

An Introduction to Sensitivity Assessment of Simulation Models

John Norton

Integrated Catchment Assessment & Management
Fenner School of Environment & Society
The Australian National University
Canberra, ACT 2601, Australia

School of Electronic, Electrical & Systems Engineering
University of Birmingham
Birmingham B15 2TT, UK

[*john.norton@cantab.net*](mailto:john.norton@cantab.net)

Highlights:

- A concise introductory overview of sensitivity assessment (SA) methods for simulation models is given
- A broad selection of methods is introduced informally and with no more mathematics than necessary
- The motivation of the methods, their scope and their limitations are discussed
- Derivative-based SA, algebraic sensitivity analysis, analysis of dynamical models, sample-based SA including variance-decomposition techniques, regional SA and SA via emulation are described

Abstract

In view of increasing application of sensitivity assessment (SA) to environmental simulation models, a relatively short, informal introduction to aims and methods of SA is given. Their variety, motivation and scope are illustrated by outlines of a broad selection of approaches. Methods based on derivatives, algebraic analysis, sparse sampling, variance decomposition, Fourier analysis and binary classification are included.

Keywords: overview, sensitivity analysis, simulation models, algebraic sensitivity analysis, sensitivity index, emulators

1 Objective

This paper is an informal, selective guide to aims and methods of sensitivity assessment (SA) for simulation models. The selection is intended to be representative but isn't claimed to be comprehensive. However, some topics which seldom get attention will be discussed, for instance algebraic SA in Section 3.3 and regionalised SA in Section 3.5.5. More detail can be found via the references.

The paper is motivated by increasing and now widespread recognition of two facts. The first is that an environmental (or any other) simulation model's credibility and utility depend on knowledge of how important each parameter is; without SA and some idea of the quality of the model's structure and parameter estimates, we don't know how far to trust any prediction by the model. The second is that the results of SA show where the model needs improvement. The more important parameters may need refinement. Conversely, parts with little influence on the outputs of interest are poorly determined in calibration by those outputs; they should be simplified or removed to leave a model well justified by the data.

2 What is sensitivity assessment?

SA investigates the relations between parameters and outputs of a simulation model. In this context "parameters" are primarily equation coefficients and threshold values in the model, but may also include features of input (forcing) variables, *e.g.* their values if constant, or the time, duration, location, spatial extent or rate of any changes. [Parameters are often called "factors" in SA, and occasionally "inputs", which is confusing]. An "output" is the value of any variable computed by the model or of any feature or statistic extracted from it, such as peak or mean value. The supposition is that each parameter and output can be described by a single number. This is appealingly simple and allows SA to be thought of as examining the shape of the response surface of each output to the parameters. It's also restrictive, of course, and may not be tenable if, in particular, the model has dynamics. In that case, an output depends at any instant on the history of the inputs, which would take a very large (theoretically infinite) number of values to describe fully. What's more, we may well be interested in the whole response to the forcing, measured at a large number of instants. Models with dynamics are considered in Section 3.4.

SA tries to answer such questions as:

- For each output, in what order of importance do the parameters influence it?
- Are there parameters which affect the outputs so little that the model should be rewritten without them?
- How well can the combined effect of a collection of parameters be found by summing their individual effects?
- Conversely, what are the significant interactions between parameters in their effect on an output?
- How closely is an output change proportional to the change in parameter value which causes it?
- More broadly, how does the effect of a given change in a parameter value vary

with that value?

These questions need not involve uncertainty, so SA is not the same thing as uncertainty assessment (UA). UA is in effect SA together with a specification of parameter uncertainties. The two get conflated because SA is often done by looking at the output values due to samples from a given distribution of parameter values, as in most of the methods described in Section 3.5. In any case, some specification of parameter uncertainty, even if only guessed ranges, is needed to define the scope of all but the most narrowly focused SA.

It isn't uncommon to find UA unaccompanied by SA. In most of the system identification literature [Norton, 1986; Walter and Pronzato, 1997; Ljung, 1999], parameter-estimation techniques provide estimates of parameter uncertainty (e.g. covariance) yet rarely try to analyse the sensitivity of the model outputs to those parameters.

This paper calls *SA sensitivity assessment rather than analysis*, as it is much more often done by numerical experiment, looking at the results of model runs, than by analysis of the model's equations. That said, such analysis may be possible if the model isn't too elaborate: see Sections 3.3 and 3.4.

3 SA techniques

3.1 Introduction

3.1.1 Assumed mathematical background

To see how any SA technique works, what assumptions it makes, what its scope and limitations are and what its results say, a little mathematics is needed. Only the bare essentials will be used here: basic calculus, the idea that a collection of items (model parameters, usually) can be viewed as a vector in a space with one axis per item, and the concepts of probability density and expectation.

3.1.2 Notation

In what follows, p denotes parameter and y output; δ means "change in", so δy is change in y , not necessarily small. A collection such as m parameters p_i , indexed by a subscript, may be written as a single entity in boldface, e.g. \mathbf{p} is the vector of elements p_i (in a column, by convention). The rate of change of a single output y with a single parameter p may be approximated, in practice, by $\frac{\delta y}{\delta p}$ where δy is the change in y caused by a finite change δp . When two parameters p_1 and p_2 have interacting effects on y , the rate of change of y with respect to p_1 is $\frac{\delta y}{\delta p_1}$, approximated by $\frac{\delta y}{\delta p_1}$.

Additive effects or components will be written out fully as summations rather than more concisely as vector inner products. For example, if interactions can be ignored and the parameter-to-output relations are smooth, the effect of small changes δp_i on y is approximated by $\delta y = \sum \frac{\delta y}{\delta p_i} \delta p_i$.

