding to change investment strategies, a decision maker needs to be confident that the change in risk will be in the direction desired and of an acceptable magnitude. Information on uncertainty will assist the decision maker to assess the likelihood of unacceptable outcomes, and to devise robust strategies that deliver acceptable outcomes, even when uncertainty is severe. Measures of uncertainty in the model output will allow decision makers to gauge how much confidence they can place in the results.

The objective here is to identify and evaluate methods for characterizing and reporting uncertainty in the stochastic RRRA model. Uncertainties arise from:

- the different ways in which operational details are simplified in the models functions and assumptions,

[1] Of course, BNs are capable of representing any computable function, but unfortunately BN software packages don't always provide the most efficient tools to represent those functions.

- the natural variation that occurs in input parameters such as the volume of trade, composition and country of origin of trade, prevalences of pests and diseases in consignments, and so on, and
- lack of knowledge about these parameters.

In this document, we will outline methods for dealing with all of these sources of uncertainty, and will scope the potential for alternative ways of analyzing them to support decision making. We begin with a survey of current approaches to making decisions with models subject to severe uncertainty in Sections 2-5, and then describe an approach tailored to the RRRA system in Section 6. Finally, we present a detailed example demonstrating the use of the techniques in the RRRA system in Section 7.

2 Decision making under uncertainty

2.1 Classical rational decision making

It is worth beginning with a brief description of the most common procedure (often called a *decision rule*) for making decisions under uncertainty, based on the expected utility model [Schoemaker, 1982]. This will be particularly useful as some of the methods described later deviate from or extend this approach.

In an expected utility framework, it is assumed that a rational decision maker is faced with a set of actions A, which, combined with current states of the world S, will lead probabilistically to a set of outcomes O. Thus, for each action $a_i \in A$, state of the world $s_j \in S$ and outcome $o_k \in O$, we can define the conditional probability:

$$P(o_k|a_i, s_j)$$

In addition, we take it that each outcome can be assigned a value or utility, $U(o_k)$. To calculate the expected value (also called the expected utility or the expected reward) of the action a_i, we weight each outcome's value by the probability of that outcome occurring, given the state of the world is s_j and we have performed a_i:

$$EV(a_i|s_j) = \sum_k U(o_k)P(o_k|a_i, s_j)$$

The classical decision rule then instructs us to choose the action that yields the highest expected value:

$$\underset{a_i}{argmax}\, EV(a_i|s_j)$$

There is a good reason for following the rule of maximising the expected value in our decisions. Assuming our decisions are independent, it leads to the maximum possible cumulative utility over time. By way of brief explanation, if the situation s_j arises in n cases (where n is very large), then, by the law

of large numbers, the average return for performing a_i across those cases will be the expected value, and the cumulative value will be $EV \times n$ — regardless of the utility we happen to receive by chance in each case (and regardless of how positive or negative those case-specific utilities). The law of large numbers means this will hold true even if s_j is unique in each case.

The classical approach has been criticised on many grounds, leading to variations in the expected utility model and the proposal of alternative systems such as prospect theory (see [Schoemaker, 1982] for a summary). We will not explore such variations here. More importantly, while the classical approach clearly handles some kinds of uncertainty (by way of its conditional probabilities), it seems to ignore other kinds of uncertainty by making several assumptions about our knowledge. These assumptions include that we know (or can assign a reasonably certain degree of belief to) $P(o_k|a_i, s_j)$; that we know $U(o_k)$; or even that we know the set of all o_k. Such assumptions can crucially affect our decision-making, and are indeed common targets for sensitivity and uncertainty analyses.

2.2 Uncertainty

2.2.1 Levels of uncertainty

Uncertainty is often divided into two kinds: measurable and unmeasurable. The most famous proponent of this distinction is Knight [1921], and unmeasurable uncertainty is often called Knightian uncertainty. Ben-Haim [2006] motivates the development of his info-gap theory (see below) on the basis of Knightian uncertainty, describing it as *severe uncertainty*.

Kwakkel et al. [2010] go further, and define four levels of uncertainty. Their categorisations are reproduced in Table 1. According to Kwakkel et al.'s definitions, uncertainty that can be quantified by probabilities is equivalent to Level 1 (or shallow) uncertainty, while Knightian uncertainty corresponds to Levels 2-4 (particularly 3-4). Many people (including Bayesians) may question whether Levels 2-4 define genuinely different kinds of uncertainty, or just categories that are of practical interest (i.e. representing cases in which uncertainty is increasingly hard to quantify, rather than impossible).

2.2.2 Second-order probabilities

An important related idea is that of second-order probabilities, or probabilities over probabilities (sometimes called hierarchical priors). For instance, we can be quite certain that the probability of an ordinary coin turning up heads is 0.5. However, if we are told that the coin might be biased, then we are no longer so sure of the probability. Nonetheless, we may still have a probability distribution over the probability of heads — if all we are told is that the coin is biased, we don't expect it to always come up heads or tails. Indeed, we expect the coin will only be modestly biased, in order to avoid detection.

The concept of second order probabilities has raised two difficult questions: 1) How should they be handled? and 2) Are higher order probabilities possible,

Level of Uncertainty	Description	Examples
Level 1 (shallow uncertainty)	Being able to enumerate multiple alternatives and being able to provide probabilities (subjective or objective)	Being able to enumerate multiple possible futures or alternative model structures, and specify their probability of occurring
Level 2 (medium uncertainty)	Being able to enumerate multiple alternatives and being able to rank order the alternatives in terms of perceived likelihood. However, how much more likely or unlikely one alternative is compared to another cannot be specified	Being able to enumerate multiple possible futures or alternative model structures, and being able to judge them in terms of perceived likelihood.
Level 3 (deep uncertainty)	Being able to enumerate multiple alternatives without being able to rank order the alternatives in terms of how likely or plausible they are judged to be	Being able to enumerate multiple possible futures or specify multiple alternative model structures, without being able to specify their likelihood
Level 4 (recognized ignorance)	Being unable to enumerate multiple alternatives, while admitting the possibility of being surprised	Keeping open the possibility of being wrong or of being surprised

Table 1: A reproduction of the levels of uncertainty defined by Kwakkel et al. [2010].

ad infinitum? Good responses have been given on both counts. Pearl [1987] has effectively argued that no new specialised logical machinery is required to handle such probabilities; conditional probabilities are perfectly suitable. Similarly, the concern about higher order probabilities is not so great; hierarchical priors are naturally robust at higher levels, meaning they can be specified with much less precision [Berger, 1990]. This line of reasoning can be seen as an attack on the distinction between measurable and unmeasurable uncertainties, and gives pause for thought as to whether such a distinction may really be needed.

2.2.3 Uncertainties in Bayesian networks

Bayesian networks (BNs) were formulated as models for performing reasoning under uncertainty. The uncertainty that BNs work well with is measurable uncertainty (Level 1 in Kwakkel et al.'s hierarchy). However, uncertainty may still exist, affecting not only our choice of parameters (conditional probabilities and utilities), but also our choice of state spaces, conditional relations and variables. The number of possible ways any given BN can be altered — meaning the number of possible ways the model can be wrong — is enormous. In theory, this is no different to other forms of modelling. Any kind of modelling can only reduce the number of parameters by making assumptions; BNs just make such assumptions explicit. Nonetheless, this can make a deeper uncertainty analysis of a BN model more difficult in practice.

Second and higher order probabilities are particularly relevant when we talk about uncertainties in the parameterisation (and to some extent, structure) of Bayesian networks and Bayesian models more generally. Indeed, most Bayesian approaches to sensitivity analysis explicitly aim to model these higher order probability distributions. This is what we will aim to do later as well, but with an eye on keeping the effort manageable given the large number of parameters involved.

3 Sensitivity analysis

Sensitivity analysis is a common technique used in validating models and simulations where there is uncertainty about the appropriate values for input parameters. Specifically, sensitivity analysis is an investigation into how the output of a model varies in response to variation in the input parameters [Saltelli et al., 2000], particularly where the input variation captures uncertainty of interest to the modeller.

3.1 Local and global sensitivity analysis

There are two main kinds of sensitivity analysis: local and global [Wainwright et al., 2013, Saltelli et al., 2008]. Local methods begin with an initial (or reference) set of input parameters, and restrict their exploration to the space around those parameters, often using derivative or difference based methods. By contrast, global methods aim to analyse the full domain of inputs (or at least the

full domain of a subset of inputs), looking at how overall variation and interactions amongst inputs affects the uncertainty in outputs. Local methods are well suited to cases in which one already has a reasonable estimate of what the input parameters should be, and wishes to assess how well the estimate holds up under small changes. Global methods, on the other hand, are better suited to understanding the features of the input parameter space that affect the model's output. Ultimately, though, both forms of sensitivity analysis can be adapted for either purpose [Wainwright et al., 2013].

Sensitivity analysis is also often divided on the basis of whether the impact of one variable is investigated at a time (called one-at-a-time or OAT) or whether the impact of variables are analysed in combination (called variation-in-combination or VIC) [French, 2003], though this classification is of mostly technical interest.

3.2 Uncertainty analysis

Sensitivity analysis is sometimes distinguished from what is called uncertainty analysis. Uncertainty analysis typically refers to an analysis of how *uncertainties* in inputs propagate to *uncertainties* in outputs [Iman and Helton, 1988] (as opposed to an analysis of how changes in inputs produce changes in outputs). An alternative, mostly operational, definition is that it is the analysis of how correlated variations in multiple inputs affect output (e.g., Bedford and Cooke [2001]). Both definitions fit comfortably within the domain of sensitivity analysis. This is particularly true when we take the advice of French [2003] and consider the purpose of such analyses, which is about understanding uncertainty in both cases. Nonetheless, if a distinction is to be made, we take the term as referring to an analysis of the full range of uncertainties (not just variations in parameters) that can affect our models and, ultimately, decisions.

3.3 Common approaches to sampling the input space

If a model can be described in the form of a closed-form expression, many analytical techniques are available for analysing the propagation of changes in inputs to changes in outputs directly, often using a derivative based approach. For more complex models, some form of simulation is required, such as Monte Carlo simulation [Saltelli et al., 2000]. The approach used (and often, as shorthand, the set of input values created using the approach) is called the experimental design of the analysis. An experimental design that uses Monte Carlo simulation involves the random sampling of values for parameters and assessing the impact on the output. In the case of local sensitivity analysis, this is restricted to some neighbourhood around a set of reference parameters. For global analysis, the entire input range may be sampled.

An improvement to a Monte Carlo design for such analysis (whether local or global) involves the use of a Latin hypercube. A Latin hypercube partitions each parameter into equal probability regions, and ensures that any new sample that is generated is located in a region not already covered by previous samples

(unless all regions are covered, in which case the sampling process begins anew). While over the long run, both Monte Carlo and Latin hypercube approaches converge to the same sampling coverage of the input space, a Latin hypercube design will more faithfully represent the variation in the parameters than a Monte Carlo design for the same number of samples, and is thus far more efficient. Other approaches that can improve the efficiency of the sampling process still further include Sobol sequences, Faure sequences and Niederreiter sequences [Sobol', 1998].

3.4 Implications for decision making

The results of a sensitivity analysis do not affect the decision maker's decision rule. Rather, the results are provided to the decision maker as additional information that they must then weigh and integrate with all the other information they have available. In some cases, this may provide the decision maker with confidence in using the model output for their decisions. In other cases, the decision maker might reject the model or ask for it to be refined, in which case the sensitivity analysis provides a guide to which parts of the model would be most worth improving. Ideally, the sensitivity analysis would provide an indication of the likelihood that a model's results apply to the system being modelled, but this seems to receive little coverage in the literature. As such, since the decision maker must handle the information from a sensitivity analysis manually, it is important that the results of a sensitivity analysis can be communicated well. Fortunately, this is frequently the case, and we will see some examples in Section 5.

4 Approaches to uncertainty analysis

4.1 Variance-based sensitivity analysis

Sobol' [2001] proposed a global sensitivity analysis approach that attributes variance in the output of a model to variances in the input parameters of the model or combinations thereof. The method, generally called variance-based sensitivity analysis (VBSA), is capable of giving *sensitivity indices* for each parameter and for interactions between all subsets of parameters. These sensitivity indices are all non-negative and sum to 1, and give the *proportion* of variation in the output that is due to a parameter (or combination of parameters). The technique can therefore be used to rank parameters by importance, identify unimportant parameters and identify the strength of interactions between parameters.

The approach begins by treating the model as a black box that can be represented by a function $f(X) = Y$, where X is a vector of n i.i.d random variables, X_i, each ranging uniformly over the interval $[0, 1]$, and Y is a scalar output. $f(X)$ can be decomposed into the following form:

$$f(X) = f_0 + \sum_i f_i(X_i) + \sum_{i<j} f_{ij}(X_i, X_j) + ... + f_{12...n}(X_1, X_2, ..., X_n)$$

which is called Hoeffding decomposition [Saltelli et al., 2010]. When the value of the integration over the interval $[0, 1]$ for each term in this decomposition is equal to 0, the terms are orthogonal and can be expressed as integrals of $f(X)$:

$$f_0 = \int f(X)dX = E(Y)$$
$$f_i(X_i) = \int f(X) \prod_{k \neq i} dX_k = E[Y|X_i] - f_0$$
$$f_i j(X_i, X_j) = \int f(X) \prod_{k \neq i,j} dX_k = E[Y|X_i, X_j] - f_0 - f_i - f_j$$

Hence, f_0 represents the mean output value for the model, $f_i(X_i)$ represents the effect on the output of varying X_i alone and $f_{ij}(X_i, X_j)$ represents the effect on the output of varying X_i and X_j in combination.

