Refining Stochastic Models of Critical Infrastructures by Observation

Stefan Rass
•
Institute of Applied Informatics, Universität Klagenfurt, Universitätsstrasse 65-67, 9020 Klagenfurt, Austria
stefan.rass@aau.at

Stefan Schauer
•
Center for Digital Safety & Security, AIT Austrian Institute of Technology, Lakeside B10a, 9020 Klagenfurt, Austria
stefan.schauer@ait.ac.at

Abstract

The simulation of cascading effects in networks of critical infrastructures (CIs) can be approached in various ways, all of which at some point call for the specification of (numeric) model parameters. Taking stochastic models as one popular class of methods, finding proper settings for the values that determine the stochastic models can be a challenge. In this work, we describe a method of graphical specification of a probability value on a qualitative scale, and how to convert and use the obtained value as a prior for Bayesian statistics. The connection is made to the point of having the initial value specified only as an "initial guess", which can be refined using Bayesian statistics. Eventually, under consistency conditions depending on the application, this amounts to an online learning approach that takes the parameter to convergence towards their true values, based on the user's subjective initial guess, but never challenging a person to give a reliable number for a probabilistic parameter.

Keywords: simulation, cascading effect, risk management, stochastic model, security

I. Introduction

Among the biggest challenges in stochastic models is probability. Scientists often provide people with sophisticated model having beautiful theoretical properties, but left with the highly nontrivial challenge of finding proper values for a set of parameters, many of which are probabilities. What if the person simply does not have these values or cannot reliably estimate them? This work proposes to avoid the issue of pulling numbers "out of the air", by instead resorting to purely graphical method and machine learning to poll or estimate probabilities.

Probabilistic models have the appeal of being often easy to define and plausible to use, yet the intuitiveness of the model specification turns into a difficulty when creating a model instance in many cases. Suppose that the model includes some probability parameter p that "simply" quantifies the likelihood of some event to occur; for example, the impact of an incident on related parts in a system (e.g., a dependent infrastructure). Likewise, we may use a parameter (probability) p to describe the likelihood of a threat along risk analysis, or call p the likelihood for human error to bring the human element into a model. How do we set such values in practice? It is tempting to use them in a model because they are easy to argue and statistics enjoys a solid mathematical fundament, but the practitioner facing the challenge of assigning some reasonable value to the variable p may find this to be an almost impossible task to accomplish reliably. In many cases, the setting of such parameters thus resorts to choices on qualitative scales, say, defining the probability just to be "low" or "high", with the meaning of these values remaining vague or defined by representative standard values specified elsewhere. In many practical cases when people try to *apply* or *use* (not define or invent) stochastic models, the choice of probability parameters is a matter of asking experts for numbers that they simply do not have. This can practically limit the applicability of such models despite any theoretical beauty.

Statistics has lots to offer to people seeking to estimate parameters of stochastic models, since the whole theory of point- and interval estimation is dedicated to the problem of finding values or ranges of values for unknown quantities. Common to most of these techniques is their use of empirical data to compute the estimators. In risk management, and particularly in the context of critical infrastructures (CIs), the situation is just not satisfying the assumptions: data is scarce, and we cannot expect having hundreds of data samples from past incidents in a critical infrastructure (simply because the CI would not have survived the necessary lot of incidents to gather enough data for a statistically reliable estimation).

Instead, we need to come up with a reasonable initial guess for the probabilistic parameters and look for a way to refine that value upon continuous experience. Bayesian estimation thus appears as a reasonable way to go, and this work describes a very straightforward and easy to implement version of such a Bayesian estimation approach, where we explicitly exploit the absence of much prior knowledge as an advantage. Indeed, if there is not too much robust prior knowledge about how a probability parameter should be set numerically, this also means that any choice is as good as the other. While it would not make sense to step forward by just picking parameter values at random, the Bayesian method is much more elegant in letting us choose a prior *distribution* to our own convenience, and – realistically reflecting the uncertainty of the person instantiating the model – leaving the parameter p actually unspecified in the beginning. The actual value for p is then obtained from the prior distribution in first place, and iteratively refined by bringing in experience about the model performance to continuously refine it towards an accurate setting for the real model.