3.1.3 Motivation of SA techniques

There are a large number of SA techniques: ways to vary parameter values and examine their effects on the outputs. One might well ask why there are so many, when all you need to do, apparently, is try a lot of parameter values in succession and record the output values. The answer is that unless you are only interested in variation of a few parameters at once, over small ranges, it would take an enormous number of model runs to explore the parameter-output relations thoroughly. For example, to cover all combinations of only 5 values each of 10 parameters, 5^{10} simulation runs are needed. At, say, 1 second per run that would take over 16 weeks. We must be much more economical. Another factor, seldom given due weight, is the difficulty of making sense of a huge volume of sensitivity results. Again we have to be selective. Two possibilities are to examine results from a relatively small set of parameter and output samples, in the hope that they are representative, or to analyse the model algebraically. Where it can be done, algebraic analysis is likely to give more insight than a heap of numbers from many model runs. Another way to keep the volume of results manageable is to compute sensitivities as quantities averaged over the whole credible range of the parameters. This is called global SA, in contrast to local SA of sensitivities at particular parameter values.

We start with the simplest (and most naïve) technique and go on to methods which require more thought but yield a great deal of useful information while being economical in computing.

3.2 One-at-a-time perturbations

The simplest idea is to perturb one parameter at a time and see how much the output changes. For now, think about a change from x to $x + \Delta x$ in one parameter, causing a change in a particular single-number output from y to $y + \Delta y$. [Section 3.4 will look at time-varying outputs]. Although $\Delta y / \Delta x$ says something about how sensitive y is to x alone, it has three limitations: (i) it may vary with the size of Δx i.e. the cause-effect relation may not be linear, (ii) it may vary with other parameters, i.e. there may be interaction, and (iii) it depends on the units employed for x and y . We can get rid of this dependence by expressing the changes as *proportions* and the result as the *normalised sensitivity* $\frac{\Delta y / y}{\Delta x / x}$. Alternatively, if we knew the uncertainty in each parameter, we could express a change in a parameter value in terms of its standard deviation or range of uncertainty. To avoid specifying the size of Δx , the sensitivity is formally defined through the rate of change approached as the parameter change is made ever smaller, giving normalised sensitivity $\frac{1}{y} \frac{dy}{dx} \frac{x}{1}$. Its value is generally valid only at whatever values x and the other parameters have, and relates only to changes in x small enough for any non-linearity in the y - x relation to be ignored. Normalisation might be objected to on the grounds that when the relation is a straight line not through the origin, it turns a constant sensitivity into one which varies with x . That's true, but on the other hand one may well prefer to think of changes, errors and uncertainties as proportions (or percentages) rather than absolute values.

Although defined for infinitesimal changes, $\frac{1}{y} \frac{dy}{dx} \frac{x}{1}$ is estimated from the result of a finite parameter change: with all the parameters set to selected values, run the model and note y , change x alone by a chosen amount Δx and rerun the model, note $y + \Delta y$ and get roughly

as Δ . Perturbing m parameters in turn, all from the same start, will take $m+1$ runs.

The first problem in this seemingly straightforward method is to decide on the size of Δ : small enough for Δy to be close to Δx , or big enough to show the effect of realistic uncertainty in x ? A small parameter change has two potential snags. The first is doubt over how linear the model is, and thus whether the result of a small change is a useful guide to the effects of larger changes. Second, calculation of $\Delta y / \Delta x$ may be ill conditioned, with a small difference of large quantities. For these reasons we might prefer a larger parameter change, comparable with the uncertainty in x . If we don't know the uncertainty, we have to try various values of Δx to find what causes a significant Δy . After choosing Δx and calculating the sensitivity $\Delta y / \Delta x$, we still have to check how far the relation between y deviates from a straight line over the credible range of x . A lot of trial and error may be needed to choose a range for each parameter and decide how finely to subdivide it.

One-at-a-time perturbations have the big weakness that they give no idea how much the sensitivity to any particular parameter varies with the values of the others, i.e. how strong the interactions are in the effects of varying the parameters. The influence of two-parameter interactions in the relation between parameters x_j and a single output y is shown by the second term of the truncated Taylor series

$$(1)$$

This expression assumes that the relation between parameters and output has no jumps, kinks or abrupt changes in curvature, and the cutting-off after the terms assumes that any interactions between three or more parameters have a negligible effect on y .

The term of (1) in Δy gives the effect of interaction between x_j and x_k and lets us define normalised second-order sensitivity, for $j \neq k$, as

$$(2)$$

[The $\frac{1}{2}$ disappears because (1) includes terms in both x_j and x_k]. For $j = k$, to deal with the curvature of the relation between x_j and y ,

$$(3)$$

To find the second derivative $\partial^2 y / \partial x_j^2$ experimentally,

$$(4)$$

If we reuse the results which gave the first derivatives, (4) needs only one extra run for each of the $m(m-1)/2$ combinations (j,k) with $j < k$ and the m with $j=k$, in all $m(m+1)/2$ extra runs. It thus takes $m/2$ times as much computing to estimate all the second derivatives, and hence two-parameter-interaction sensitivities, as the one-parameter sensitivities (often called "main effects").

Again the snag is that the results apply only for the parameter values that produced them. What's more, it's quite possible that higher-order interactions make (1) only part of the story.

3.3 Algebraic “no box” SA

SA is usually performed as if the model were a “black box”, with only designated parameters and outputs accessible. However, the model isn't a “black box”; we know its equations and in principle can analyse them. When not too complicated, this alternative to numerical experiment can provide highly informative results with little or no computing. The model may be so complex that analysis is unattractive, but often at least part of the model can be analysed algebraically. Let's call it “no box” SA (Norton, 2008).

With luck, we can derive an expression for the output change which is valid for parameter changes of *any* size from *any* initial value.

Example

Model with inputs .