Given this decomposition, and assuming $f(X)$ is square integrable, we can calculate *variances* for each of the terms by squaring and integrating (over $[0, 1]$) the decomposition:

$$\int f^2(X)dx = f_0^2 + \sum_{i<j} f_{ij}^2(X_i, X_j) + ... + f_{12...n}^2(X_1, X_2, ..., X_n)$$

which we can rearrange as:

$$\int f^2(X)dx - f_0^2 = \sum_{i<j} f_{ij}^2(X_i, X_j) + ... + f_{12...n}^2(X_1, X_2, ..., X_n)$$

Note that the left hand side of this last equation can be restated $E[Y^2] - E[Y]^2$, which is the formula for the variance of Y, $\mathrm{Var}(Y)$. Similar transformations apply to the terms on the right, giving:

$$\mathrm{Var}(Y) = \sum_{i=1} \mathrm{Var}_{X_i}(f_i(X_j)) + \sum_i \sum_{j>i} \mathrm{Var}_{X_i,X_j}(f_{ij}(X_i, X_j)) + ... + \mathrm{Var}_{X_i,...,X_n}(f_i j(X_i, ..., X_n))$$

or, substituting V for $\mathrm{Var}(Y)$, V_i for $\mathrm{Var}_{X_i}(f_i(X_j))$, and V_{ij} for $\mathrm{Var}_{X_i,X_j}(f_{ij}(X_i, X_j))$, etc.:

$$V = \sum_{i=1} V_i + \sum_i \sum_{j>i} V_{ij} + ... + V_{i..n}$$

Dividing the above by V tells us the proportion of output variation that is produced by each parameter, and each parameter combination:

$$V/V = \sum_{i=1} V_i/V + \sum_i \sum_{j>i} V_{ij}/V + ... + S_{i..n}/V$$
$$1 = \sum_{i=1} S_i + \sum_i \sum_{j>i} S_{ij} + ... + S_{i..n}$$

Where the $S_\bullet$ are called sensitivity indices. Both the sensitivity indices and variances can be used directly by modellers to identify parameters that would benefit from greater accuracy and ultimately by decision makers to examine the robustness of the model outputs.

VBSA can be performed on analytically tractable functions and numerical models equally well. Monte Carlo approaches, as well as quasi-Monte Carlo approaches (i.e., Latin hypercubes or Sobol sequences) that can be used to approximately calculate these sensitivity indices for numerical models are well-established [Saltelli et al., 2010, Sobol', 2001], although can be very computationally expensive depending on the speed of the model.

4.2 Value of information

Value of information (VoI) [Raiffa and Schlaifer, 1961] approaches uncertainty from a different angle. It starts from the idea that if a classically rational decision maker has a lack of information about some decision, and she is in a position to acquire information that fills this gap, she will be willing to pay up to as much as the new information is expected to yield. Applications of VoI include fishery management modelling [Mäntyniemi et al., 2009], evaluation of medical equipment [Oostenbrink et al., 2008] and assisting in the allocation of research funding [Karnon, 2002]. (See [Yokota and Thompson, 2004] for a review of VoI and its applications.)

Mäntyniemi et al. [2009] provide a good example to illustrate the idea. Suppose a population of 1000 fish moves between two habitats — offshore O and estuary E — such that at any given time 80% of the fish are in one of the habitats. A fisher has two choices: fish offshore, a_O, or fish in the estuary, a_E. In either case, she will catch all of the fish present. Suppose she believes there is a 0.7 probability that 80% of fish are in the estuary:

$$\begin{aligned} P(F_E) &= 0.7 \\ P(F_O) &= 0.3 \end{aligned}$$

If $C(F_{H1}, a_{H2})$ is her catch given that she chooses to fish in habitat $H2$ when the majority of fish are in $H1$, her expected values for each choice is then as follows:

$$\begin{aligned} EV(a_E) &= P(F_E)C(F_E, a_E) + P(F_O)C(F_O, a_E) \\ &= 0.7 \times 800 + 0.3 \times 200 = 620 \\ EV(a_O) &= P(F_E)C(F_E, a_O) + P(F_O)C(F_O, a_O) \\ &= 0.7 \times 200 + 0.3 \times 800 = 380 \end{aligned}$$

Suppose the fisher can choose to pay for perfect information about where the fish are located. Such information will consist of two possible messages: M_O, the majority of fish are offshore, with perfect information implying $P(F_O|M_O) = 1$; or M_E, the majority of fish are in the estuary, with perfect information implying $P(F_E|M_E) = 1$.

The maximum expected value of the decision given the new perfect information is then the maximum possible catch of 800 (unsurprisingly, since the information is perfect and free):

$$\begin{aligned} EV(a_E|M_E) &= P(F_E|M_E)C(F_E, a_E) + P(F_O|M_E)C(F_O, a_E) \\ &= 1 \times 800 + 0 \times 200 = 800 \\ EV(a_O|M_O) &= P(F_E|M_O)C(F_E, a_O) + P(F_O|M_O)C(F_O, a_O) \\ &= 0 \times 200 + 1 \times 800 = 800 \end{aligned}$$

However, the *change* in expected value for each choice is the expected value of the choice she makes given the new information (which is now always 800), minus the expected value of the choice *she would have made* before seeing the new information:

$$\begin{aligned} \Delta_{M_E} &= EV_{\text{new}}(a_E|M_E) - EV_{\text{old}}(a_E|M_E) = 800 - 800 = 0 \\ \Delta_{M_O} &= EV_{\text{new}}(a_O|M_O) - EV_{\text{old}}(a_E|M_O) = 800 - 200 = 600 \end{aligned}$$

(Seeing M_E would not change the fisher's choice, hence $\Delta_{M_E} = 0$.) We can then calculate the expected value of the new information to the fisher if we have probabilities for the messages. We have such probability estimates — they are simply the fisher's prior probabilities for the majority of fish being offshore or in the estuary:

$$\begin{aligned} P(M_E) &= P(F_E) = 0.7 \\ P(M_O) &= P(F_O) = 0.3 \end{aligned}$$

Giving the expected value of new information in this case to be:

$$\begin{aligned} EV(M) &= P(M_E)\Delta_{M_E} + P(M_O)\Delta_{M_O} \\ &= 0.7 \times 0 + 0.3 \times 600 = 180 \end{aligned}$$

Thus, the fisher would be willing to pay up to the equivalent value of 180 fish in order to acquire the new information.

In addition to the expected value of perfect information (EVPI), there are also techniques for calculating the expected value of partial information [Yokota and Thompson, 2004]. This is sometimes abbreviated as EVPXI, the expected value of perfect X information, where X represents an uncertain model input (and, by implication, other uncertainties in the model are set aside). EVPXI can be calculated as the expected value of the optimal action given that we know X exactly, minus the expected value of the optimal action given only prior information about X.

It is easy to see how VoI can be used to assist modellers and decision makers. The most natural application is in identifying which input parameters would provide the most benefit from increased accuracy (assuming one has prior distributions over the model inputs). VoI can also be applied to dispelling deeper uncertainties, so long as prior distributions can be defined over them. For instance, one can define a model space, and a distribution over that space, in

order to determine the value of learning a better model, as done by Williams et al. [2011]. VoI can also be used to screen out unimportant parameters. Felli and Hazen [1998] reanalyse several sensitivity analyses from the literature that used traditional sensitivity techniques and compare them to an EVPI approach, finding that EVPI is able to screen out unimportant parameters much more effectively than other approaches.

While VoI essentially comes equipped with its own decision rule (i.e. that of classical decision theory), the rule is applied only to whether gathering new information is worthwhile. It is of no help to the decision maker if she can't (or she is unwilling to) gather new information. This is despite the fact that VoI provides information about the relative importance of uncertainties surrounding inputs on a model's output, regardless of whether one is able to reduce that uncertainty in reality. Such information can inform a decision maker's assessment of the reliability of a model, and her assessment of the choices based upon it. Indeed, it would be interesting to see if such information could be used within a new decision rule. But since no way of obtaining such information currently exists, it seems that VoI is best suited to screening parameters for the present.

4.3 Info-gap theory

Info-gap theory, originally developed by Ben-Haim [2006], is an attempt at dealing with severe uncertainty in decision making. The gap referred to is that gap between what is known now (i.e. the information we have now) and what could be known (i.e. the information we could have). In the language of info-gap theory, severe uncertainty refers to uncertainty that can't be quantified by a probability. Info-gap theory has been applied to the surveillance of invasive species [Davidovitch et al., 2009], handling uncertainty in conservation management [Regan et al., 2005] and calculating the parameters for hypothesis tests [Fox et al., 2007], amongst many other applications.

The info-gap approach is fundamentally a sensitivity analysis approach (possibly, a local sensitivity analysis approach; see [Sniedovich, 2012]), however it goes one significant step further. Whereas sensitivity analysis does not impact on one's decision rule, info-gap proposes its own method in place of the classical rational decision rule. In particular, it suggests that an action is to be chosen on the basis of how robustly it can meet desired performance criteria, rather than on its expected value.

More formally, suppose that $\tilde{f}$ is a function that models our best guess as to how our actions, a, input parameters x, and states of the world, s, lead to outcomes, o:

$$o_k = \tilde{f}(a_i, x, s)$$

(For simplicity, we will assume that o_k are numerical, but this is not essential.) Since we assume that this model is subject to severe uncertainty, there are many alternative functions, f, that may be better models. Info-gap focusses on those functions that produce outputs around our best guess function's outputs. It

defines a critical parameter, α, that defines how far the alternative functions can be. Specifically, we define the set of functions, $\mathcal{F}$, such that:

$$\mathcal{F}(\alpha, \tilde{f}, a_i) = \{f(a_i, x, s) : |f(a_i, x, s) - \tilde{f}(a_i, x, s)| \leq \alpha\}$$

The parameter α is called the *uncertainty parameter* or the *horizon of uncertainty*. We then define a reward (or utility) function on the outputs of our model:

$$R(f(a_i, x, s))$$

and a minimum desired reward:

$$R_{\min}$$

For a given action, we then find the maximum α, which we denote $\hat{\alpha}$, that identifies a set of functions, where the minimum reward for those functions is greater than $R_{\min}$:

$$\hat{\alpha}(a_i, R_{\min}) = \max\{\alpha : (\min_{f \in \mathcal{U}(\alpha, \tilde{f}, a_i)} R(f(a_i, x, s)) \geq R_{\min})\}$$

In place of choosing the action with the maximum expected value, as per classical rationality, we choose the action that has the greatest $\hat{\alpha}$:

$$\hat{a}_i = \underset{a_i \in a}{argmax}\, \hat{\alpha}(a_i, R_{\min})$$

That is to say, we choose the action that will achieve (at least) the minimum reward under the widest range of uncertainty.

Two major criticisms have been levelled at info-gap theory (for the discussion, see [Sniedovich, 2012, Burgman and Regan, 2014, Sniedovich, 2014]). The first is that the decision rule it proposes is not new — namely, it is a form of maximin optimisation. The second is that the claim of being able to handle *severe* uncertainty is false, given that it is ultimately based on a local robustness approach. Whether or not the criticisms are correct, info-gap's inclusion of a sensitivity analysis into a clear decision rule still seems novel and worthwhile.

One additional concern that might be raised with info-gap theory is that $\hat{\alpha}$ defines the equivalent of an unweighted volume of uncertainty. When making use of info-gap's decision rule, we are required to choose that action which yields the largest such volume. However, each point in the model space is given equal weight in this measure — which is equivalent to assigning a uniform distribution over the space of all possible models. As Ben-Haim himself well explains (Chapter 2, [Ben-Haim, 2006]), assuming the uniform is not always benign, depending on our model representations, and can lead to contradictions if our representation is changed in semantic-preserving ways. Thus, it seems to be a dangerous choice of prior if we have good evidence that the space is not uniformly probable — and this is almost always the case (recall even the simple example of the biased coin from Section 2.2.2). While Ben-Haim eschews the ability to quantify such higher order probabilities (classifying them as unmeasurable uncertainties),

the info-gap approach seems to have quantified things uniformly by default. A simple remedy would be to make this an explicit part of the approach, so that the distribution over models can be changed from uniform when appropriate.

4.4 Robust Bayesian analysis

Robust Bayesian analysis [Berger, 1990] is a form of sensitivity analysis for Bayesian models (though not necessarily Bayesian networks). It aims to characterise the uncertainty in the input parameters in terms of a class of priors, with a prior distribution over the class (i.e. higher order probabilities), and assesses the impact of this uncertainty on the posterior of some variable of interest. The class of priors need not be specified with any great accuracy, and loose, rather than exact, shapes for the distribution can be given. Berger [1990] gives the example of a class of priors defined by a median, quartiles and the statement that it is symmetric and unimodal, all of which an expert can reasonably be expected to provide in many circumstances. Pericchi [1998] gives several further examples of classes of priors that can often be easily specified.

While a promising approach, Berger et al. [2006] note that it has run into severe limitations when used in practice — namely that prior classes that are easy to work with result in posterior intervals that are too wide to be of practical use — and so may be premature as a practical technique.

4.5 Metamodelling

One approach that is becoming increasingly common in the analysis of particularly complex computer models is to create a "metamodel" that approximates the more complex model [Conti and O'Hagan, 2010]. Typically, this is done whenever the complex computer model runs too slowly to perform the large number of iterations required to perform a full global (or local) sensitivity analysis. However it has several further advantages. For example, as French [2003] points out, it can be used to assess whether the process represented by the original complex model can effectively be described by a simpler model, allowing one to identify factors that are not relevant to the original model. Furthermore, simple approximations may be easier to work with for a range of purposes, including analysis, efficient communication with decision makers and stakeholders, comparison with other models, and so forth. Indeed, arguably the meta-models need not be simpler in any sense than the original models; they merely need to be more convenient for the purpose at hand. From this perspective, the decomposition used in VBSA is a metamodel. In most cases the decomposed model remains theoretical, but that is not the case in an approach called high dimensional model representation (HDMR) [Li et al., 2002].

Metamodelling approaches often treat the model as a black box function that takes a vector of inputs X and returns an output $f(X) = Y$ (in the same manner as VBSA). Simpler models can then be defined manually (e.g., Kennedy and O'Hagan, 2000), derived analytically using simplifying assumptions, fitted to a model using regression on sets of input-output pairs yielded by the complex

model (e.g., Conti and O'Hagan, 2010) or learned from the same input-output pairs using machine learning techniques.

When the metamodel is kept simple, either the model or its key results can be passed on to decision makers directly, as is the case with the sensitivity coefficients in a regression, or indeed the sensitivity indices from VBSA. While there is no clear way to explicitly factor this into a decision rule, such information is clearly valuable in assessing the uncertainty of the model.

5 Reporting on uncertainty

Once we have a means of assessing the uncertainty in our model across the choices we have available, we need to represent these uncertainties to decision makers. This section briefly covers some of the approaches that have been used to achieve this.

5.1 General approaches

Confidence intervals, or their Bayesian equivalents, credible intervals, are commonly used to depict the uncertainty in an estimate. They have the advantage of being broadly familiar and easily represented graphically. (One should keep in mind that frequentist confidence intervals can be misleading as to their meaning and credible intervals should be preferred where possible.) Box plots containing quantile information on the uncertainty are also commonly used, such as the plot in Figure 1 (from Helton et al. [2006], who provide further graphical examples of displays of uncertainty).