To the end of using that method for model parameterization, we thus have to devise (i) a method to pick a reasonable initial guess for some (generic) probability parameter p (Section II will describe an example model for illustration), and (ii) describe a method to define that guess, which assures that we will eventually end up with the correct value for p over the long run. We dedicate Section III to this.

As a running example, we will pick a specific model to describe critical infrastructure dependencies, to study cascading effects by simulation. Our choice of the CERBERUS model [1] is arbitrary here, and can be replaced by any other stochastic model based on Markov chains, percolation theory, or others. The palette is rich, and we refer the reader to [2–11] for models to which our work may offer an aid to get a practical instance, meaning concrete numeric settings, for the involved probability parameters.

II. The CERBERUS Risk Simulation Model

Consider a network of interdependent critical infrastructures that we represent as a directed graph $G = (V, E)$ with edges $A \rightarrow B$ meaning that CI B somehow depends on CI A. For example, A could provide energy, water, food, transport, etc. for B. To all infrastructures in the (node) set V, we assign one out of k possible operational states, reflecting their degree of "health". Typically, this state ranges from "fully functional" (state 1) to "outage" (state k), with intermediate states from 2 to $k-1$ corresponding to ascending limitations in a CI's service(s). The dependency of a CI B on one or more of its providers may be of arbitrary form and dynamic. For example, a CI may have providers that it vitally depends on, or whose service can be substituted for a limited period of time (e.g., emergency power generators can cover a power outage for some time, until they run out of fuel). Other dynamics of dependency may involve the kind of service more explicitly, say, if CI B relies on online-services of A (e.g., an outsourced data center) in order to coordinate the shipping of goods from another provider C to B.

Commonly, authors distinguish the type of dependency here, dividing it into physical dependencies (e.g., supply with physical utilities), cyber-dependencies (e.g., communication and data exchange), geographic dependency (often physical proximity or reachability), and others (cf. [6,12–15]), including temporal dependencies (that are outside our scope here since we look for the setting of probability parameters).

To study cascading effects in such models, we thus need to describe what happens to an infrastructure if its providers fail. While there is lots of work on understanding dependencies (see [16] for a considerable collection of respective references), quantitative studies on how to describe the parameter value for some stochastic model are rare (not so the models themselves; see the references in the introduction). In this context, we want to highlight the work in [16], where an empirical study on how strong the impact of several critical infrastructures may be on others is provided.

The CERBERUS model uses precisely such information to describe an infrastructure model and cascading effects therein in the following way:

- The behavior of a CI B is described by a bipartite graph (see Figure 1):
    - The top layer has exactly k nodes, one for each operational state in which the CI can be
    - The bottom layer has k nodes per CI A that CI B depends on. That is, each supplier CI A is represented in the graph model as its own set of k nodes, one per operational state of CI A, and every other supplier of B having its own copy of these k input nodes.
- The bipartite inner graph is complete, meaning that there is an edge from each state node of each supplier to the overall state node of CI B. These edges are annotated by probabilities, indicating how likely it is that CI B moves into state j, if infrastructure A is in state ℓ. For each $\ell \in \{1,2,\dots,k\}$, we thus have to specify a probability $p_{\ell j} = \Pr(\text{CI B is put into state } j \mid \text{ CI } A \text{ is in state } \ell)$. If the change is a (deterministic) fixed consequence, we can put $p_{\ell j} := 1$ to model this.
- Since the edges connect only two nodes at a time (the model is a graph, not a hypergraph), the effects of a supplier on B are independent on what other suppliers do. Moreover, B can be put into distinct operational states upon different of its providers changing their state individually. Intuitively, this reflects the real world quite well, since a problem at provider A_1 may cause only slight stress for CI B, while another (independent) problem at provider A_2 may have a substantial impact on B's functionality. Thus, there is an aggregation function being applied on the states that probabilistically follow from the supplier states, which in the simplest case is just the maximum of all possible states that the suppliers may put B into. For example, if provider A_1's failure puts B into state "normal" (i.e., no immediate effect), but supplier A_2's outage causes severe problems in B, the overall state of B is the worst of the two, set to be "severe problems".