Subtract from to get the effect of any specified changes in on the output:

(5)

This is *exact*, whatever the sizes of the changes. It's easy to see some qualitative things from it, even without finding the normalised sensitivities:

- is linear in at any given and is unaffected by , or
- is linear in at any given , and , but depends on their values
- is non-linear in (in a way not obviously well approximated by finding)
- there are no interacting effects on y of two or more parameter changes.

One-at-a-time changes in would have given exact values for , but doesn't approximate well unless .

It's worth emphasising that (5) is obtained by first principles, without any use of calculus. □

This example illustrates how no-box analysis can yield insight through *general, exact* expressions for the sensitivities, as well as avoiding any need for model runs. We would be lucky to gain as much insight by black-box SA, even if we inspected the results of a large number of runs.

3.4 No-box SA of a dynamical model: influence equations

So far we've looked at models whose outputs depend only on a limited number of parameters (some of which may apply to input variables) in a static set of relations. Many simulations in fact have dynamics: the outputs at any instant depend on earlier input behaviour. In other words, the model has memory. It's also likely that the entire behaviour of the outputs as time goes on, not just a modest number of isolated values,

is of interest. Clearly SA of such a model has more to it. Chemical reaction kinetics has a considerable literature on the topic (Atherton et al., 1975; Horenko et al., 2005; Turanyi, 1990, 2008).

Most such models are written, or can be rewritten, as a set of rate equations for the dynamics, together with some algebraic, non-differential equations. Such a model has a vector *state equation*

$$(6)$$

for the dynamics. Here state vector \mathbf{x} consists of a selection of variables whose values at the start time are enough to specify fully the condition of the system, \mathbf{u} consists of input variables and \mathbf{p} of parameters. That leaves a set of algebraic equations to describe how the outputs are connected (instantaneously) to the state variables. These equations form the vector *observation equation*

$$(7)$$

Here \mathbf{u} appears only if some of the influence of input on output is instantaneous, as well as through the dynamics. There are systematic ways to select variables making up the state and to turn the original model into state-space form, producing the (matrix) functions \mathbf{f} and \mathbf{h} . The main limitation of state-space models is inability to deal exactly with pure delays. However, delays can be incorporated approximately by use of extra state variables. Moreover, with time-sampled inputs and outputs, as in the great majority of environmental models, delays which are integer multiples of the sampling interval can be handled precisely.

The simple form of the state-space model (6) and (7) makes its SA in general terms algebraically uncomplicated. If we differentiate (6) with respect to \mathbf{p} (not forgetting the effect of \mathbf{p} via \mathbf{x}) and take the inputs to be independent of the parameters, we get *influence equations*

$$(8)$$

This is a set of rate equations saying how the unnormalised sensitivities of the state variables to \mathbf{p} evolve over time. So long as \mathbf{p} doesn't include any initial state value, starts from zero since the initial state is fixed. We solve (8) for by numerical integration, the same sort of process as solving (6), *i.e.* running the dynamical part of the model, but for many more equations. Once \mathbf{x} is known, the output sensitivities follow from the differentiated version of (7):

$$(9)$$

This all looks easy, in principle.

Example

We examine the same model as in Section 3.3 but with dynamics between the non-linearity producing and the multiplication by :

$$(10)$$

The model says that affect y instantaneously but also influences the output through its effect on the rate of change of x . The model is already in state-space form, with only one state variable (so \mathbf{x} is x) and one output, so is a vector with elements and

similarly for \mathbf{y} . Differentiating the state and observation equations as in (8) and (9),

$$(11)$$

so, with \mathbf{y} initially zero, throughout. If \mathbf{p} is constant, we can integrate \mathbf{y} analytically to get \mathbf{y} . As time goes on, \mathbf{y} tends from zero towards \mathbf{y} at an ever slower rate. If instead \mathbf{p} varies in a complicated way with time, we have to do the integration numerically. To find \mathbf{y} once we have \mathbf{p} , we need \mathbf{x} also, found by integrating the state equation, i.e. running the simulation. In any case, \mathbf{y} , \mathbf{p} , and hence the sensitivities, vary with time. \square

It isn't always quite so easy. First, if there are n state variables, \mathbf{A} is an $n \times m$ matrix, so there are m times as many influence equations in (8) as state equations in (6). Similarly there are m times as many in (9) as in (7). Luckily, in most cases many elements of \mathbf{A} and \mathbf{C} will be zero. Second, in (8) and (9) \mathbf{p} , \mathbf{u} and t vary with \mathbf{x} , \mathbf{p} , \mathbf{u} and t in general, so does \mathbf{A} , and everything on the right-hand side of (9) does, hence \mathbf{y} does. For the conditions applying in any one model run, this is no great problem; we just update \mathbf{x} , \mathbf{p} , \mathbf{u} and t in (8) and (9) as we go. However, it may be hard to interpret sensitivities which vary with the variables as well as with the parameters and time. Third and most important, again we only get *local* sensitivities, valid for small variations about specified conditions. To explore them across a range of parameter values, initial conditions and/or inputs, we may have to generate and examine a huge volume of results. For changes in the parameters big enough for the local sensitivities to vary significantly, we could add up the output changes due to a series of small parameter steps $\Delta \mathbf{p}$, in other words integrate (8) and (9) over the overall change in \mathbf{p} , but it is much easier just to rerun the model at the changed parameter values.

One way to reduce the volume of results, at some loss in information, is to find the sensitivity of an objective function combining all the outputs, e.g. a time-varying weighted sum of squares of relative (proportional) output changes. Results can be reduced further by integrating over time if we are content to know average sensitivity over a period.

The following sections discuss ways to employ fewer runs in exploring sensitivities which vary with the simulation conditions. From here on, we'll assume that sensitivities are independent of time; in a dynamical model they might be made so by confining attention to a fixed instant or averaging over some period.