5.2 Sensitivity analysis

The results of a sensitivity analysis can be depicted using many standard techniques. Sensitivity coefficients and sensitivity indices are effective numerical means of reporting the results of a sensitivity analysis, and can be reported graphically in the form of tornado plots. OAT sensitivity analyses have particularly straightforward graphical representations. These can be plotted using scatter plots or confidence intervals and box plots (if the exact co-variation with the input parameter is not of interest). The results of a VIC analysis is, of course, more complicated to represent graphically given the higher number of dimensions, however approaches such as 3 dimensional plots, parallel coordinate plots and glyph plots can assist.

5.3 VBSA

Saltelli et al. [1998] provide a good approach to visualisation for their case study, in which they performed a VBSA of a nuclear waste disposal model with 12 input parameters. They group the input parameters into categories that would be of interest to the decision maker. Figure 2a shows 3 groupings

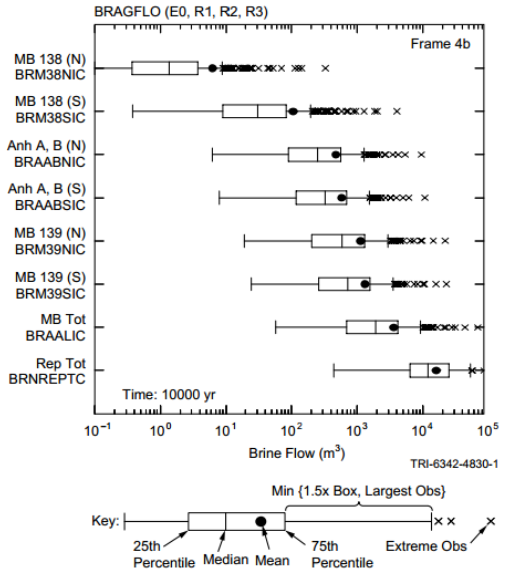
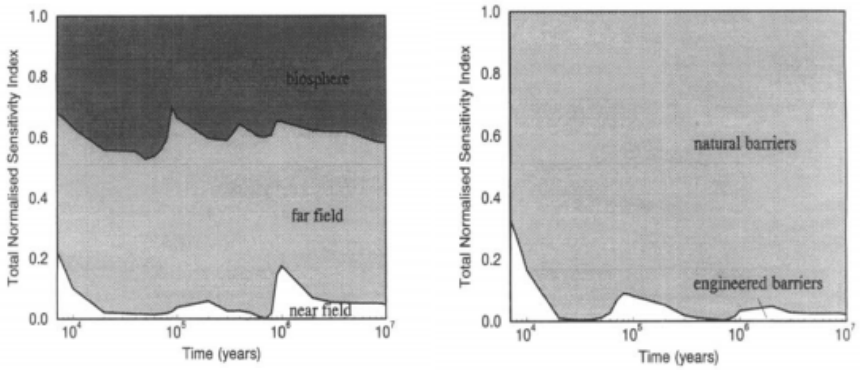


Figure 1: An example box plot. (From Helton et al. [2006].)

(biosphere factors, far field factors and near field factors), while Figure 2b shows an alternative set of 2 groupings (natural barrier factors and engineered barrier factors). (Both figures show the sensitivity of the model output to these different groups of factors over time.) These figures very effectively communicate the relative importance of the different categories of variables.



(a) (a)

Figure 2: The sensitivity of a nuclear waste disposal model to 12 input parameters, grouped into (a) 3 categories focussed on field and (b) 2 categories focussed on barriers.

VBSA approaches also provide a very natural approach to visualising the uncertainty around model outputs, since characterising the input uncertainty using distributions that represent the input variation accurately will produce outputs that also vary accurately. Thus, output distributions can be displayed directly, and we will make substantial use of this technique in our own analysis.

5.4 Info-gap

In info-gap theory, the trade-off between the horizon of uncertainty and the value function lends itself to a simple and clear approach to visualising the impact of uncertainty. An example can be found in Regan et al. [2005], and is shown in Figure 3. This figure shows the result of evaluating three different management strategies for protecting the Sumatran rhinoceros (using a decision tree model based loosely on Maguire et al. [1987]). We can clearly see that as the horizon of uncertainty increases (i.e. as the amount of uncertainty allowed in the model increases), the minimum expected utility for each strategy falls away. We can also see that for small horizons, captive breeding is the best strategy (down to an expected value of 0.12), but as the horizon increases, the dominant strategy is clearly the development of a new reserve. Hence, information about the uncertainty in the model can be communicated very clearly to the decision maker — even in the absence of a pre-determined minimum desired expected utility.

6 Analysing uncertainty in the RRRA model

6.1 Overview of the model

Before describing the proposed approach to analysing uncertainty in the RRRA model, it will be helpful to provide a more detailed overview of the system as it currently operates. The RRRA model applies to over 60 organisms that may enter the country via over 50 different pathways, the bulk of these pathways (approximately 70%) being forms of cargo. In the RRRA system, each of these pathways is represented by a model, called an *entry model*, which handles all of the organisms that may enter via that pathway. In principle, an entry model can be any kind of computational function that takes management actions as inputs, and produces probabilities of organism entry as outputs (see Figure 4). At present, all entry models are implemented as BNs, although in many cases, the parameters for the BNs are calculated or derived from external programs and databases. An example prototype entry model for containers (not including any cargo the container might carry) can be seen in Figure 5.

After entry, *post-entry models* are used to determine whether an organism will establish and spread. Often, post-entry models have the same structure, but are parameterised differently for each organism. The current prototype for the post-entry model for aquatic organisms is shown in Figure 6.

For each organism on a given pathway, the entry model is coupled to the

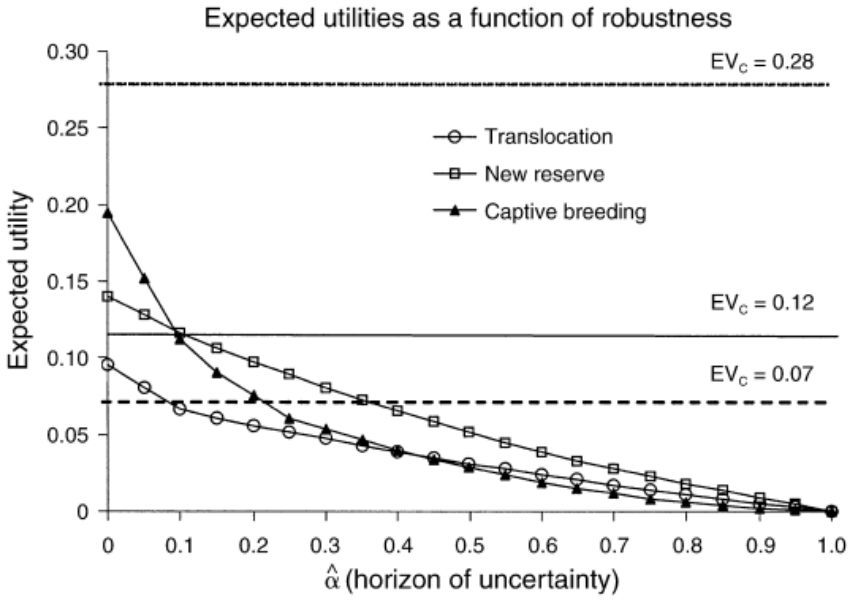


Figure 3: The minimum expected utility for increasing horizons of uncertainty, for 3 different strategies.

organism's post-entry model to give the expected number of *spread events* given a set of management actions. The expected number of spread events is the expected number of times an organism will enter the country, establish and spread. The total expected number of spread events for that organism is obtained by summing across each entry pathway. The system then calculates the expected consequence (or risk) of the organism allowing for the possibility that the consequence of a second or subsequent spread event is not necessarily the same as the first, to give an estimate of the impacts per organism under a given management scenario. The estimated impact is broken into 6 different categories: the impact to animals, the environment, infrastructure, health and society, as well as the financial impact to industry. A high level algorithm for this process is given in Algorithm 1.

6.2 Comparing management strategies with the RRRA model

The purpose of the RRRA model is to allow comparisons of the impact of different management strategies. For a given management strategy, the RRRA model calculates the expected impact on the 6 different categories of interest as described above. One way to compare strategies, therefore, is to determine which strategy has the lower expected impact, adjusting for the cost of each

Algorithm 1 High-level algorithm for the overall RRRA model

Require: EMs ← Set of entry models
Require: PMs ← Set of post-entry models
Require: A ← Set of management actions

for each organism **do**
    for each pathway **do**
        EM ← Entry model for pathway
        T ← Number of transmission units for pathway

        ▷ *Given EM and A, calculate probability of organism entering on one transmission unit*
        EP ← Probability of organism entering on one transmission unit

        ▷ *Expected number of organism entries on this pathway*
        $N_p \leftarrow T \times EP$
    end for

    ▷ *Expected number of organism entries via any pathway*
    $N_m \leftarrow \sum_p N_p$

    PM ← Post-entry model for organism

    ▷ *Given PM and A, calculate probability of spread event for organism*
    $S_m \leftarrow$ Calculate post-entry spread probability for organism

    ▷ *Expected number of spread events for this organism*
    $ES_m \leftarrow N_m \times S_m$
    for each consequence category i **do**
        ▷ *Calculate expected impact for category i. d is a diminishing function*
        $EI_{i,m} \leftarrow$ Cost per spread event $\times\ d(ES_m)$
    end for
end for

for each consequence category i **do**
    ▷ *Calculate total expected impact for category i over all organisms*
    $EI_i \leftarrow \sum_m EI_{i,m}$
end for

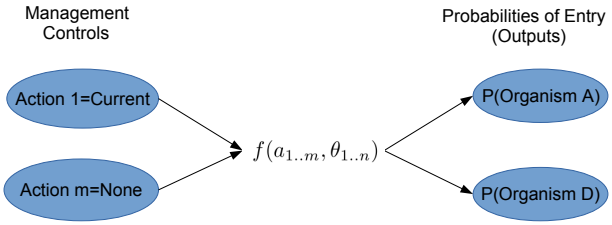


Figure 4: Entry model template. f is the model, $a_{1..n}$ the actions, and $\theta_{1..n}$ the parameters of the model.

strategy. In the case of the financial impact to industry, the expected value is in dollars, as is the cost of the strategy, hence the comparison is potentially quite simple. If:

$$EV(S_1) - EV(S_2) > 0$$

then strategy S_1 is to be preferred[2], where:

$$EV(S_i) = E(\text{IndustryCost}(S_i)) - E(\text{StrategyCost}(S_i))$$

Unfortunately, things aren't quite so simple. First, a proper assessment would need to be based on *expected utility*, rather than the expected dollar amount. Even in the restricted case of comparing strategies based on impact to industry, this point is important, since a dollar saved on implementing the strategy will be valued very differently (and by different people) to a dollar saved by industry. The RRRA model already handles this by reporting the impacts separately, and allowing the human decision maker to make the trade-off (i.e. to apply the utility functions mentally). What's more, the model assesses the impact on other categories, such as health and the environment, in units of utility. While an analysis in terms of utility for all aspects would be more proper, to keep things simple, the focus of this study will be on impacts that can be easily estimated in dollars, and hence will focus on the model's estimates of the financial impact to industry.

A second issue that complicates the comparison, and the one that concerns us in this study, is the uncertainty over the expected values calculated by the model. The RRRA model is very complex, incorporating 48 sub-models (one for each entry pathway) each having probability and utility parameters numbering into the hundreds and sometimes thousands, and each parameter subject to its own uncertainty. The main model that unifies these sub-models also includes its own parameters (though many fewer). Hence, the final expected values, $EV(S_i)$, that the model calculates will be affected by the uncertainties present across many *thousands* of parameters.

[2]One could also compare the ratio of the expected values:

$$\frac{EV(S_1)}{EV(S_2)} > 1$$

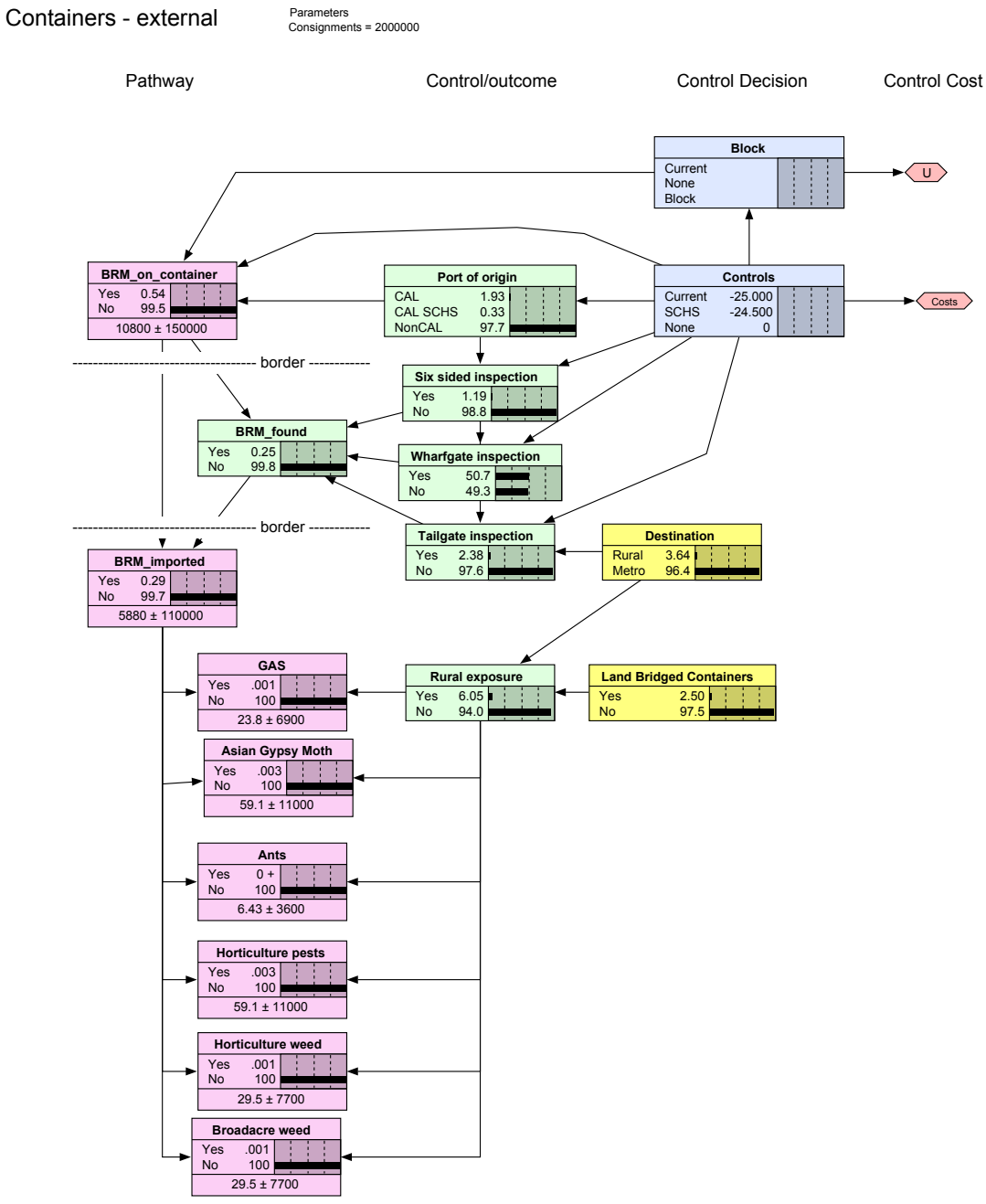


Figure 5: The **Container** entry model

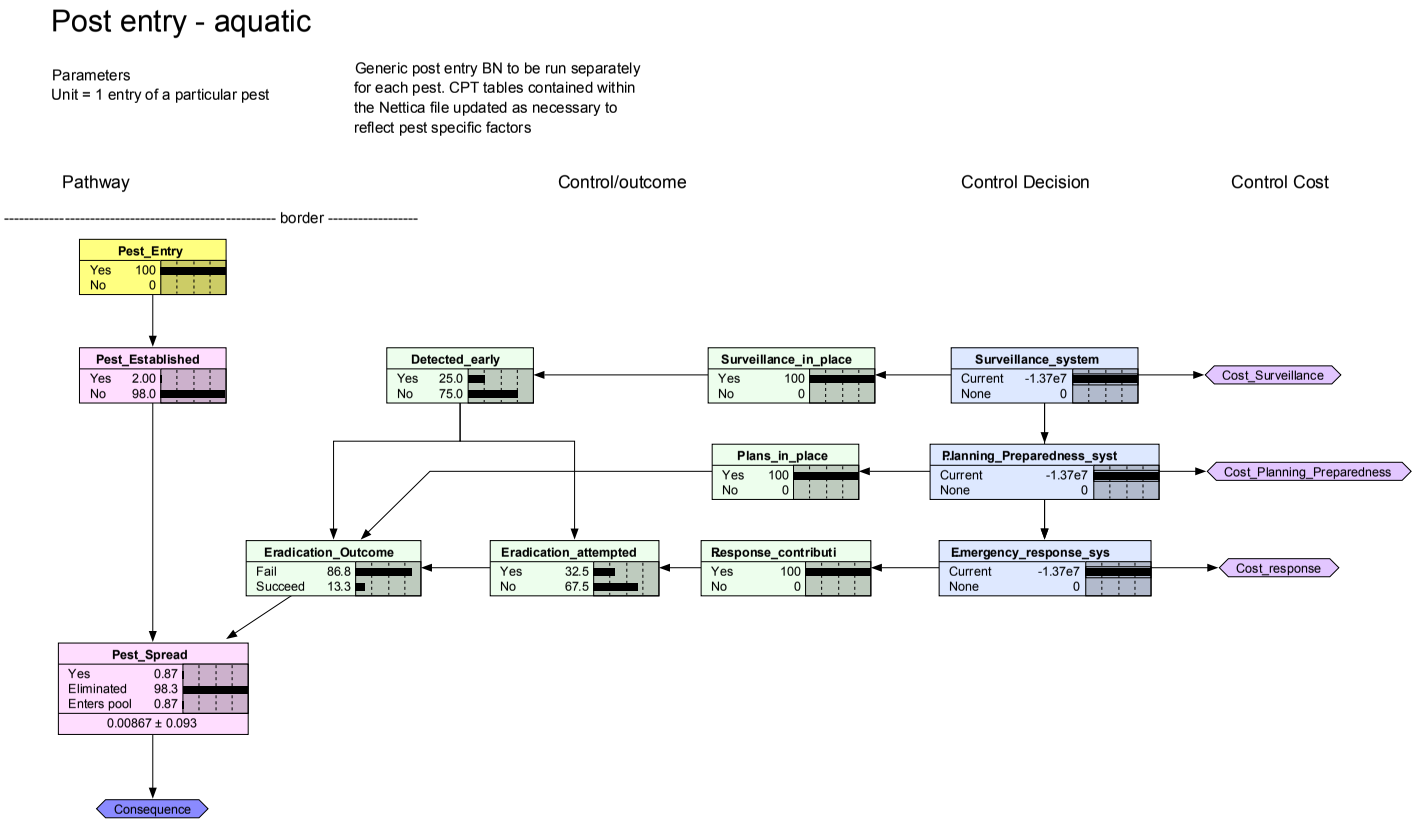


Figure 6: The **Aquatic** post-entry model

Indeed, the issue is more significant than even the above suggests. Models always implicitly reflect a large number of modelling choices and the RRRA model is no different. These choices are implicit parameters, that may be subject to the same sorts of uncertainty as explicit parameters. This may be particularly true in the RRRA model, where the structure of the BN sub-models have not yet had a chance to be widely reviewed and agreed upon. In addition to the probability and utility parameters, the BN sub-models include uncertain choices for the arcs, which variables to include and omit, the variable arities (i.e. number of states) and discretization methods used, the level of abstraction for the model, etc. Of course, the unifying model is also the result of many modelling choices, including the modelling choice that organisms (on and after entry) can be treated separately, and that as the number of spread events increases, the impact diminishes. Fortunately, all of these modelling choices can be treated as just further uncertain parameters during the uncertainty analysis (see, for example, the "Decision Analysis" example from Chapter 2, Saltelli et al., 2004). Unfortunately, however, even on their own, these implicit parameters constitute an intractable number of parameters to analyse!

Finally, to make matters seem especially difficult, if we place no constraints on the parameters during our uncertainty analysis other than those that are logically necessary (by, for example, the rules of probability), the range of uncertainty around $EV(S_i)$ will be so large as to be completely uninformative.

We believe these problems can be managed (if not resolved) with the following techniques:

1. Provide uncertainty distributions (second order priors) for input parameters of the model
2. Implement a model review process to identify parts of the model that can be treated as certain
3. Group parameters and take advantage of local structure wherever possible to reduce the effort needed for confidence assessments and sensitivity analyses
4. Treat the RRRA model as a hierarchy of models, and analyse each part of the hierarchy separately (with a summary of the analysis from lower levels feeding into higher level metamodels)

The first two techniques will allow us to produce an informative assessment of the uncertainty in the model and is ultimately required when doing any uncertainty analysis. The last two techniques will help us manage both the computational demands of the analysis, as well as the manual effort required. Each of the above techniques are described in the following sections.

6.3 Uncertainty distributions for parameters

In order to get a sense of how uncertainties in the inputs of a model affect uncertainties in the outputs, one has to begin by quantifying the uncertainty

on the inputs. Variance-based sensitivity analysis techniques explicitly require distributions to be provided for the input parameters. Indeed, any sensitivity analysis technique explicitly or implicitly requires distributions over the inputs, usually defaulting to the assumption of a uniform distribution over whatever range the parameter happens to span. Using an uninformative uniform distribution may not pose a problem if the number of parameters being investigated is small or their effects on the output are small. In most (interesting) cases, however, this is not true, and it is clearly not true for the RRRA model. Thus, in order to have any hope of providing insight into the uncertainty of the model, specific distributions will need to be provided for the input parameters.

Given the complexity of the RRRA model, providing such input distributions may at first seem to add an unworkable burden. For example, suppose we were to perform an analysis of the Container model in Figure 5 in isolation. It contains 217 probability parameters, or 110 probability parameters if we take into account degrees of freedom. Let's call the parameters of these input distributions *meta-parameters*. Specifying a distribution for each parameter would require at least 3 meta-parameters per parameter (i.e., which distribution to use, such as normal, beta, etc., along with the parameters needed by the distribution, such as means and standard deviations or alphas and betas), giving a total of at least 330 meta-parameters. Specifying all these meta-parameters would indeed be unworkable.

However, we can reduce the effort required significantly. To begin with, the models come with values for the parameters already supplied. Even if we know nothing else about the models, we can make some assumptions about how these parameter values were chosen. In particular, it's reasonable to hope the modeller chose what they considered the most likely value of a parameter, and we can also assume that their uncertainty will be some spread around this value. Hence, it is not unreasonable to use this as an estimate of the input distribution, taking it to be a normal distribution (appropriately constrained; or perhaps a Beta distribution), with the parameter value as mean, leaving only a standard deviation unspecified. This reduces the number of meta-parameters to 110. Depending on how easy it is to specify the spread, this might already be a workable number of meta-parameters. However, it's possible to make further assumptions that reduce the number of required meta-parameters still further, as we will see in Section 6.5.

It's important to note that specifying input distributions makes an important contribution to the documentation of the models. To the extent that input distributions apply to modelling and parameter assumptions (rather than natural variation) they document the modeller's uncertainty across different parts of the model. The modeller's uncertainty is not information that can be derived from any (current) automated analysis.[3]

[3]Though developing such an automated analysis is no doubt possible, and would constitute its own interesting project.

6.4 Structural uncertainty

The above discussion is centered on numerical parameters such as probabilities, utilities and distribution parameters. As noted, however, the models themselves are the result of many modelling choices (such as the choice to include or exclude arcs and nodes, choice of discretizations, etc.), and each such choice can be treated as its own parameter. If alternative choices require additional modelling work, then it is likely to be impractical to include it in the analysis; however, inclusion can be practical if the modelling work can be automated in a computationally efficient way (such as when working with models created via machine learning, or when applying model simplifications).

Some elements of the structure of a BN can feasibly be investigated in an uncertainty analysis without requiring extra modelling work. For instance, the modeller can assign a confidence (or probability) to the existence of each of the arcs in a network. The existence of each arc (or perhaps just low-confidence arcs) can then be treated as a parameter, using a binomial for the input distribution, with p equal to the assigned confidence. Testing the model without the arc marginalises out the influence. This has the side-effect that the meta-parameters associated with any removed conditional probabilities will also disappear. A similar approach can be taken to uncertainty about the existence of nodes. Again, removing the node will marginalise it out, which can both destroy and *create* meta-parameters (for instance, removing X in a $X \rightarrow Y \rightarrow Z$ relationship will create derived parameters between X and Z that need their own meta-parameters). However, meta-parameters created this way can be inferred rather than specified manually. Alternative discretisations can also be tested, assuming the modeller is able to specify a function relating the child to its parents.[4]

The most general approach to dealing with structural uncertainties is to perform model reviews. Model reviews can work well because we usually attach a much higher degree of certainty to structure than we do to parameters — for example, we have much more knowledge about the existence of influences between variables than we have about the degrees of those influences. Thus, with sufficient reviews, we may be able to achieve a wide consensus on the structure[5], and can thereby treat the structure as certain when doing our uncertainty analysis. In the (hopefully limited) cases where this is not true, we can apply meta-parameters to the uncertain structural elements.

6.5 Grouping parameters and local structure

So far, the number of meta-parameters that we need appears almost manageable — potentially, one meta-parameter per free probability parameter, plus perhaps some meta-parameters for structural uncertainty. However, we would still like to drive down the number of meta-parameters as low as possible. One way to do this is to group parameters where the nature of the uncertainty is likely to be similar. An example of where this approach can clearly be applied is in the

[4] BN software such as AgenaRisk allow re-discretisations on the fly if a function is provided.

[5] Something that is often rarely true for numerical parameters!

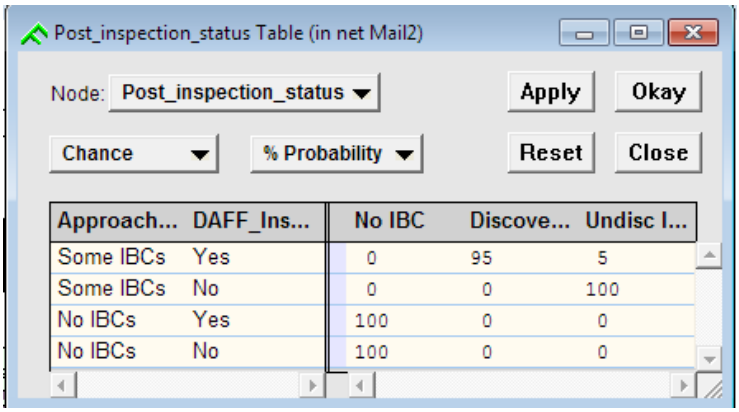


Figure 7: The Post_inspection_status CPT from the Mail network

distributions of the CPT. Figure 7 shows the CPT for the Post_inspection_status node in the Mail entry model. Without grouping, we would need to specify 8 meta-parameters (since each of the 4 distributions has 2 degrees of freedom). However, for a given distribution, it seems reasonable to suppose that the uncertainty that affects the probability of None is the same as the uncertainty that affects the probability of Detected (or Undetected). The uncertainty *might* differ, but in many cases (perhaps the vast majority of cases) they are likely to be the same. Thus, we only really need to specify one measure of spread per row of the CPT, allowing us to halve the number of meta-parameters in this case.

We can scale this kind of grouping up the model. Perhaps our uncertainty for the whole Post_inspection_status node is roughly similar, in which case we can have just one meta-parameter for the whole node. We might go further, and decide we only need one meta-parameter for the network as a whole, or even the entire RRRA model. A practical approach here would be to specify the meta-parameters from the top-down, starting at the entire RRRA model, and then altering the meta-parameter where we believe our confidence differs from the value given at a higher level. (We will see an example of this later.) We need not adhere to this hierarchical approach to grouping — in some cases, we may also want to group parameters in other ways. For example, the cargo models all have a Release_status node, and we might have a similar level of uncertainty for this node across all the cargo entry models (which would override meta-parameter values from higher levels).

This clearly makes meta-parameter specification very manageable. Unfortunately, reducing the number of meta-parameters won't help at all with making the sensitivity analysis computationally tractable. For that, we need to reduce the number and range of input parameters.