This kind of maximum-aggregation assumes that higher state indices correspond to more severe problems (taking the lowest state as the best). Logically, it corresponds to an OR, since B has troubles if at least one of its critical providers fails. This logic can be changed into an AND by resorting to a minimum-value aggregation, causing the state of B to remain "healthy", unless all of its providers fail. The proper choice per infrastructure is up to the application.

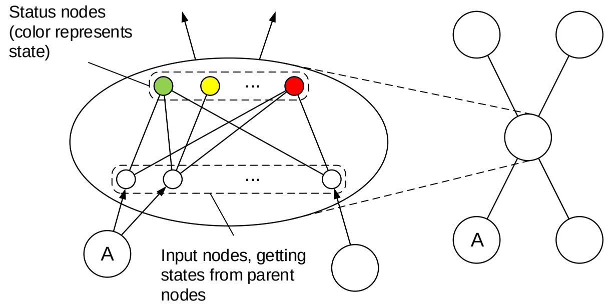

Status nodes (color represents state)
...
...
A
Input nodes, getting states from parent nodes
A

Figure 1 *CERBERUS Model (picture adapted from [1])*

The CERBERUS model includes this simplification to avoid a combinatorial explosion of parameters that would need specification otherwise. For example, the most powerful description of dependency (that includes the above OR/AND dependencies as trivial special cases) is that of a Bayesian network [17]. This approach is similar to the CERBERUS model, however, requires a worst-case exponential number of parameters specified to describe the dependency as a full-fledged conditional *distribution*. The above reduces that number to "only" polynomially many (exactly $k \cdot n$ conditional probability *values*, if k states are used and the CI depends on n other CIs). Since both, A and B have a common set of possible states, the transition regime can be described as a matrix of the general form:

	State of CI B (depending on A's state)			
State if CI A	1	2	…	k
1	p_{11}^A	p_{12}^A	…	p_{1k}^A
2	p_{21}^A	p_{22}^A	⋱	p_{2k}^A
⋮	⋮	⋮	⋮	⋮
k	p_{k1}^A	p_{k2}^A	…	p_{kk}^A

The superscript A is here only a reminder that these transitions relate to infrastructure A, and more such matrices would be required to describe the dependency of B on other CIs. The specification is very much like (though not identical) to a transition matrix of a Markov chain, since in each row, there has to be one target state for CI B. Our problem in the following will thus be the specification of these (many) values, using an initial guess and online learning to refine it.

Again, we stress that the choice of this model for illustration is arbitrary, and replaceable by others. The reader feeling more familiar with Markov chains or other models is safe to think along these lines during the remainder of this work. Indeed, we will become more general than the above in considering the estimation of a whole vector of probability values, constrained to form a probability distribution (thus covering the more complex case of Bayesian network specification too).

III. Model Parameterization: Initial Guesses

In absence of empirical data, the best that we can do is resorting to domain expertise, subjective experience and empirical studies as far as they are available (e.g., [16]). However, the problem remains one where experts have to provide (qualitative or better quantitative) values that are usually hard to obtain. One possibility is getting domain experts into discussion to agree on a common assessment (e.g., using systematic methods such as Delphi and/or opinion pooling [18]), which generally means aggregating different assessments into an object (number) that we can start with – an initial guess. Lossless aggregation into a distribution is also possible and has been described for general risk management in [19]; however, this method is out of our scope here, but mentioned as another option to get a prior distribution for Bayesian updating (met later in Section IV)