3.5 Sampling-based SA

3.5.1 Introduction

We want to explore sensitivity over the whole range of credible model parameters with an acceptable computational load. Exhaustive, evenly spaced sampling of all parameters looks hopeless, even on a coarse grid, unless the model is small; two extreme values and one in the middle for each of 20 parameters give almost 3.5 billion grid points. The best we can do is sample the output at as many points as we can afford, trying to spread them well, and reduce the volume of results to be

examined by averaging in some way over the whole range of the parameters, *i.e.* do global, rather than local, SA. In global SA, as in other areas where the relative influence of a number of factors in a complex process is investigated by sampling, there are many ways to address the experiment-design problem of achieving a good compromise between sample size and informativeness of the results (Kleijnen, 2005; Saltelli et al., 2008b, pp. 53-96).

If we are prepared to swallow the idea that model parameters are random variables, any knowledge we have of their probability density function (pdf) can be used to concentrate on the most likely regions, as in Section 3.5.2. Once we adopt this view, an obvious way to avoid the difficulty of comprehending results at many points is to define sensitivities via expected values (probability-weighted averages) of some measure of variation. That is what the techniques in Sections 3.5.4 and 3.5.5 do. [In doing so, they turn SA into uncertainty analysis; some authors do not distinguish the two]. Note, however, that Latin hypercube sampling, Section 3.5.3, and regional sensitivity analysis, Section 3.5.6, are global yet may not regard the parameters as random variables.

If all we know about each parameter is its credible range, random sampling with uniform probability density over the range of each seems logical. However, it isn't ideal, as samples from an m -dimensional uniform distribution leave random-sized holes, some big, in parameter space; by chance some regions are very much better covered than others. There are several techniques which hope to mitigate this problem.

3.5.2 Importance (stratified) sampling

The idea is to give each of a number of regions in parameter space an equal quota of samples, randomly distributed within each region. The regions are defined according to both the estimated/guessed probability density of the parameters (uniform or Gaussian by default) *and* the importance of the resulting outcomes, so we can ensure adequate coverage of parameter values around where we expect critical features of behaviour (Castaings et al., 2012). The regions need not have equal probabilities, although we must include the probabilities in any analysis.

3.5.3 Latin hypercube sampling

The impressive name dignifies a simple strategy. For each parameter of the m making up \mathbf{p} , its range is split into N equal-probability divisions and one sample taken in each division. An m -vector of parameter values is then made up by random choice from the samples for each parameter. The next vector is formed by random choices from the $N-1$ remaining samples, and so on until we have used up all the samples. We thus generate sample points in only N of the N^m subdivisions, in contrast to one or more per subdivision in importance sampling.

As Latin hypercube sampling ensures full coverage of every parameter's range, it can be effective in revealing non-linearities through scatter plots, mainly when one or two parameters dominate the effects (Manache and Melching, 2008).

3.5.4 The Morris screening method

An extreme case of trading coverage against computing load is the Morris method (Morris, 1991). For models with many parameters, its rôle is to screen the parameters to find out which to include in more detailed SA (Ruano et al., 2012). With a modest number of model runs, it ranks parameters according to influence on each output and extent of non-linearity. The hope is that many of the lower-ranked can be left out of later assessment by a more computationally expensive technique.

The method works by stepping m parameters along r trajectories, where r is typically less than m . Each trajectory consists of m single-parameter steps. The parameters are all perturbed once, in turn, on the trajectory. The step size is a fixed integer multiple of a given integral fraction of the factor's range (usually half the range), with half the steps positive. The starting points are random and uniformly distributed, and the parameters are stepped in random order. The changes in output due to the r changes in each parameter alone are treated as a sample. A large mean of absolute values of the changes (Campolongo et al., 2007) indicates large influence of that parameter. Large standard deviation indicates that the effect depends strongly on the parameter values, implying strong non-linearity (including multilinearity due to products of parameters, invisible in single-parameter perturbations).

There is scope for selecting trajectories for good spread, preserving the method's economy and its ability to rank sensitivity and detect non-linearity (Campolongo et al., 2007; Norton, 2009). In testing a selection method, small examples may give misleading results (Norton, 2009).

3.5.5 Variance-based sampling methods

3.5.5.1 Variance decomposition, variance ratios and sensitivity indices

So far, computing load has limited how thoroughly we can explore the parameter-output relations. A radical way to reduce the load is to restrict the nature of the information sought. Variance-based SA methods only investigate how much of the overall variability of an output y is due to variability of the parameters, singly or in combinations. The variability is thought of as being random and measured by variance: the mean of the squared deviations from the mean. Squaring emphasises the larger deviations and takes equal notice of equal deviations above and below the mean. Because variance is a probability-weighted average over the entire range considered for all parameters, details of local behaviour are lost; on the other hand, such a summary of behaviour is at the least a useful guide to where to focus a narrower analysis looking for details.

The trick is to estimate, without too much computing, the output variances with and without contributions from various parameter sets. To get down to specifics, fix some subset of the parameters \mathbf{p} , leaving free all the other factors which make output y vary. This causes y to become, say, with mean \bar{y} . The fixing of \mathbf{p} is assumed not to bias the output, so say \bar{y} . The variance σ_y^2 due to all influences on y can be split into the variance of \bar{y} and the mean, over all possible \mathbf{p} , of the variance left after \mathbf{p} is fixed:

$$=+ \tag{12}$$

[As this isn't obvious, here's a proof. It relies on the fact that for a random variable x , so long as the expected values exist (*i.e.* the integrals which yield them don't blow out to infinity; for bounded model variables, they don't). Put μ_x for x in this handy result to see that μ_{y^2} , and μ_y^2 for x to see that μ_{xy} . Add these two expressions to get $\mu_{y^2} - \mu_y^2$, which is $\text{var}(y)$ by the handy result. \square]

The term of the decomposition of $\text{var}(y)$ shows how far μ_x accounts for the variability of y , that is, how strongly x influences y . The remainder, the mean variance of μ_x , indicates the influence of all other factors. The variance ratio $\frac{\mu_{y^2} - \mu_y^2}{\text{var}(y)}$ $\tag{13}$

is thus a measure of the sensitivity of y to x . Being dimensionless, the ratio is independent of units. Also, it lies in the range from 0 (no effect of x on y) and 1 (x accounts for all the variability of y). The variance-based first-order *sensitivity index* of y to x on its own is $\frac{\mu_{y^2} - \mu_y^2}{\text{var}(y)}$.