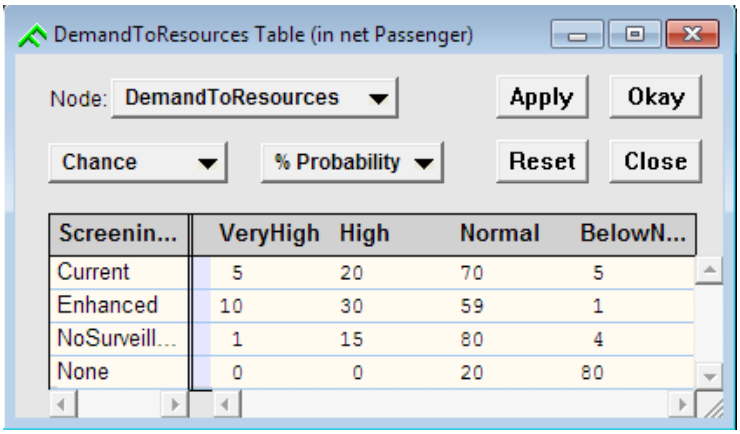


Figure 8: The Demand_to_resources node from the Passenger network

6.5.1 Local structure

One critical technique we can use to achieve a reduction in the number of input parameters is to identify an underlying structure over subsets of parameters. This is particularly relevant when using BNs with discrete CPTs, as the tables are often the discretised product of a more general relation. An example can be seen in Figure 8, which shows the Demand_to_resources node from the Passenger network.[6] Note that each distribution looks to have an underlying model with a central tendency. In particular, given the node states sit on a scale, we might be able to fit a beta distribution that ranges over this scale to each row. We would then have two input parameters per row (i.e. mean and dispersion) rather than three, reducing the total number of input parameters from 12 to 8.

More importantly in this case, sampling the beta distribution's parameters will produce variations in our CPT parameters that are more realistic than if we were to directly sample the three parameters themselves. For instance, we would be surprised to see a row in this table with parameters 0.4,0.1,0.1,0.4 — in fact such a distribution is very likely to be impossible — so we would like to avoid wasting our time testing sensitivity to parameter values like these (and giving a wider spread to the output uncertainty than is warranted).

To provide a quick example of how this would work in practice, we will fit the first row with a beta distribution. Using a simple stochastic search over the beta parameters yields a best-fitting beta with mean of 0.56 and standard deviation of 0.14. Rediscretising this beta distribution in the same way over the 4 states gives the multinomial distribution 0.0135, 0.3223, 0.5747, 0.0895 — by no means an exact fit. In this case, no exact fit is possible and this is likely a common occurrence. Nonetheless, we might prefer the new distribution if we believe it better captures the modeller's intention. Alternatively, we can discretise the distribution in such a way as to achieve a better fit (ideally, taking into account

[6] This CPT has been modified to better illustrate the technique being described.

the fit for all the rows in the CPT at the same time). Once satisfied, we would then have a beta distribution description of each row, and we could then estimate our uncertainty over the beta parameters (the mean and dispersion, or alpha and beta if we prefer).

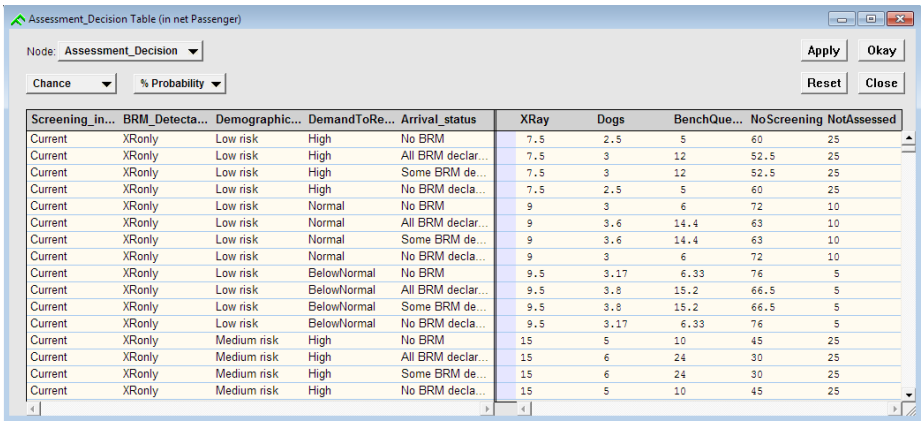
Typically, the model that underlies one row also applies to other rows in the CPT in a related fashion. We can see this in Figure 8; in comparison to the first row, the mean for the second row has shifted to the left, and significantly to the right for the fourth. In the third row, the mean has stayed relatively still, but the distribution has narrowed. Thus, we could specify one beta distribution (for the first row), and provide further meta-parameters indicating how either the mean or the dispersion (or both) change in relation to the first row. Doing so in this case allows us to reduce 12 free parameters to 5. More importantly, our input parameters now capture our intentions better, and hence describe our confidence distributions better. Thus, a dramatic change to the mean of the first row entails similar dramatic changes to the other rows, and specifying local structure allows us to make such related changes automatically, making them amenable to automated analysis.

In many cases, the relations between rows can be summarised by a formula that combines the parents in some way. In Netica, this is often represented by using an equation for the node. In such cases, it is the parameters of the equation that should be treated as input parameters, rather than the CPT probabilities generated from the equation. In other cases (and as is often the case in the RRRA system) the CPTs may be generated by external programs, and again the parameters of those external programs should be the subjects of the analysis. Ultimately, the analysis should be performed on the input parameters to whatever model (be it CPT, equation, program or whatever else) captures the true uncertainties of the modeller.

This also extends to any implied connections between the input parameters. For example, if the modeller knows that the product of a set of input parameters must come to a certain value, these constraints should be a part of the analysis — possibly by including the known constraints (or correlations) in the existing model. In the present example, this might mean using the final product value as the main input parameter rather than treating the individual factors as separate parameters.[7]

The RRRA model currently contains only a few cases where the above approach to exploiting local structure is of value. However, it can often be crucial in such cases. Figure 9 shows an excerpt of the Assessment_decision table from the Passenger entry model. The full table contains 720 rows, and 2880 free parameters, making a VBSA-style analysis completely intractable. However, the parameters are clearly not all uniquely specified — for example, as Demand_to_resources moves from High to BelowNormal, the probability of NotAssessed falls and as Demographic_Profile moves from LowRisk to HighRisk, the probability of NoScreening drops, both independently conditional on Screen-

[7] Alternatively, correlations between inputs can be handled with some VBSA techniques, but these can be complicated to work with [Saltelli et al., 2004].



Screening_in...	BRM_Detecta...	Demographic...	DemandToRe...	Arrival_status	XRay	Dogs	BenchQue...	NoScreening	NotAssessed
Current	XRonly	Low risk	High	No BRM	7.5	2.5	5	60	25
Current	XRonly	Low risk	High	All BRM declar...	7.5	3	12	52.5	25
Current	XRonly	Low risk	High	Some BRM de...	7.5	3	12	52.5	25
Current	XRonly	Low risk	High	No BRM decla...	7.5	2.5	5	60	25
Current	XRonly	Low risk	Normal	No BRM	9	3	6	72	10
Current	XRonly	Low risk	Normal	All BRM declar...	9	3.6	14.4	63	10
Current	XRonly	Low risk	Normal	Some BRM de...	9	3.6	14.4	63	10
Current	XRonly	Low risk	Normal	No BRM decla...	9	3	6	72	10
Current	XRonly	Low risk	BelowNormal	No BRM	9.5	3.17	6.33	76	5
Current	XRonly	Low risk	BelowNormal	All BRM declar...	9.5	3.8	15.2	66.5	5
Current	XRonly	Low risk	BelowNormal	Some BRM de...	9.5	3.8	15.2	66.5	5
Current	XRonly	Low risk	BelowNormal	No BRM decla...	9.5	3.17	6.33	76	5
Current	XRonly	Medium risk	High	No BRM	15	5	10	45	25
Current	XRonly	Medium risk	High	All BRM declar...	15	6	24	30	25
Current	XRonly	Medium risk	High	Some BRM de...	15	6	24	30	25
Current	XRonly	Medium risk	High	No BRM decla...	15	5	10	45	25

Figure 9: The Assessment_decision node from the Passenger network

ing_inspection_system and nothing else. Ultimately, there may be just a handful of rules and corresponding parameters that were used to generate this table, reducing 2880 free parameters to perhaps a few dozen — a much more manageable prospect.[8]

There is one further approach that is especially worth highlighting, as it works particularly well for the RRRA model — namely, identifying parameters that can be treated as fixed. Returning to the Post_inspection_status CPT in Figure 7, we can see that the second through fourth rows are deterministic, suggesting very high confidence (in this case, they suggest logical certainties given the structure of the problem). Similarly, the first row shows that No_IBC is impossible; again, this suggests there is no reason to assume uncertainty over the No_IBC probability in this row, reducing the number of free parameters from two to one. In other cases, we may have domain-specific reasons for having a high enough confidence in a parameter (or group of parameters) that we need not investigate it.

The Mail entry model provides a good example of how this technique can reduce the number of parameters. This model contains 427 conditional probabilities, 249 of which are free parameters. By eliminating deterministic rows from our analysis, we can reduce the number of input parameters to 134. If we also eliminate conditional probabilities of 0 or 1, wherever they might appear (i.e. not just in rows that are entirely deterministic), we can reduce this to 105 parameters.

[8] It's also possible in this case that the node could be further broken up using BN modelling techniques, such as the technique known as 'divorcing parents'; see p.319, [Korb and Nicholson, 2010].

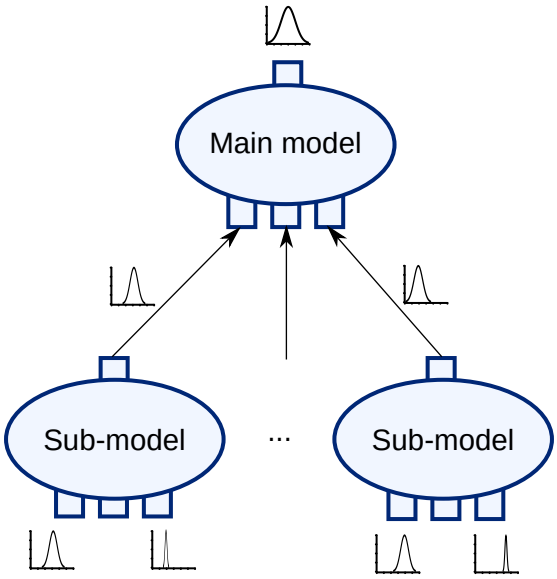


Figure 10: Using sub-models to estimate the variance of inputs in a higher level model

6.6 Hierarchical sensitivity analysis of the RRRA model

While the above techniques are useful, they will not allow us to reduce the number of parameters in the highly complex RRRA model far enough to allow for computational efficiency. One way to resolve this problem is to break down the model in our analysis. The structure of the RRRA model is naturally hierarchical. As noted, the model has already been broken down into sub-models, with 48 entry models (covering the 51 organisms) and 3 post-entry models, as well as the main model that glues them together as described in Algorithm 1. All of the sub-models can be analysed separately, as these do not interact.[9] Furthermore, the main model can also be analysed separately. Such an analysis can be informed by the analyses of the sub-models — in particular, we can characterise the uncertainties in the outputs of the sub-models using a sensitivity analysis, and use those to specify prior distributions over the inputs to the main model. Figure 10 gives a schematic of how this can work. Breaking up the full model in this way can be viewed as a form of metamodelling. This is especially true when we analyse the main model, since it is merely a reduced version of our full model.

In some cases, it may be possible to extend this hierarchical approach further, by breaking down the sub-models themselves. Given that the sub-models are presently all represented as BNs, one possibility would be to re-work the models

[9] Of course, that sub-models don't interact is a modelling choice that we are very confident about, and thus assume does not impact the uncertainty of our output.

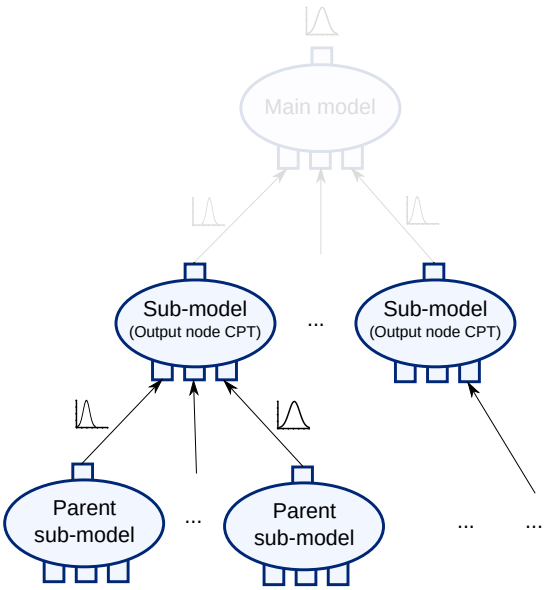


Figure 11: Breaking up the BNs into sub-models by analysing the output node separately

as object-oriented BNs (OOBNs). However, we need not go to that effort in order to break up the analysis, as we can take advantage of the structure of a BN. While any node in a BN can be treated as an 'output' node, in the RRRA sub-models, output nodes always sit at the bottom — i.e. as leaf nodes, with parents but no children. One way to split up the analysis of the BN, therefore, is to analyse the CPT of the output node, separately from the sub-parts of the network capable of influencing the parent nodes. (See Figure 11.) This will work well when the parent nodes are independent. When the parents are not independent (as is often the case in the RRRA models), one can use node absorption (see Section 7.2.3, [Kjærulff and Madsen, 2008]) to retain any correlations between the two parents that are present in the underlying joint distribution.

7 Example analysis

We will now look at examples of how some of the above techniques can be employed. Not all the techniques are covered, but the examples provided should provide enough grounding in the basic techniques to make extensions straightforward.

It should be noted that the following analysis is based on an in-development version of the RRRA model, filled with many placeholder parameters. Therefore, the following analysis is solely to illustrate the techniques, and cannot be

Organism	Entry Frequency	Spread Probability	Impact per Spread Event ($m)
AGM	67.85636691	0.00614	220
animal_other_bacteria	11.31414285	0.014354998	100
animal_other_microorg_other	2.484612038	0.014354998	100
Animal_other_virus	2.491915444	0.014354998	100
Ant	6.141447392	0.00614	0
...			

Table 2: Extract of resultant input values for a full RRRA model run, with all controls set to their default

used to draw any conclusions about the final version of the RRRA model.

7.1 The main RRRA model

We will begin with a metamodel analysis of the main (highest level) model, with sub-models excluded from the analysis. Our metamodel will look much like the algorithm in Algorithm 1, however lines 3-22 will be removed, along with all consequence categories aside from 'financial impact to industry'. To make up for this removal, for each organism, m, the number of organism entries via any pathway, N_m, and the spread probability, S_m, will be taken as inputs to the metamodel. This gives us the simplified model in Algorithm 2.