I. Graphical Specification of Parameters

To avoid asking people for numbers, graphical ways of specifying probabilities and general risk parameters have been developed. One method aiming to help with the quantification of risk as the product of "likelihood" and "impact" is to let experts draw a "risk rectangle", whose horizontal length reflects the person's (subjective) assessment on a range for the unknown likelihood, and the vertical breadth acts as an interval estimate for the potential impacts; see Figure 2 for an illustration. The area of the rectangle *can*, but with care, be associated with the usual formula $likelihood \times impact = risk$, where both inputs are ranges reflecting uncertainty. Intuitively, the larger the rectangle is, the more uncertain would the specification be, stressing that even for small areas, the width and height still need consideration in their own meaning of uncertainty (a very thin rectangle has small area, yet may express large uncertainty about one of the coordinates).
As an initial guess for a parameter, such a graphical method may serve as a replacement for a number, since the actual numeric value is easy for a computer to compile from the rectangle's coordinates.

In any case, this is just a heuristic and there is no formal or scientific reason (so far) why any such graphical method should deliver more reliable results than a direct specification. It is as such a matter of usability and convenience to specify values in this way. This potential benefit becomes even more evident if we transfer the idea to the specification of a whole matrix of values, say, a transition matrix of a Markov chain. Why not think of the matrix as a rectangular grid, on which our task is to place masses, proportionally to as how likely it is that state i will take the chain into the target state j. Returning to the CERBERUS model above, we would, for each supplier CI A, have one such matrix to tell B's target state based on A's current state.

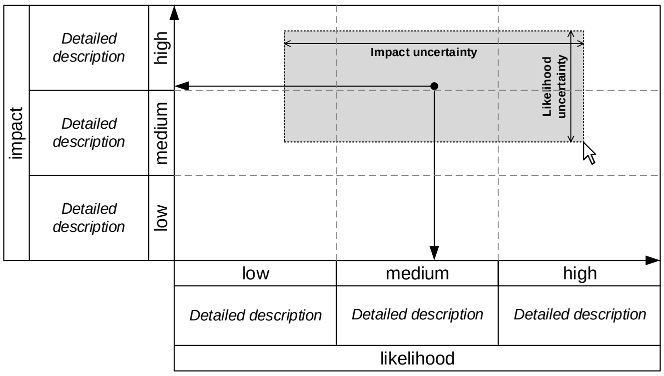
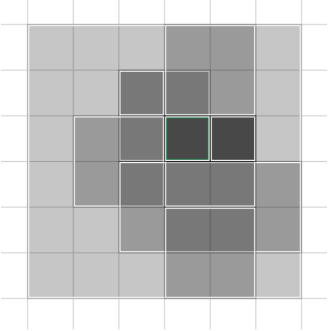


Figure 2 *Graphical Risk Specification Method (picture adapted from* **[20]***)*

The idea is a straightforward extension of the graphical specification from before: assuming that the states are ordered (in ascending or descending levels of criticality), we can go and draw a bunch of rectangles into the grid, which may even overlap, and each of which places some mass onto a cell in the grid, i.e., element in the matrix. The amount of weight being placed is then a matter of how much the rectangle overlaps the respective region. Intuitively, if we draw a rectangle over several cells (horizontally and vertically), we may express something like "any state between i_1 and i_2 may put the dependent CI B into some state between j_1 and j_2" – not becoming too specific on how likely a specific transition is, but only telling what one may think is possible. The more such possibilities are supplied, the more weight accumulates on a cell, and the more likelihood is assigned accordingly. Figure 3 displays the idea, with some example values compiled from the cumulative areas.



$$\rightarrow \quad \begin{pmatrix} \frac{1}{8} & \frac{1}{8} & \frac{1}{8} & \frac{1}{4} & \frac{1}{4} & \frac{1}{8} \\ \frac{1}{11} & \frac{1}{11} & \frac{3}{11} & \frac{3}{11} & \frac{2}{11} & \frac{1}{11} \\ \frac{1}{15} & \frac{2}{15} & \frac{1}{5} & \frac{4}{15} & \frac{4}{15} & \frac{1}{15} \\ \frac{1}{14} & \frac{1}{7} & \frac{3}{14} & \frac{3}{14} & \frac{3}{14} & \frac{1}{7} \\ \frac{1}{12} & \frac{1}{12} & \frac{1}{6} & \frac{1}{4} & \frac{1}{4} & \frac{1}{6} \\ \frac{1}{8} & \frac{1}{8} & \frac{1}{8} & \frac{1}{4} & \frac{1}{4} & \frac{1}{8} \end{pmatrix}$$