With interaction possible, a more comprehensive measure of x 's importance is the *total sensitivity index* of its influence on y both by itself and through all interactions in which it takes part. If μ_{-x} , the collection of all parameters except x is frozen then $\mu_{y^2}(\mu_{-x})$ is the proportion of the variance of y left once the part due to x alone is accounted for, so $\frac{\text{var}(y) - \text{var}(y|\mu_{-x})}{\text{var}(y)}$ $\tag{14}$

where the second step uses (12) with μ_{-x} as μ_x . The practicality of these sensitivity indices depends on whether numerical integration to estimate the means and variances can be carried out with a bearable amount of computing. Saltelli et al. (2010) discuss in detail how to compute total sensitivity indices. More generally, variance-based SA, or any other sampling SA technique which averages over the whole range of credible parameter values, ends up having to evaluate multiple integrals over that range. This isn't the place to discuss Monte Carlo integration, based on pseudo-random sample sequences, and more rapidly converging quasi-Monte-Carlo schemes (Lemieux, 2009) which use low-discrepancy sequences, but the important point, here and in other SA sample-based SA methods, is that there are sampling schemes with far more uniform coverage than simple sampling from a uniform distribution (Press et al., 1992, pp. 309-315; Saltelli et al., 2008b, pp. 82-89). For example, Damblin et al. (2013) describe optimised Latin hypercube sampling and space-filling designs.

3.5.5.2 The Sobol' method: decomposition of the parameter-output relation and output variance

This approach takes a broader look at measuring sensitivities in terms of proportions of the output variance due to each parameter individually and to combinations. The parameters are taken as independent and for the moment each equally likely, after any necessary scaling and shifting, to be anywhere in the range [0,1]. The relation between output y and parameter vector \mathbf{p} is split into components with increasing numbers of the parameters:

$$\tag{15}$$

If μ_j and each of the other components of \mathbf{p} integrates to zero over $[0,1]$ with respect to each of its parameters, then any two of the components μ_j are orthogonal to each other, i.e. the integral of their product over the whole of \mathbf{p} is zero. [To see this, integrate first with respect to a parameter that appears only in one of the components, say the first (as at least one does). The other component is constant while we integrate, so the integral is that constant times the zero integral of the first component]. The individual components in (15) are given by

$$\begin{aligned}
 &= \\
 &= \\
 &\text{etc.}
 \end{aligned}
 \tag{16}$$

The orthogonality greatly simplifies the variance of the output; in easy stages,

$$\tag{17}$$

where $q(\mathbf{p})$ is the (uniform) joint pdf of the parameters and \mathbf{I} is the unit box covering 0 to 1 for every parameter. In the next-to-last step, when we write out the squared sum term by term, all the cross-products integrate to zero (they're orthogonal). In the last step, we needn't consider the means of the μ_j 's, μ_k 's and so on, as they're all zero. The assumption of uniform parameter distribution isn't essential; any other pdf can be absorbed into q and its components in the integrations.

Equation 15 goes under various names, most often the Sobol' decomposition (Sobol', 1993). We can define sensitivity as variance ratio as in (13); dividing (17) by V ,

$$\tag{18}$$

where μ_j^2 is defined as μ_j^2 , μ_k^2 and so on. This variance decomposition crucially displays all interactions, i.e. the influence of all combinations of two or more parameters. As noted earlier, the task of evaluating the integrals in (17) relies on finding an effective Monte Carlo scheme.

3.5.5.3 Fourier Amplitude Sensitivity Testing: FAST

This technique is one way to greatly reduce the amount of computation in finding variance-based sensitivity indices. Consider computing the variance and mean of the output:

$$\tag{19}$$

where q is the joint probability density function of the parameters. If all values in the range of μ_j are equally likely, q is constant; we'll assume so, for simplicity. Numerically, integration as in (19) has to sum over many sample points covering the ranges of the parameters, a great deal of computing. The FAST technique reduces it by making the parameters vary all together, depending on a single variable s . For $j=1, 2, \dots, m$, parameter μ_j is required to be some chosen function with s ranging from $-\pi$ to π . The parameters thus vary in a concerted way along a prescribed path in parameter space. Integer k_j fixes a frequency, since μ_j does k_j cycles as s goes from $-\pi$ to π .

There are two reasons for making the parameters vary like this:

- (i) with the k_j 's fixed, all the parameters are function of the single variable s , so multiple

integrals over \mathbf{p} as in (19) become integrals over a single variable, far less work to evaluate. To evaluate the integral, we sum over sample values of y along this path, instead of over all points of an m -dimensional grid of values of y . The path can be made to cover the entire range of all the parameters. There are several possible choices of $G(\cdot)$: see below;

(ii) for any two sinusoids with frequencies which are different integer multiples of a base frequency, the integral of their product over a cycle of the base frequency is zero. This makes integration to find the variance of y much easier. Essentially the idea is to approximate $y(\cdot)$ by a sum of sinusoids at frequencies given by the ω_j 's and other multiples of the base frequency, then

$$(20)$$

where the A 's and B 's are the Fourier coefficients

$$(21)$$

and M is chosen to give an acceptable approximation.