Algorithm 2 RRRA metamodel

1: **for** each organism **do**
2: ▷ *Expected number of spread events for this organism*
3: $\mathrm{ES}_m \leftarrow \mathrm{N}_m \times \mathrm{S}_m$
4:
5: ▷ *Calculate expected financial impact to industry. d is a diminishing function*
6: $\mathrm{EC}_m \leftarrow$ Impact per spread event $\times\ d(\mathrm{ES}_m)$
7: **end for**
8:
9: ▷ *Calculate total expected impact for category i over all organisms*
10: $\mathrm{EC} \leftarrow \sum_m \mathrm{EC}_m$

In order to anchor our uncertainty over the inputs, we will run the full RRRA model to obtain point estimates of the relevant parameters. We will leave the management actions for this run at their default setting ('Current' for all actions except Vessel_surveillance_system, which is 'None'). Table 2 shows an extract of the results of such a run for the input parameters to our metamodel.

For this analysis, we will assume all these input parameters to be fixed, aside from the entry frequencies, giving us 51 input parameters (one for each

organism). Also note that the final expected impact to industry for this model run is about $30 billion.

We now need to choose a type of distribution to characterise the uncertainty over these inputs. Given that, for each parameter, we have a point estimate for the mean of the parameter (derived from the sub-models), and that we expect such estimates to vary around a centre point, a normal distribution seems appropriate. Of course, none of the frequencies can be *below* zero, so we would need to use a truncated normal distribution. We have other choices, of course, such as beta or truncated log-normal distributions, if we believe those better characterise our uncertainty. However, for simplicity, here we will stay with the normal.

At this point, we have several further choices for how we specify the meta-parameters of the normal distribution for each parameter. One direct approach would be to take the values from the model run as means, and specify a standard deviation for each. A slightly more intuitive approach would be to specify an error interval around the point estimate (e.g. $\pm 10\%$) coupled with an assumption on how much of the distribution's mass this interval captures (e.g. 99.7%). From this, we can estimate a standard deviation (e.g. $99.7\% = 3$ standard deviations). We could also allow the user to specify their own interval boundaries (using the point estimate as information only), and estimate a mean and standard deviation from those boundaries. One good approach if a group of experts is available is to use the 4 step Delphi protocol [Burgman et al., 2011], in which the group is asked for their highest, lowest and best estimates, along with a rating of their confidence, which can easily be translated into truncated normal or (better still) beta distributions.

Here, we will use an error interval of $\pm 10\%$ for every point estimate, along with the assumption that this captures almost all (99.7%) of the input uncertainty. Thus, for example, our interval for AGM entry frequency is $[61.1, 74.6]$, which translates to a S.D. of 2.26. This choice of error interval is arbitrary, however it will serve to illustrate the method. In practice, these particular intervals would be estimated by analysis of the sub-model outputs, as described in Section 7.3. In the absence of sub-models, estimation from data or elicitation approaches like the 4 step Delphi protocol would be recommended. In the absence of any solid knowledge, a variety of intervals can be investigated to at least produce an understanding of how the model is affected by different kinds of uncertainty.

With our input distributions specified, we can now perform our variance-based sensitivity analysis. For this analysis, we have use the R package called 'sensitivity' that implements a range of efficient algorithms for calculating sensitivity indices. The overall analysis process is managed from within a Python script (calling into R as needed), which also encodes the metamodel. There are 3 major phases in the sensitivity analysis:

1. Setup the experimental design (i.e. generate a sample over the input space)

2. Run the model on each point in the experimental design

	A	B	C	D	E	F
1	AGM	animal_ot	animal_ot	Animal_ot	Ant	Aquatic_b
2	67.85637	11.31414	2.484612	2.491915	6.141447	5.210479
3	70.60907	11.36821	2.496485	2.503823	6.170795	5.235377
4	75.63023	11.42255	2.508419	2.515793	6.200294	5.260405
5	71.8997	11.47747	2.520479	2.527888	6.230103	5.285695
6	68.83591	11.53327	2.532732	2.540177	6.26039	5.31139
7	66.20019	11.59029	2.545254	2.552736	6.291342	5.337651
8	62.73018	11.64894	2.558134	2.565653	6.323177	5.364661
9	62.05894	11.70969	2.571474	2.579033	6.356153	5.392637
10	65.84845	11.77312	2.585404	2.593004	6.390585	5.42185
11	68.50656	11.83998	2.600087	2.60773	6.426878	5.452641
12	71.43755	11.91126	2.61574	2.623429	6.465568	5.485466

Figure 12: Extract of the metamodel experimental design

3. Calculate the variances and sensitivity indices

An extract of the experimental design generated using the extended-FAST method (called 'fast99' in R) is shown in Figure 12. Each row constitutes a set of input parameters for a single run and each cell corresponds to one of the entry frequency parameters (named in the column header). This design is simply a sampling of the input space that allows variances and sensitivity indices to be calculated with as few model runs as possible. We could also have used a Latin hybercube or simple Monte Carlo design, but at a greater cost to efficiency.

After creating the experimental design, we run our model on each point (i.e. each row of Figure 12). This allows us to produce a distribution over the outputs, giving Figure 13. This gives us our first insight into the output variance given the uncertainties we specified in the inputs. Note that the variance is roughly centred around $30 billion (our expected impact from the model run) as we would hope (though not always expect, if there were non-linear responses from significant parameters).

We can also now identify which parameters the model is most sensitive to. Table 3 summarises the sensitivity indices computed for each of the parameters, as well as their impact on the output variance. Only the ten most sensitive parameters are shown. The *Main Effect* column refers to the first order sensitivity index for a parameter — that is, what proportion of the total output variance does the parameter *by itself* account for. The *Total Effect* column refers to both the main effect plus all of the interaction effects it has with other parameters. If there are no interactions between parameters, then the total effect will be equal to the main effect, and the sum of the main effects will be equal to 1. If there are indeed interaction effects, the sum of the main effects will be less than 1 (since 1 - the sum of main effects is equal to the effect of the interactions) and the sum of the total effects will be greater than 1 (due to double counting of interactions across parameters). The final two columns show the proportion of the output variance (represented by the standard deviation here, since it is

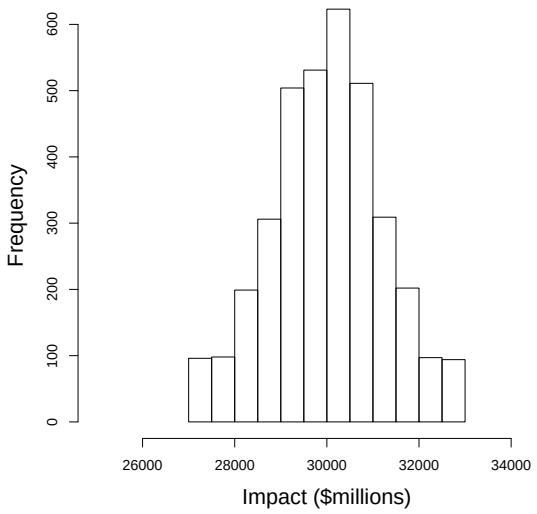


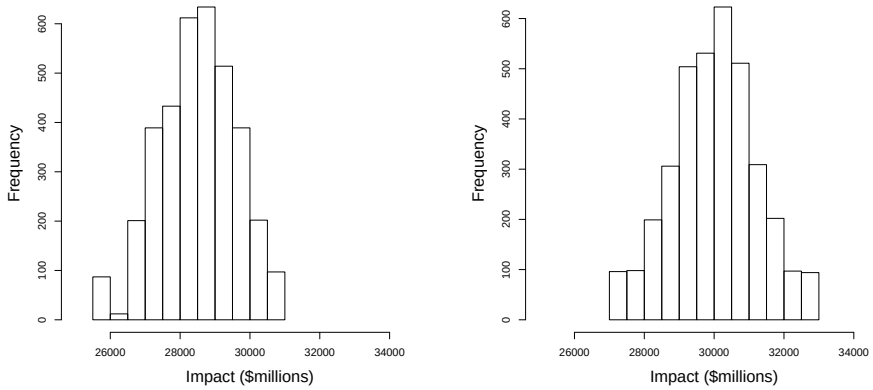
Figure 13: Output distribution of the metamodel sensitivity analysis

easier to interpret) caused by the parameter either by itself (Output SD), or in interactions with other variables (Output Total SD).

The sensitivity analysis indicates the metamodel is highly sensitive to the FMD, Kaphra and fruitfly entry frequency parameters. This is as we would expect: inspection shows the impact to industry per unit for these organisms is much higher than for any other organism.

Entry Frequency Parameter	**Main Effect**	**Total Effect**	**Output SD**	**Output Total SD**
FMD	0.3084676	0.337481661	472.9537387	494.6966559
Khapra_beetle	0.083661018	0.113564265	273.9412843	319.1661036
fruitfly	0.031406199	0.061502309	180.2453882	252.2331587
Zoonotic_bacteria	0.002508103	0.032688937	56.47561472	203.886618
Horticulture_bacteria	0.000536184	0.030728995	26.78916765	202.8039701
Horticulture_virus	0.000529311	0.030722185	26.62083472	202.8112635
Livestock_microorg_oth	0.000315339	0.030433094	20.65194211	202.882784
Broadacre_fungus	0.000202883	0.030397607	16.62685886	203.5200334
horticulture_fungus	0.000160603	0.030354954	14.81717124	203.7058103

Table 3: Sensitivity indices for the 10 most sensitive parameters for the metamodel run



(a) (b)

Figure 14: (a) Output distribution of the metamodel Scenario B sensitivity analysis vs (b) the original Scenario A output distribution

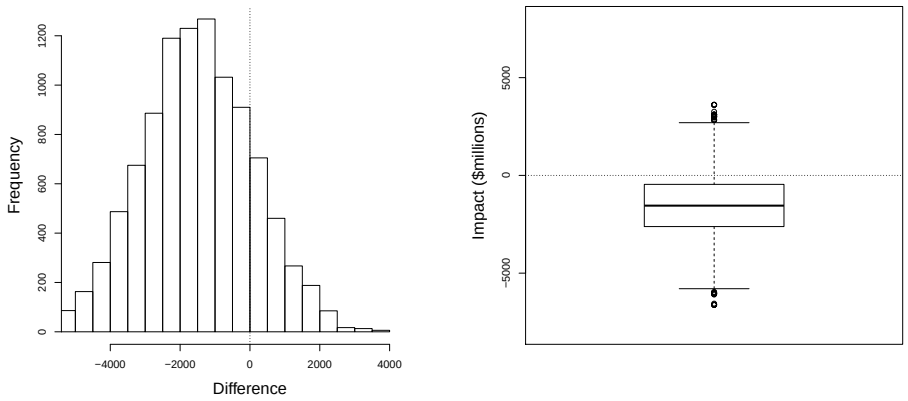
7.2 Comparing management strategies with the RRRA model under uncertainty

As described in Section 6.2, the RRRA model is capable of comparing the consequences of two management strategies based on expected value. However, uncertainty affects this comparison. Ideally, we would like to know how probable it is that the management strategy with the higher expected value will in fact be better. This will depend on the uncertainty in our model, and the example above shows how we can begin to understand that uncertainty. Here, we will show how the decision maker can make use of that understanding to compare scenarios.

We will run the RRRA model again, this time setting the management controls to 'Modified' where possible, or otherwise 'Current'.[10] Let's call the original run Scenario A and the run in which controls are set to 'Modified' as Scenario B. When we do so, the model calculates an expected impact to industry of $28.5 billion. The implementation costs are roughly equal ($329 million to $332 million), thus, there is an expected $1.5 billion saving to industry in Scenario B. Let's setup the input parameters to our metamodel as before, by taking the point estimates from the RRRA model under Scenario B as the parameter value, ±10%. Figure 14 shows the distribution over the outputs.

If we do a statistical test, the distributions are highly significantly different, but of course that's of little interest (we knew they were different to begin

[10] 'Modified' is a placeholder option that, in practice, would be replaced by specific management interventions. However, we use it here to provide a simple illustration of the technique, by creating a contrast between the 'current' management strategy and some 'modified' management strategy.



(a) (b)

Figure 15: (a) Distribution over differences between Scenario A and Scenario B (i.e. the Scenario B effect size). (b) Box plot of the same data.

with). More interestingly, we can derive a distribution over the possible differences between outcomes for Scenario B and Scenario A (Figure 15a). This distribution was produced by sampling the Scenario A and Scenario B outputs in pairs, 10,000 times, and calculating the difference for each pair. This allows to calculate the chance that Scenario B will produce a lower impact; in this case, it is 82%. Thus, we can now say to our decision maker that, assuming our model is correct and given ±10% uncertainty in each of our input parameters, Scenario B has an 82% chance of producing a better outcome than Scenario A, and our best estimate of that improvement is $1.5 billion. If the decision maker is open to having more information, we can provide the full distribution over the expected effect sizes. Alternatively, we can provide a box plot with the quantiles which may prove easier to understand (Figure 15b) or the equivalent in words (e.g. 25% chance of at least $2.6 billion saving, 50% chance of at least $1.5 billion saving and 75% chance of at least $400 million saving).

It is important to note that this is not a final or absolute statement about the value of Scenario A vs Scenario B. In particular, we have assumed certainty for our model, as well as for other input parameters (such as impacts and postentry spread probabilities). If the decision maker is happy with such assumptions, then she can decide if an 82% chance of producing an improved outcome is worth the cost. Otherwise, we can revisit the model, and reduce our uncertainty, by focusing in on the variables that the sensitivity analysis indicates are most in need of attention. For example, by reducing the uncertainty for FMD, Kaphra_beetle and fruitfly to ±2%, we get the distributions shown in Figure 16, giving a 99.5% chance that Scenario B produces an outcome better than Scenario A (given we consider our model and other parameters to be

certain).

It should also be noted that the metamodel that we created here is not the only one that could be created at this level of abstraction. We have focused on a metamodel that looks at sensitivities to organism frequencies because it provides a clear demonstration of how the metamodel analysis can be separated from the analysis of the sub-models, while still being informed by that analysis. However, other elements of the model can also be incorporated into the analysis. For example, spread probabilities, consequence estimates and diminishing impact functions (none of which appear in the sub-models) can all be treated as input parameters, depending on the question one wants to answer. It's also possible that one is only interested in a restricted set of input parameters to the metamodel, perhaps because proposed management strategies affect only one or two pathways. In such a case, the metamodel could treat all but those pathways of interest as having fixed parameters, so as to focus on the uncertainty surrounding the changes in those specific pathways. There should be little risk in doing so, given the independence of the sub-models. The approach proposed here allows for all of these alternatives,[11] requiring only slight changes to the metamodel used.