Figure 3 *Graphical Specification of a transition matrix*

II. Prior Distribution for Online Learning

Suppose that we have a set of probabilities $p_1, \dots, p_k$ that jointly form a distribution, i.e., satisfy $p_1 + p_2 + \cdots + p_k = 1$. For the example of the CERBERUS model, given a dependency of CI B on A, such a set would be a matrix as outlined above, or at least a single row in it.

Most likely, the initial guess is inaccurate, subjective, not well founded on empirical data or experience, or suffers from other sources of vagueness. This is most naturally so, since we cannot expect an(y) expert to have precise or objectively reliable figures for likelihoods in a quality better than to the best of her/his knowledge.

It is, however, possible to refine and "correct" these initial guesses in the long run by observing the system, tracking the real state changes, and refine our hypothesis iteratively, knowing that it will converge to the "objective" and hence correct probabilities. The mechanism is Bayesian updating of a properly chosen prior distribution, which makes the whole process even computationally efficient and trivial to implement.

Our choice is the Dirichlet distribution, having $k \geq 2$ parameters $(\alpha_1, \dots, \alpha_k)$ satisfying $\alpha_i > 0$ for all $i = 1, \dots, k$, and the probability density function

$$f_{Dirichlet}(x_1, \dots, x_k | \alpha_1, \dots, \alpha_k) = \frac{\Gamma\left(\sum_{i=1}^{k} \alpha_i\right)}{\prod_{i=1}^{k} \Gamma(\alpha_i)} \prod_{i=1}^{k} x_i^{\alpha_i - 1}.$$

The interesting point for our purpose is the fact that this distribution relates to a vector $\boldsymbol{X} = (X_1, \dots, X_k) \in (0,1)^k$ constrained by $X_1 + \cdots + X_k = 1$, so that it can be used to describe a probability distribution. That is, our sought probability vector, the distribution to be specified, is viewable as a sample of the random vector $\boldsymbol{X}$, whose distribution is Dirichlet with the density as above. Under that perspective, we can equate the desired likelihoods $p_i := E(X_i)$ with X_i being the i-th coordinate in $\boldsymbol{X}$.

For the Dirichlet distribution, this expectation is simply

$$E(X_i) = \frac{\alpha_i}{\sum_{i=1}^{k} \alpha_i}$$

Now, suppose that we have an initial guess for the values $p_1, \dots, p_k$; then even without those normalizing to unit sum, we can plainly specify the parameters α_i as $\alpha_i := p_i$ to start with, since the denominator in the above expression is nothing else than a normalization, so that the soinstantiated Dirichlet density, encodes our initial guess for the probability parameters by the component-wise expectations.

Remark: The case for a single parameter is treated only slightly different; noting that above, we require at least two values. If there is only a single probability parameter in question, the prior would be the Beta distribution, having the density $f_{Beta}(x|\alpha_1,\alpha_2) \coloneqq f_{Dirichlet}(x, 1-x|\alpha_1,\alpha_2)$, with the expectation following the same formula as given above. The major (only) difference is that while the Dirichlet distribution describes a set of k probability values, the Beta distribution describes only a single value that is also a probability; in both cases, the last value ($x_2 = 1-x$ or $x_k = 1 - x_1 - x_2 - \cdots - x_{k-1}$) is fixed by its predecessors (not surprisingly so, since we have the constraint of all these values to sum up to 1).

IV. Bayesian Updating

On a level of abstraction, the CERBERUS model is a set of Markov chain instances, where a state transition of a CI triggers another state transition of a dependent CI. Suppose that this switch is observable, i.e., we would note the change in reality, and can relate it to an edge in the model (see Figure 1).