[The first step in (20) again uses the handy result *variance = mean-square value - mean value squared.*]

In a more recent version of FAST, *extended FAST* (Saltelli, Tarantola and Chan, 1999; Wang et al., 2013), each parameter is standardised so that its range is 0 to 1, then s_j is $\frac{\pi}{2}$. As s goes from $-\pi$ to π , s_j does cycles of a triangular wave with peaks 0 and 1. Altogether the parameters follow a zigzag path and cover the whole range of each uniformly (but not reaching extremes at the same time) and end up where they started. The triangular wave can be resolved into a fundamental at the same frequency and harmonics at integer multiples of that frequency (Fourier components). They will give rise to output components at those frequencies and integer multiples, and their interaction with Fourier components of other parameters' variation will produce output components at the sum and difference frequencies (since e.g. $\sin(\omega_1 t) \sin(\omega_2 t) = \frac{1}{2} [\cos(\omega_1 - \omega_2)t] - \frac{1}{2} [\cos(\omega_1 + \omega_2)t]$). With suitable choice of the ω_j 's, the components of the output variance due to θ_j can be pulled out readily (by computing the output's power spectrum in terms of its Fourier coefficients; never mind the details). We can thus find the expected fractional output variance reduction you'd get if θ_j were known (fixed), and the total output variance fraction due to θ_j on its own or in combination with other parameters, found as $1 - \text{fraction due to all interactions not involving } \theta_j$. The output samples in these computations are uniformly spaced along the \mathbf{p} path. They must be at intervals under half a period of the highest-frequency component present (which may depend on non-linearity in the model) to avoid loss or misinterpretation of information by aliasing.

This scheme takes samples in a regular pattern. As in all economical sampling schemes for SA, there's a risk of missing some part of parameter space where sensitivities are very high or low or change rapidly. Another limitation of FAST is that it makes the parameter variations mutually independent (orthogonal, to be mathematical), so it may give unrepresentative results for a model in which parameter variations or errors are mutually correlated, e.g. because they are spatially linked. This is a problem for any technique which ignores such correlation, and to avoid it one needs insight into the physical circumstances modelled. The Sobol' method can be extended to tackle correlation by applying the parameters' joint pdf in computing the conditional expectations and multidimensional integrals in section 3.5.5.2. However,

variance decomposition and interaction terms cannot be derived as neatly when parameter variations are correlated.

A variation of FAST (Tarantola et al., 2006) reduces the number of model runs by employing the same frequency for all factors, randomly permuting the set of samples of s to generate the coordinates of the parameters. This random balance design technique produces the first-order, variance-based sensitivity indices with a fraction of the computing required by FAST for similar accuracy.

A recent development is the “distributed evaluation of local sensitivity analysis” (DELSA) approach, where local analysis methods based on derivatives are used to examine the distribution of sensitivity across the parameter space (Rakovec et al., 2013).

3.5.6 Regionalised SA (RSA): binary classification of Monte Carlo results

This is a quite different approach, devised in the 1970’s (Hornberger and Spear, 1980; Spear and Hornberger, 1980). It pictures a parameter vector \mathbf{p} as a point in m -dimensional space whose coordinates are the values of the m parameters which make up \mathbf{p} . For most purposes, it’s good enough to think of \mathbf{p} in three dimensions even though in practice m is large.

RSA asks the question “What spread of parameter values meets every one of a list of requirements?” For instance, a requirement related to SA is to keep each of one or more outputs within a specified range. Another possible requirement, for a time-varying output, is to be above a given value or between given bounds for a given part of the time. The method tests sample values of the m parameters making up \mathbf{p} against all the requirements. Those points which meet them all are accepted and the rest rejected. Eventually the cloud of accepted points in parameter space gives an idea of the location, extent and shape of the *feasible set* of all values which meet all requirements. The same idea has been widely used to identify acceptable parameter values in models with dynamics (Norton, 1987, 1996; Milanese et al., 1996). There, the parameters must make the model output match a sequence of output observations to within specified bounds. Those requirements are turned into constraints on the parameters, the active ones of which define the boundary of the feasible set. In both RSA and parameter identification, the aim is to identify, via the feasible set, critical parameters or parameter combinations and uninfluential ones.

A notable feature of RSA is that the basic idea of accepting and rejecting parameter samples doesn't depend on a prior parameter pdf and doesn't generate preferences (e.g. as posterior probability densities) among the feasible values. In this sense it is pure SA. The shape of the feasible set may indicate those parameters or combinations to which the output is most sensitive, and others to which it is quite insensitive and which might be written out of the model. As noted in the next paragraph, complexity of the feasible set may make this difficult or even impracticable even without considering computational load, but let us have a look at what is involved. First we have to scale the parameters so that equal distances in parameter space are equally significant. If then the set is thin in direction α , i.e. the feasible points have a small

spread in the direction of a line from the origin to some point α with coordinates α , then the output is highly sensitive to α . Conversely if every possible transit of the feasible set in some direction β covers a specific large feasible range of α , then the output is little influenced by that parameter combination. We might therefore fix it, e.g. at the middle of its feasible range. That allows one parameter to be written in terms of the others and thus eliminated from the model. However, it is not obvious how to identify α or β . The feasible set's extent (thickness) in any specified direction is the longest projection in that direction of the line joining two samples, over all pairs of feasible samples. To minimise the thickness by varying direction γ , say, we can fix its length, say to 1, leaving $m-1$ degrees of freedom to search over, still a lot of work. It's much easier (and much less informative) to measure the overall sensitivity to a single parameter via the difference between the largest and smallest feasible values of α . Another relatively easy case is that the feasible set is thickest along the longest of all lines joining pairs of feasible samples. Its direction is a candidate for β but it's a heavy task to check if the feasible range of α includes some large range throughout the feasible set.