7.3 Analysing the BN sub-models

Now that we've seen the framework for analysing the metamodel, we can step down the hierarchy to the sub-models. In some sense, if we had enough confidence in our uncertainty estimates over the input parameters of the metamodel, we might not need to take this step. In particular, if our sensitivity analysis gave us enough confidence to pick one scenario over another at this stage of the analysis, we would not need to move further down the hierarchy. However, assuming we needed further surety, we can analyse the sub-models to improve our understanding of the uncertainty surrounding their outputs, and we can feed such improvements directly into the uncertainty distributions for the metamodel input parameters.

As noted in Section 6.5, we can assign confidence meta-parameters to groups of parameters within our BN, such as to CPT rows, CPT columns, whole CPTs, or parameters across an entire network. In this example, we will apply that approach to the analysis of the Container entry model (see Figure 5). As noted earlier, the Container model has 110 free probability parameters. This allows for a relatively tractable sensitivity analysis.[12]

As we did for the metamodel, we have created a Python script to manage the overall analysis process, which in this case also makes use of the Netica API to manipulate the BNs. We also take advantage of the Netica API to embed the meta-parameters into the Netica BN files. Figure 17 shows how we have chosen to embed the meta-parameters that specify the spread (as a standard deviation) in this example. By default, the parameters in the network are assumed to have

[11] As do the scripts developed for this report.

[12] The experimental design takes just 3 minutes to produce on a single CPU on a 2011 model laptop, and the model evaluations take about a minute overall.

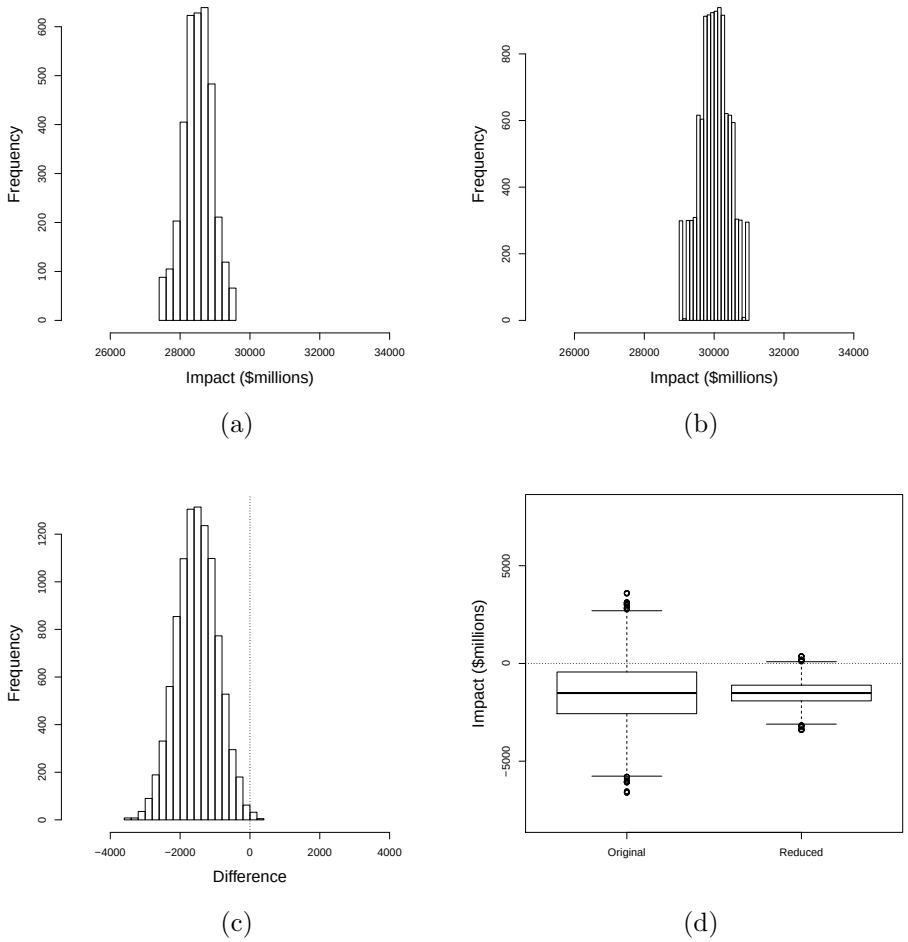


Figure 16: The effect of reducing uncertainty in FMD, Kaphra_beetle and fruit-fly. (a) Output distribution of the metamodel Scenario B sensitivity analysis vs (b) the Scenario A output distribution. (c) The new effect size distribution between Scenario A and Scenario B. (d) Box plots of the original output distribution (Original) and output distribution with reduced input uncertainty (Reduced).

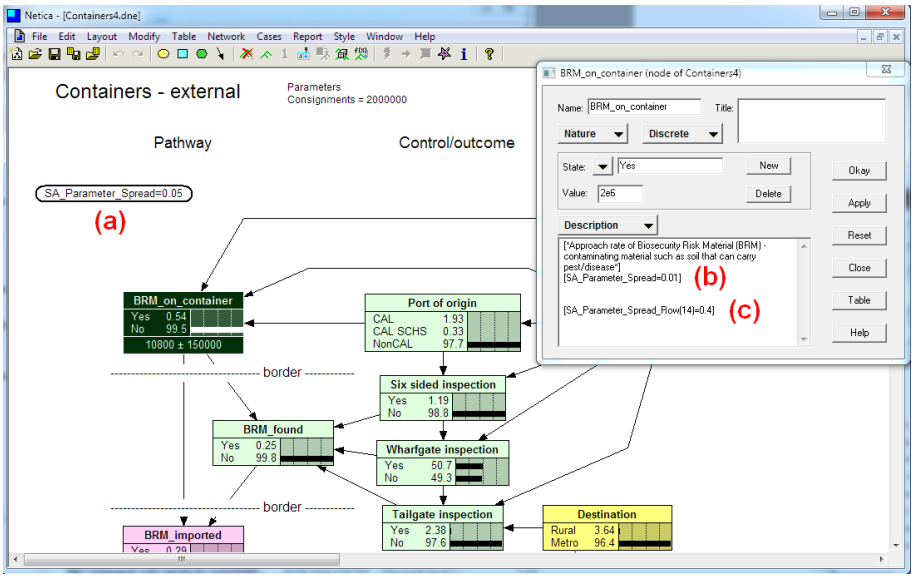
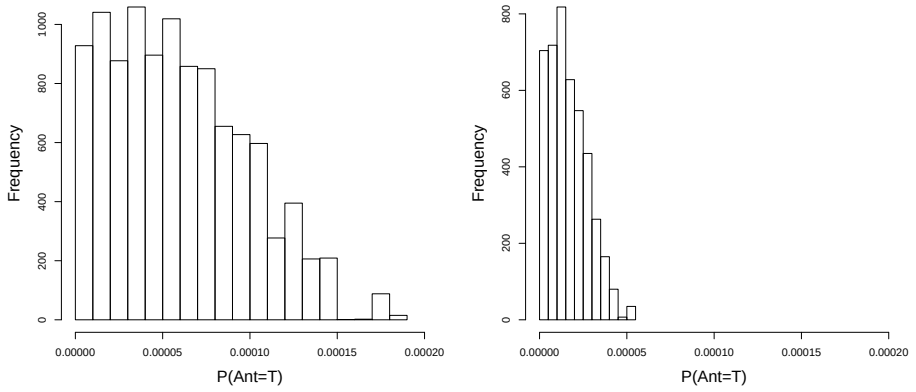


Figure 17: Embedded meta-parameters for the network. (a) The default standard deviation meta-parameter for the entire network, (b) the default for the BRM_on_container node and (c) a custom meta-parameter for row 14 of the BRM_on_container node.

a spread defined by a normal distribution, with the mean being the existing value of the parameter, and the standard deviation given by the constant node (a). We specify a custom spread meta-parameter for the BRM_on_container nodes in (b) and another for one of the CPT rows (row 14) in the BRM_on_container node in (c). The Python script reads the network, extracts the meta-parameters, and then runs through the 3 phases of the sensitivity analysis (setup experimental design, run models and calculate variance and sensitivity indices) as before, but taking into account the extracted meta-parameters when setting up the experimental design. We focus on the posterior for the Ant output of this model — in particular, $P(\text{Ant} = \text{T})$ — leaving the controls unspecified (i.e. assuming uniform probabilities over the controls).

7.3.1 A full CPT analysis

When we run the analysis with no meta-parameters other than the default spread for the network, we get the output distribution for the Ant posterior shown in Figure 18a. This distribution may be of interest in itself, or it may be used as the input distribution for our metamodel (for the Ant organism entry probability on the Container pathway). As always, the degree to which we are justified in doing this will depend on the degree to which we trust our meta-parameters. In the absence of such trust, we should try a variety of generic meta-parameters to understand how the model responds to different levels of



(a) (b)

Figure 18: Output distribution over $P(\text{Ant} = \text{T})$ with (a) all parameters and (b) probabilistic parameters only

uncertainty.[13]

Of particular interest, we can now identify the parameters that have the greatest impact on our uncertainty over the output (Table 4). Note the parameter names encode the cell of the CPT from which they come — the first number after the node name indicates the row number of the CPT (as viewed in Netica), and the second number indicates the column. These accord with what we expect — in particular, it is no surprise that BRM_Imported has the greatest impact on Ant, being a direct parent. It's also not especially useful information, as the BRM_Imported node is deterministic and encapsulates a logical relationship that couldn't be otherwise. BRM_on_container is also clearly influential, which accords with the impact we observe when changing its value in Netica (but keep in mind that the sensitivity analysis varies over CPT values, not over sets of possible evidence).

7.3.2 Specifying custom uncertainties

Let us narrow our analysis now to the parameters in which we are most interested (and likely to be least confident). We will do so first by omitting deterministic rows and probabilities. Doing this in the Container model gives us 44 free parameters, and generates the output distribution shown in Figure 18b and the sensitivity indices shown in Table 5. Our uncertainty has now clearly been reduced. In addition, BRM_imported disappears from our sensitivity table (as it should), and the order of the remaining parameters changes

[13] We may even adopt an info-gap style approach here to discover how our management strategies compare over different levels of uncertainty, though how to do so systematically is not clear.

Parameter	Main Effect	Total Effect	Output SD	Output Total SD
BRM_imported_3_0	0.776702487	0.801993602	2.65E-05	2.69E-05
BRM_on_container_8_0	0.005833892	0.023369383	2.81E-06	5.63E-06
BRM_on_container_17_0	0.005833892	0.023369383	2.81E-06	5.63E-06
BRM_on_container_26_0	0.005430693	0.02289524	2.72E-06	5.58E-06
Port_of_origin_2_0	0.001574675	0.018378988	1.52E-06	5.19E-06
BRM_on_container_5_0	0.001456881	0.018095519	1.46E-06	5.16E-06
BRM_on_container_14_0	0.001456881	0.018095519	1.46E-06	5.16E-06
BRM_on_container_2_0	0.001456447	0.018094623	1.46E-06	5.16E-06
BRM_on_container_11_0	0.001456447	0.018094623	1.46E-06	5.16E-06
BRM_imported_2_0	0.001430924	0.015608651	1.42E-06	4.70E-06

Table 4: Sensitivity indices for the 10 most sensitive parameters (not including the Ant CPT itself) in the Container network

Parameter	Main Effect	Total Effect	Output SD	Output Total SD
BRM_on_container_8_0	0.135835696	0.152595755	3.13E-06	3.32E-06
BRM_on_container_17_0	0.135835696	0.152595755	3.13E-06	3.32E-06
Port_of_origin_2_0	0.014142188	0.030223666	1.18E-06	1.73E-06
BRM_on_container_2_0	0.008719815	0.025824417	9.52E-07	1.64E-06
BRM_on_container_11_0	0.008719815	0.025824417	9.52E-07	1.64E-06
BRM_on_container_5_0	0.008183333	0.025544052	9.26E-07	1.64E-06
BRM_on_container_14_0	0.008183333	0.025544052	9.26E-07	1.64E-06
Destination_0_0	0.00139647	0.019143149	4.22E-07	1.56E-06
BRM_on_container_6_0	0.001316926	0.016413419	3.90E-07	1.38E-06
BRM_on_container_15_0	0.001316926	0.016413419	3.90E-07	1.38E-06

Table 5: Sensitivity indices for the 10 most sensitive *probabilistic* parameters (not including the Ant CPT itself) in the Container network

slightly, but the story for these other parameters otherwise remains much the same. Now suppose we have some reason to be more confident about almost all of the BRM_on_container node's parameters, but substantially less confident about row 14 of that node's CPT. Rerunning the analysis gives us Table 6. Here we see that the sensitivity to BRM_on_container has reduced somewhat, except for the one free parameter in the uncertain row, which has the highest sensitivity index by quite a way. The result in this case is intuitive, but it's important to note that a higher uncertainty distribution does not always produce higher sensitivities — if we increase the uncertainty on the BRM_Found node to even very high levels (say, with a standard deviation of 0.8), one of its non-deterministic parameters has the second highest sensitivity index, while the remaining 6 parameters all fall outside the top 10. This counter-intuitive result can be double-checked by opening the Container BN and removing its conditional probability table: doing so leaves the marginal probabilities for the Ant node nearly unchanged!

Parameter	Main Effect	Total Effect	Output SD	Output Total SD
BRM_on_container_14_0	0.671063368	0.694115992	4.06E-06	4.13E-06
Port_of_origin_2_0	0.062791163	0.067718781	1.38E-06	1.44E-06
BRM_on_container_8_0	0.016972884	0.022364641	7.82E-07	8.98E-07
BRM_on_container_17_0	0.016972884	0.022364641	7.82E-07	8.98E-07
Destination_0_0	0.001402837	0.008937726	2.62E-07	6.61E-07
BRM_on_container_2_0	0.001288608	0.007363271	2.34E-07	5.59E-07
BRM_on_container_11_0	0.001288608	0.007363271	2.34E-07	5.59E-07
Wharfgate_inspection_4_0	0.001254049	0.008236173	2.46E-07	6.32E-07
BRM_on_container_5_0	0.001217699	0.007359897	2.28E-07	5.60E-07
Port_of_origin_1_0	0.000939848	0.008009676	2.12E-07	6.20E-07

Table 6: Sensitivity indices for the 10 most sensitive probabilistic parameters (not including the Ant CPT itself) in the Container network, with uncertainty on BRM_on_container reduced, and the uncertainty on Row 14 increased

7.3.3 Structural analysis

Finally, it's worth demonstrating that structural elements can also be treated in the analysis. Here, we will focus on the arcs in a model. Again, we look at the Container model, and will treat four arcs as uncertain (chosen to illustrate a range of sensitivities):

1. Controls → Point_of_origin
2. Controls → BRM_on_container
3. BRM_found → BRM_imported
4. BRM_on_container → BRM_imported

Each arc will be assigned a probability (confidence) of 0.5 — thus, the arc will be present in 50% of the model evaluations. Figure 19 shows the output distribution when we take these arcs into account in the analysis. (We use a density plot here to capture more of the detail of in the output distribution, that would otherwise be lost in a histogram.) Table 7 shows the sensitivity indices.