Adopting a Bayesian statistics perspective, the observation is nothing else than data sampled from a distribution whose parameters we seek to estimate. More specifically, consider only the i-th row $\boldsymbol{p}_{i,\cdot}$ in a transition matrix $\boldsymbol{P}$, telling us that if the current state is i, then the next possible states $j \in \{1,2,\dots\}$ will occur with probabilities $p_{i1}, p_{i2}, \dots$. This single row is a categorical distribution, and the values in it are exactly the parameters (the distribution is, in a way, not only determined, but actually directly represented by its parameter set). Now, suppose that an observation is made, which tells that out of the current state i, our system has (physically, in reality) moved into the state j. Formally, this is $\boldsymbol{x} = (0,0,\dots,1,0,0,\dots)$, with only the j-th entry being 1, sampled from the aforementioned categorical distribution $\boldsymbol{p}_{i,\cdot}$ (which in turn is just the i-th row in the transition matrix $\boldsymbol{P}$).

More importantly, this view takes the incoming observations as samples from a 0/1-valued random variable. Such a variable is an indicator, and the expectation of an indicator variable is a probability, thus making the approach meaningful to estimate probability parameters.

Now, let us put this to practice: suppose that we observed the event of our system to have undergone a transition from state i into state j. If the Bayesian prior distribution is a Dirichlet (or Beta), with parameters $\alpha_1, \dots, \alpha_k$ (in the case of a single parameter p to be estimated, we would only have α_1 and α_2, with $p = \frac{\alpha_1}{\alpha_1+\alpha_2}$), the Bayesian update of the row $\boldsymbol{p}_{i,\cdot}$ in the transition matrix P, which is described by a prior distribution with parameter vector $(\alpha_1, \dots, \alpha_k)$, proceeds via the assignment

$$\left(\alpha_1, \dots, \alpha_{j-1}, \alpha_j, \alpha_{j+1}, \dots, \alpha_k\right) \leftarrow \left(\alpha_1, \dots, \alpha_{j-1}, \alpha_j + 1, \alpha_{j+1}, \dots, \alpha_k\right),$$

i.e., only the j-th parameter gets increased by 1. What could be simpler? It essentially amounts to counting the occurrences of each transition! Even if several observations are collected in a data vector, say, $\boldsymbol{d} = (n_1, n_2, \dots, n_k)$ with n_1 observed transitions into state 1, another n_2 transitions observed into state n_2, etc., the update to $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_k)$ would simply be $\boldsymbol{\alpha} \leftarrow \boldsymbol{\alpha} + \boldsymbol{d}$.

The current estimate $\hat{p}_j$ of the j-th (not precisely known) probability parameter p_j vector is for each $j = 1,2,\dots,k$ given as

$$\hat{p}_j = E(X_j) = \frac{\alpha_j}{\alpha_1 + \alpha_2 + \cdots + \alpha_k}$$

Now, let us suppose that we started from initial values (guesses) $\alpha_1^*, \ldots, \alpha_k^*$. What would happen in the long run? If we observe the transition into the j-th state for N times out of $M \gg N$ cases and let $M \to \infty$, then the estimator $\hat{p}_j$ after a total of M updates is

$$\frac{\alpha_j^* + N}{\left(1 - \alpha_j^* + M - N\right) + \left(\alpha_j^* + N\right)};$$

this is easy to see from the fact that we increase the pseudo-count[1] α_j^* for N times, whilst increasing any of the other parameters for the remaining $M - N$ times (whose totality is collected in the term $1 - \alpha_j^* + M - N$). Overall, since the initial guess does not change, the limit is

$$\frac{\alpha_j^* + N}{(1 - \alpha_j^* + M - N) + (\alpha_j^* + N)} \to \frac{N}{M} = p_j,$$

Since this is merely the fraction of "good cases" among "all cases", i.e., by definition the sought probability. The key insight here is that this limit does not depend on the initial guess! That is, no matter if we were wrong with our initial parameter choice (and in most cases, we may have been wrong), the long-run updating will asymptotically "correct" our error automatically. Of course, the speed of convergence depends on how far off the inaccuracy of the initial guess put us away from the real value of p_j. The closer our initial guess has been, the earlier we get into a reasonable proximity of the true value p_j.