If we are lucky and the feasible set is one not-too-complicated lump (not markedly non-convex), we may be able to fit an object such as an m -dimensional box (orthotope) or ellipsoid tightly around the feasible set then examine the lengths of its axes. This analytically and computationally simple approach is popular in parameter identification, where the feasible set is convex if the model is linear in its parameters. It is clearly analogous to principal-component analysis.

All of this assumes that the feasible set isn't too complicated. It's possible that interaction due to non-linear combinations of parameters makes the feasible set complicated. If we're unlucky, it may be folded in places, have holes or even consist of two or more disjoint sections. Think, for instance, of a partly folded pancake or an American doughnut. In such a case, it may be both thin and thick in the same direction, i.e. the output may be highly sensitive to a parameter combination in some range and insensitive over another. RSA, like other SA methods, faces the problem of interpreting complex behaviour of the model.

Spear, Grieb and Shang (1994) recognise the difficulties outlined above and describe an alternative RSA approach to identify high-sensitivity parameter combinations. It samples the parameters according to a prior pdf and operates essentially by density estimation on the collection of samples, the density being the local proportion of samples found to be feasible. It aims to partition parameter space into small regions with high densities of feasible points and large ones with low densities. The boundaries of these regions and the parameters determining them allow analysis of local sensitivity. Spear et al. find in practice that relative sensitivities, including interactions, vary greatly between high-sensitivity regions.

There are other practical limitations of RSA. A good deal of initial trial and error may be needed to find *any* samples which meet all the requirements. For instance, elaborate models of processes with fairly uncomplicated overall behaviour are often met in environmental applications; they typically have acceptable parameter values

spread over a wide range, since some combinations are not much restricted by the requirements, yet the feasible set is thin, with at least one combination critical. The feasible set then occupies only a small part of the volume of the region defined by the test ranges of the parameters. As a result, the proportion of accepted samples may be tiny and much of the computing wasted.

3.5.7 Use of an emulator

As mentioned above, a complex model may represent relatively simple and regular behaviour, at least for some inputs and outputs over some parameter ranges. This can easily happen when the model is arrived at by aggregating sub-models without ever asking whether simplification is possible. In such cases, there is a good chance that a much simpler model, an emulator, can be fitted to its sampled parameter-to-output behaviour. SA can then be performed on the simpler model, with very large computational saving. Among the forms of model employed for emulation are Gaussian processes (Oakley and O'Hagan, 2004), where the samples are fitted by kriging, and arbitrary polynomial chaos expansion (Crestaux et al., 2009; Oladyshkin and Nowak, 2012), where the fitting is by orthogonal polynomials. Regression-based smoothing techniques for emulation and SA are described by Storlie et al. (2008) and Ratto and Pagano (2010).

The limitation of SA by emulation is that the emulator is, of course, only as good as the sample set which it fits. Use of an emulator for SA implies that the sample set conveys all the simulation-model behaviour whose sensitivity is of interest. In essence, an emulator is no more than an interpolator. Nonetheless, it may be a computationally cheap means to reveal features of the sensitivity which would be hard to see in the samples obtained from the simulation model. Castelletti et al. (2012) provide a framework for dynamic emulation modelling in environmental problems and give over 150 references, within a thematic issue on emulation techniques for the reduction and SA of complex environmental models in Volume 34 of *Environmental Modelling and Software* (Ratto et al., 2012).

4 Further reading

The aim of sensitivity analysis is understanding of how model parameters influence outputs. Where possible, analysis of derivatives or more general algebraic analysis of model equations can provide detailed insight. For more complex models, this may not be possible, but in that case sampling methods (such as sparse sampling, variance decomposition, Fourier analysis and binary classification) can often provide useful information at acceptable computational cost. It is important to be aware of the motivation, assumptions, capability and limitations of each method. This introduction has provided a sketch of them. A number of readable textbooks on SA techniques (Saltelli et al., 2004, 2008a, 2008b) give a more comprehensive picture, in far greater detail. Kleijnen (2005), Ratto et al. (2012) and Press et al. (1992) provide reviews of many aspects of SA, while Turanyi (2008) offers examples of SA in dynamical (reaction kinetics) models.

A complement to SA is investigation of parameter identifiability, i.e. whether a given

model structure allows unique parameter values to be obtained from input and output measurements in the absence of measurement or model-structure error (structural identifiability) or more generally whether satisfactory (unambiguous, accurate enough) values can be obtained from realistic measurements. Shin et al. (2014) review this topic.

5 Acknowledgements

The notes which gave rise to this paper arose from long-standing collaboration with members of the Integrated Catchment Assessment and Management group at The Australian National University. The author thanks them for their interest, encouragement and tact.

6 References

Atherton, R. W., Schainker, R. B., Ducot, E. R., 1975. On the statistical sensitivity analysis of models for chemical kinetics. *AIChE Journal* 21, 441-448.

Castaigns, W., Borgonovo, E., Morris, M. D., Tarantola, S., 2012. Sampling strategies in density-based sensitivity analysis. *Environmental Modelling & Software* 38, 13-26.

Castelletti, A., Galelli, S., Ratto, M., Soncini-Sessa, R., Young, P. C., 2012. A general framework for Dynamic Emulation Modelling in environmental problems. *Environmental Modelling & Software* 34, 5-18.

Campolongo F., Cariboni J., Saltelli, A., 2007. An effective screening design for sensitivity analysis of large models. *Environmental Modelling & Software* 22, 1509-1518.

Crestaux, T., Le Maître, O., Martinez, J.-M., 2009. Polynomial chaos expansion for sensitivity analysis. *Reliability Engineering & System Safety* 94, 1161-1172.

Damblin, G., Couplet, M. and Iooss, B., 2013. Numerical studies of space filling designs: optimization of Latin hypercube samples and subprojection properties. *Journal of Simulation* 7, 276-289.