We can immediately note that this distribution fails to have a neat centre, and appears to be multimodal. In fact, if we analyse the experimental design and the resultant outputs, we can note that when the BRM_found → BRM_on_imported arc is missing, the posterior for P(Ant=T) is always 0 (hence the large spike in the distribution at 0). We can also note that when BRM_on_container → BRM_imported arc is missing, P(Ant=T) is two orders of magnitude lower than when it is not missing. This explains the two arcs high sensitivity indices. The other two arcs don't have nearly the same effect; in particular, Controls → Point_of_origin is quite insignificant in both main effect and total effect[14] and might well be a candidate for removal if further model simplicity were sought.

[14]Opening the Container BN and deleting this arc shows that it does not have much impact on the marginal probability of Ant, unlike the other arcs.

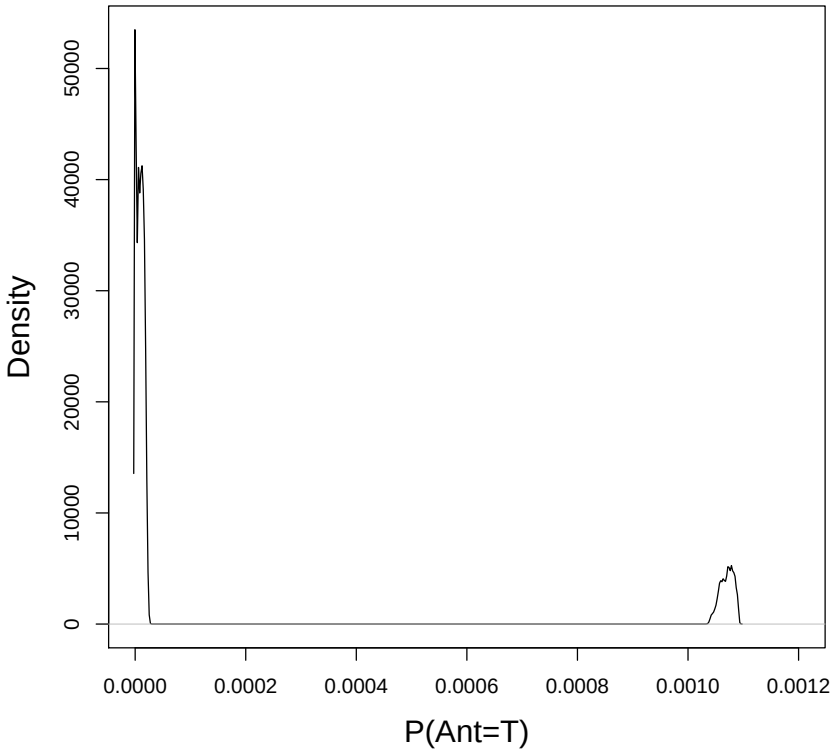


Figure 19: Output distribution over $P(\text{Ant} = \text{T})$ for probabilistic parameters and selected arcs

Parameter	**Main Effect**	**Total Effect**	**Output SD**	**Output Total SD**
BRM_found → BRM_imported	0.316553213	0.682320037	0.000257984	0.000378759
BRM_on_container → BRM_imported	0.296958654	0.666954495	0.000255037	0.000382211
BRM_on_container_14_0	0.134378139	0.327031974	2.27E-06	3.55E-06
Controls → BRM_on_container	0.087042325	0.328248605	9.83E-07	1.91E-06
Port_of_origin_2_0	0.027984185	0.175550477	7.57E-07	1.90E-06
BRM_on_container_8_0	0.014053708	0.156627114	5.14E-07	1.72E-06
BRM_on_container_17_0	0.014053708	0.156627114	5.14E-07	1.72E-06
BRM_on_container_2_0	0.006965147	0.148836792	3.51E-07	1.62E-06
BRM_on_container_11_0	0.006965147	0.148836792	3.51E-07	1.62E-06
BRM_on_container_5_0	0.006948859	0.148841704	3.50E-07	1.62E-06

Table 7: Sensitivity indices for the 10 most sensitive parameters, including arcs

It's also very interesting to see that the total effects of the arcs are substantially higher than their main effects — this is quite intuitive, as their presence enables other parameters to have more of an effect. In the extreme case, BRM_found → BRM_imported disables the effect of other parameters entirely when it is missing.

8 Summary and future work

While there are a variety of approaches for making decisions with models subject to severe uncertainty, many have shortcomings or are still at an early stage of development. Therefore, here we chose to develop on an approach for the RRRA model that is based on the widely used and tested technique known as variance-based sensitivity analysis, while folding in useful ideas from standard Bayesian analysis such as higher level priors (i.e. confidence estimates). The core of this approach was outlined in Section 6.2 and consisted of the following techniques:

1. Provide uncertainty distributions (second order priors) for input parameters of the model
2. Implement a model review process to identify parts of the model that can be treated as certain
3. Group parameters and take advantage of local structure wherever possible to reduce the effort needed for confidence assessments and sensitivity analyses
4. Treat the RRRA model as a hierarchy of models, and analyse each part of the hierarchy separately (with a summary of the analysis from lower levels feeding into higher level metamodels)

We also proposed a method for incorporating the results of such an analysis directly into the decision making process, allowing the decision maker to compare strategies on the basis of both expected value and uncertainty, and in this case, allowing the decision maker to weigh the cost of a management intervention against the changes in risk. This information can be communicated to the decision maker and stakeholders in either straightforward or technical forms (e.g. summary numbers versus effect size distribution graphs), or in forms that aim to balance the level of information against simplicity of communication (such as a box plots or verbal quantile scales).

Importantly, we also ensured that the method we developed was practical and showed how it can be applied to the RRRA model. While the model contains thousands of parameters and modelling choices, each requiring its own confidence assessment, we can reduce both the assessment burden and the computational burden by grouping parameters, taking advantage of local structure, and breaking up the model into hierarchical components (techniques 3-4 above). This allows submodels containing several hundred parameters to be reduced to

several dozen, and allows easily analysed metamodels to be created that are informed by more detailed lower level models only where necessary.

While the proposed approach is ready to be used as is, there are some extensions that may be worth exploring in future. At a technical level, it would be worth extending the investigations into structural uncertainty to better account for modelling choices such as level of discretisation and variable selection. More generally, it would be useful to see if an info-gap inspired approach could be applied, in which a range of confidence assessments are explored in order to identify the strategy that achieves a desired performance criterion most robustly. It would also be of at least curiosity value to see if the method could be extended in such a way as to create a complete decision rule.

References

T. Bedford and R. Cooke. *Probabilistic risk analysis: foundations and methods*. Cambridge University Press, 2001.

Y. Ben-Haim. Information-gap decision theory: decisions under severe uncertainty. 2006.

J. Berger et al. The case for objective bayesian analysis. *Bayesian Analysis*, 1 (3):385–402, 2006.

J. O. Berger. Robust bayesian analysis: sensitivity to the prior. *Journal of Statistical Planning and Inference*, 25(3):303 – 328, 1990. ISSN 0378-3758. doi: http://dx.doi.org/10.1016/0378-3758(90)90079-A. URL `http://www.sciencedirect.com/science/article/pii/037837589090079A`.

M. A. Burgman and H. M. Regan. Information-gap decision theory fills a gap in ecological applications. *Ecological Applications*, 24(1):227–228, 2014.

M. A. Burgman, M. McBride, R. Ashton, A. Speirs-Bridge, L. Flander, B. Wintle, F. Fidler, L. Rumpff, and C. Twardy. Expert status and performance. *PLoS One*, 6(7):e22998, 2011.

S. Conti and A. O'Hagan. Bayesian emulation of complex multi-output and dynamic computer models. *Journal of Statistical Planning and Inference*, 140(3):640 – 651, 2010. ISSN 0378-3758. doi: http://dx.doi.org/10.1016/j.jspi.2009.08.006. URL `http://www.sciencedirect.com/science/article/pii/S0378375809002559`.

L. Davidovitch, R. Stoklosa, J. Majer, A. Nietrzeba, P. Whittle, K. Mengersen, and Y. Ben-Haim. Info-gap theory and robust design of surveillance for invasive species: The case study of barrow island. *Journal of environmental management*, 90(8):2785–2793, 2009.

J. C. Felli and G. B. Hazen. Sensitivity analysis and the expected value of perfect information. *Medical Decision Making*, 18(1):95–109, 1998.

D. R. Fox, Y. Ben-Haim, K. R. Hayes, M. A. McCarthy, B. Wintle, and P. Dunstan. An info-gap approach to power and sample size calculations. *Environmetrics*, 18(2):189–203, 2007.

S. French. Modelling, making inferences and making decisions: the roles of sensitivity analysis. *Top*, 11(2):229–251, 2003.

J. C. Helton, J. D. Johnson, C. J. Sallaberry, and C. B. Storlie. Survey of sampling-based methods for uncertainty and sensitivity analysis. *Reliability Engineering & System Safety*, 91(10):1175–1209, 2006.

R. L. Iman and J. C. Helton. An investigation of uncertainty and sensitivity analysis techniques for computer models. *Risk analysis*, 8(1):71–90, 1988.

J. Karnon. Planning the efficient allocation of research funds: an adapted application of a non-parametric bayesian value of information analysis. *Health Policy*, 61(3):329–347, 2002.

M. C. Kennedy and A. O'Hagan. Predicting the output from a complex computer code when fast approximations are available. *Biometrika*, 87(1):1–13, 2000.

U. B. Kjærulff and A. L. Madsen. *Bayesian Networks and Influence Diagrams: A Guide to Construction and Analysis*. Springer Publishing Company, 2008.

F. Knight. *H., 1921. Risk, uncertainty and profit*. Houghton Mifflin, Boston, 1921.

K. B. Korb and A. E. Nicholson. *Bayesian artificial intelligence*. Chapman & Hall/CRC Press, 2nd edition, 2010.

J. Kwakkel, W. Walker, and V. Marchau. From predictive modeling to exploratory modeling: how to use non-predictive models for decisionmaking under deep uncertainty. In *Proceedings of the 25th Mini-EURO Conference on Uncertainty and Robustness in Planning and Decision Making (URPDM2010), 15-17 April*, 2010.

G. Li, S.-W. Wang, H. Rabitz, S. Wang, and P. Jaffé. Global uncertainty assessments by high dimensional model representations (hdmr). *Chemical Engineering Science*, 57(21):4445–4460, 2002.

L. A. Maguire, U. S. Seal, and P. F. Brussard. Managing critically endangered species: the sumatran rhino as a case study. *Viable populations for conservation*, pages 141–158, 1987.

S. Mäntyniemi, S. Kuikka, M. Rahikainen, L. T. Kell, and V. Kaitala. The value of information in fisheries management: North sea herring as an example. *ICES Journal of Marine Science: Journal du Conseil*, 66(10):2278–2283, 2009.

J. B. Oostenbrink, M. J. Al, M. Oppe, and M. P. Rutten-van Mölken. Expected value of perfect information: an empirical example of reducing decision uncertainty by conducting additional research. *Value in Health*, 11(7):1070–1080, 2008.

J. Pearl. Do we need higher-order probabilities and, if so, what do they mean? In *Proceedings of the 1981 AAAI Workshop on Uncertainty in AI*, number 87003, July 1987.

L. Pericchi. Sets of prior probabilites and bayesian robustness. Documentation Section on the website of the Society for Imprecise Probability Theory and Applications (SIPTA), 1998.

H. Raiffa and R. Schlaifer. *Applied statistical decision theory*. Boston, Division of Research, Graduate School of Business Administration, Harvard University, 1961.

H. M. Regan, Y. Ben-Haim, B. Langford, W. G. Wilson, P. Lundberg, S. J. Andelman, and M. A. Burgman. Robust decision-making under severe uncertainty for conservation management. *Ecological Applications*, 15(4):1471–1477, 2005.

A. Saltelli, S. Tarantola, and K. Chad. Presenting results from model based studies to decision-makers: Can sensitivity analysis be a defogging agent? *Risk Analysis*, 18(6):799–803, 1998.

A. Saltelli, K. Chan, E. M. Scott, et al. *Sensitivity analysis*, volume 134. Wiley New York, 2000.

A. Saltelli, S. Tarantola, F. Campolongo, and M. Ratto. *Sensitivity analysis in practice: a guide to assessing scientific models*. John Wiley & Sons, 2004.

A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, and S. Tarantola. *Global sensitivity analysis: the primer*. John Wiley & Sons, 2008.

A. Saltelli, P. Annoni, I. Azzini, F. Campolongo, M. Ratto, and S. Tarantola. Variance based sensitivity analysis of model output. design and estimator for the total sensitivity index. *Computer Physics Communications*, 181(2): 259–270, 2010.

P. J. H. Schoemaker. The expected utility model: Its variants, purposes, evidence and limitations. *Journal of Economic Literature*, 20(2):pp. 529–563, 1982. ISSN 00220515. URL `http://www.jstor.org/stable/2724488`.

M. Sniedovich. Fooled by local robustness. *Risk Analysis*, 32(10):1630–1637, 2012.

M. Sniedovich. Response to burgman and regan: The elephant in the rhetoric on info-gap decision theory. *Ecological Applications*, 24(1):229–233, 2014.

I. Sobol'. On quasi-monte carlo integrations. *Mathematics and Computers in Simulation*, 47(25):103 – 112, 1998. ISSN 0378-4754. doi: http://dx.doi.org/10.1016/S0378-4754(98)00096-2. URL `http://www.sciencedirect.com/science/article/pii/S0378475498000962`.

I. M. Sobol'. Global sensitivity indices for nonlinear mathematical models and their monte carlo estimates. *Mathematics and computers in simulation*, 55 (1-3):271–280, 2001.

H. M. Wainwright, S. Finsterle, Y. Jung, Q. Zhou, and J. T. Birkholzer. Making sense of global sensitivity analyses. *Computers & Geosciences*, 2013.

B. K. Williams, M. J. Eaton, and D. R. Breininger. Adaptive resource management and the value of information. *Ecological Modelling*, 222(18):3429–3436, 2011.

F. Yokota and K. M. Thompson. Value of information literature analysis: a review of applications in health risk management. *Medical Decision Making*, 24(3):287–298, 2004.