Let's also take a closer look at the case of a single parameter: if we don't have a whole Markov chain, but rather a single parameter that describes an event by a probabilistic value, there is no conceptual change to the above. The respective prior has two parameters (α_1, α_2), which we update to $(\alpha_1 + 1, \alpha_2)$ if the event has been observed, or into $(\alpha_1, \alpha_2 + 1)$ if the event did not occur; both cases assume that the parameter p in question describes the probability of the event's occurrence (otherwise, the update would be done with the roles of α_1 and α_2 being switched). We refer to [21] for a fully detailed elaboration of this prior idea, which we here generalized. The reference cited treats the topic in the different direction of using the idea for predictive analytics (see [22] for a survey).

Example for the CERBERUS Model

The application of the above scheme in the CERBERUS model is straightforward, based on what we have: suppose that a history of cascading effects was recently observed in the network of critical infrastructures, or is available from documented cases of incidents or experience. Then, we can consider each part of the chain of events described in the following form: "CI A changed its state from x to y, causing CI B to change its state from u to v". To update our model, we look into the inner model for CI B, which embodies a transition matrix $\boldsymbol{P}_A$ that tells us how likely a change into state v is for CI B, provided that CI A is in state y. Taking that row i of P_A that corresponds to state y, and associating it with its (Bayesian) Dirichlet prior $\boldsymbol{\alpha}_{i,\cdot}^A = (\alpha_1, \ldots, \alpha_k)$, where k ranges over the possible states of CI B, the update is simply an addition of 1 to the j-th coordinate in the vector α, relating to the state v that CI B turned into. The Bayesian update on this set of transitions is done

[1] A pseudo-count is a fractional count value; this term is technically exact here since we may start from a fractional value α_j, but add 1 upon an observation of the respective transition. Thus, although we do count, the counter's value remains fractional at all times; hence it is called "pseudo".

by that point.

Note that here we did not make any use of the previous states x of CI A or u for CI B. This is due to the fact that the change of state for CI A would be subject to an according update of the inner model for CI A (just as described). The prior state of CI B plays indeed no role here.

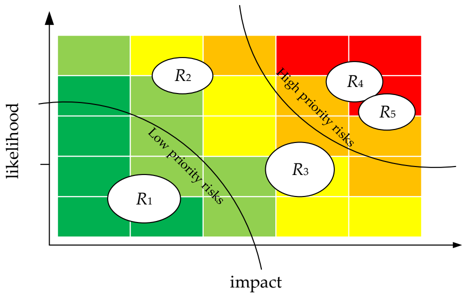
V. Avoiding Numbers – Asking for Ranks

In some cases, we may be able to completely avoid the specification of numbers, and poll people only for ranks. As with CERBERUS, risk management has such cases that we will look into now. One use case (among other possible ones) for the CERBERUS model relates to decision making towards risk mitigation. Considering a framework like ISO31000, a risk manager may roughly follow these steps:

1. Identification of context: this means a clear delineation of what assets we are concerned with, what level of protection is required, and seeks an understanding of external and internal factors with impact on the assets.
2. Risk identification: this is the identification of all threats with the potential of realizing themselves as risks to the previously defined assets.
3. Risk analysis: this is the actual challenge that we are concerned with here, being an *estimation* of impacts and likelihoods, so as to "quantify" risks by the well known rule of thumb

   $$\text{risk} = \text{impact} \times \text{likelihood}$$

   This formula has the statistical appeal of resembling an expected value, since it is easily extended into a weighted sum of impacts, each related to another threat on the list from the risk identification step. Asking for precise numbers here is the same problem as asking for a general probability parameter, and asking for a value for "impact" is even more difficult, since this can be any number (such as for financial losses), but also just an indicator (such as loss of human lives, where a quantification of "damage" would induce substantial ethical issues beforehand).
4. Risk Evaluation: with risks identified, the evaluation step asks for a ranking of those to assign priorities to risks whose mitigation is more urgent than for others. Here, we actually do not need to evaluate the risk formula from above, as all we require is a ranking of risks, based on impact and likelihood. This degree of freedom is important to stress, since we can create the familiar risk bubble charts like shown in Figure 4 without numeric information up to this point, even though the risk management process itself may be quantitative in the end. All we need is ranks, rather than precise numbers!