Horenko, I., Lorenz, S., Schütte, C., Huisinga, W., 2005. Adaptive approach for non-linear sensitivity analysis of reaction kinetics. *Journal of Computational Chemistry* 26, 941-948.

Hornberger, G. M., Spear, R. C., 1980. Eutrophication in Peel Inlet, 1, The problem: defining behavior and a mathematical model for the phosphorus scenario. *Water Research* 14, 29-42.

Kleijnen, J. P. C., 2005. An overview of the design and analysis of simulation experiments for sensitivity analysis. *European Journal of Operational Research* 164, 287-300.

Lemieux, C., 2009. Monte Carlo and Quasi-Monte Carlo Sampling. Springer, New York.

Ljung, L., 1999. System Identification: Theory for the User (2nd ed.). Prentice-Hall, Upper Saddle River, NJ.

Manache, G., Melching, C. S. 2008. Identification of reliable regression- and correlation-based sensitivity measures for importance ranking of water-quality model parameters. *Environmental Modelling & Software* 23, 549-562.

Milanese, M., Norton, J., Piet-Lahanier, H., Walter, E., Eds., 1996. Bounding Approaches to System Identification. Plenum Press, NY.

Morris, M. D., 1991. Factorial sampling plans for preliminary computational experiments. *Technometrics* 33, 161-174.

Norton, J. P., 1986. An Introduction to Identification. Academic Press, London and New York. Reprinted 2009 by Dover Publications, Inc., Mineola, NY.

Norton, J. P., 1987. Identification and application of bounded-parameter models. *Automatica* 23, 497-507.

Norton, J. P., 1996. Roles for deterministic bounding in environmental modelling. *Ecological Modelling* 6, 157-161.

Norton, J. P., 2008. Algebraic sensitivity analysis of environmental models. *Environmental Modelling & Software* 23, 963-972.

Norton, J. P., 2009. Selection of Morris trajectories for initial sensitivity analysis. *SYSID2009 15th IFAC Symp. on System Identification*, paper 431.

Oakley, J., O'Hagan, A., 2004. Probabilistic sensitivity analysis of complex models: a Bayesian approach. *Journal of the Royal Statistical Society B* 66, 751-769.

Oladyshkin, S., Nowak, W., 2012. Data-driven uncertainty quantification using the arbitrary polynomial chaos expansion. *Reliability Eng. & System Safety* 106, 179-190.

Press, W. H., Flannery, B. P., Teukolsky, S. A., Vetterling, W. T., 1992. *Numerical Recipes in C*, second edition. Cambridge University Press, Cambridge, UK.

Rakovec, O., Hill, M.C., Clark, M.P., Weerts, A.H., Teuling, J.J., Uijlenhoet, R., 2013. Distributed evaluation of local sensitivity analysis (DELSA), with application to

hydrologic models. *Water Resources Research* 50, 1-18.

Ratto, M., Castelletti, A., Pagano, A., (Eds.), 2012. Special issue on Emulation techniques for the reduction and sensitivity analysis of complex environmental models. *Environmental Modelling & Software* 34, 1-115.

Ratto, M., Pagano, A., 2010. Using recursive algorithms for the efficient identification of smoothing spline ANOVA models. *Advances in Statistical Analysis* 94, 367–388.

Ruano, M. V., Ribes, J., Seco, A., Ferrer, J., 2012. An improved sampling strategy based on trajectory design for application of the Morris method to systems with many input factors. *Environmental Modelling & Software* 37, 103-109.

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

Saltelli, A., Chan, K., Scott, E. M., 2008a. *Sensitivity Analysis* (2nd ed.). Wiley, Chichester, UK.

Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D., Saisana, M., Tarantola, S., 2008b. *Global Sensitivity Analysis: The Primer*. Wiley, Chichester, UK.

Saltelli, A., Tarantola, S., Campolongo, F., Ratto, M., 2004. *Sensitivity Analysis in Practice: A Guide to Assessing Scientific Models*, Wiley, Chichester, UK.

Saltelli A., Tarantola, S., Chan, K.P.S., 1999. A quantitative model-independent method for global sensitivity analysis of model output. *Technometrics* 41, 39–56.

Shin, M-J., Guillaume, J.A.H., Croke, B.F.W. and Jakeman, A.J. 2015. A review of foundational methods for checking the structural identifiability of models: results for rainfall-runoff. *Journal of Hydrology*, 510, 1-16.

Sobol', I. M., 1993. Sensitivity analysis for non-linear mathematical models. *Mathematical Modelling & Computational Experiment* 1, 407-414.

Storlie, C. B., Helton, J. C., 2008. Multiple predictor smoothing methods for sensitivity analysis: Description of techniques. *Reliability Engineering & System Safety* 93, 28-54.

Spear, R. C., Hornberger, G. M., 1980. Eutrophication in Peel Inlet, 2, Identification of

critical uncertainties via generalized sensitivity analysis. *Water Research* 14, 43-49.

Spear, R. C., Grieb, T. M., Shang, N., 1994. Parameter uncertainty and interaction in complex environmental models. *Water Resources Research* 30, 11, 3159-3169.

Tarantola, S., Gatelli, D., Mara, T., 2006. Random balance designs for the estimation of first order global sensitivity indices. *Reliability Engineering & System Safety* 91, 717-727.

Turanyi T., 1990. Sensitivity Analysis of complex kinetic systems: Tools and applications. *Journal of Mathematic Chemistry* 5, 203-248.

Turanyi T. (Ed.), 2008. Sensitivity analysis in chemical kinetics. Special issue of *International Journal of Chemical Kinetics* 40, 685-768.

Walter, E., Pronzato, L., 1997. *Identification of Parametric Models from Experimental Data*, Springer, London.

Wang, J., Li, X., Lu, L., Fang, F., 2013. Parameter sensitivity analysis of crop growth models based on the extended Fourier Amplitude Sensitivity Test method.