Impact ranking: $R_1 < R_2 < R_3 < R_4 < R_5$
Likelihood ranking: $R_1 < R_3 < R_5 < R_4 < R_2$

Figure 4 *Risk Bubble Chart with Induced Rankings (Example)*

5. Risk treatment: this is the point where we actually need to become "somewhat numeric",

since decisions about mitigation actions will in most cases depend on the expected efficacy, or equivalently said, the *return on investment* for a security control. Based on the risk after mitigation, obtained from the formula above, we can step forward by taking decisions for those actions that optimize the impacts or likelihoods (or both), so as to optimize the risk. This is what risk treatment is about, and what the parallel task of "monitoring and review" prescribes as part of ISO31000, as well as a continuous communication of all these steps to the outside (see Figure 5).

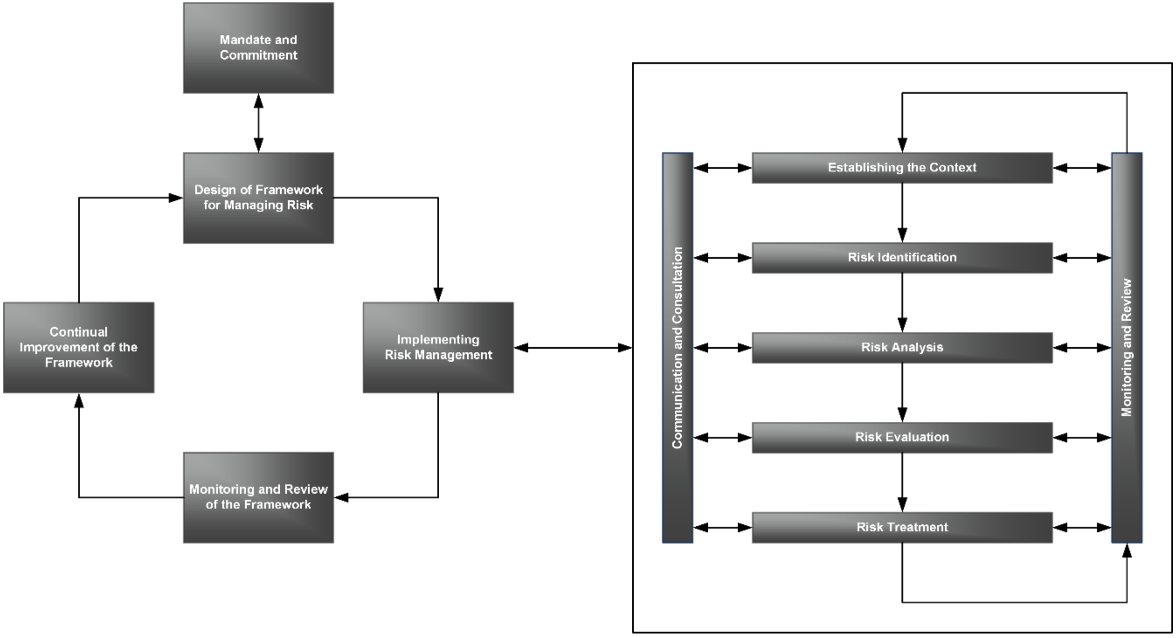

Mandate and Commitment
Design of Framework for Managing Risk
Continual Improvement of the Framework
Implementing Risk Management
Monitoring and Review of the Framework
Communication and Consultation
Establishing the Context
Risk Identification
Risk Analysis
Risk Evaluation
Risk Treatment
Monitoring and Review

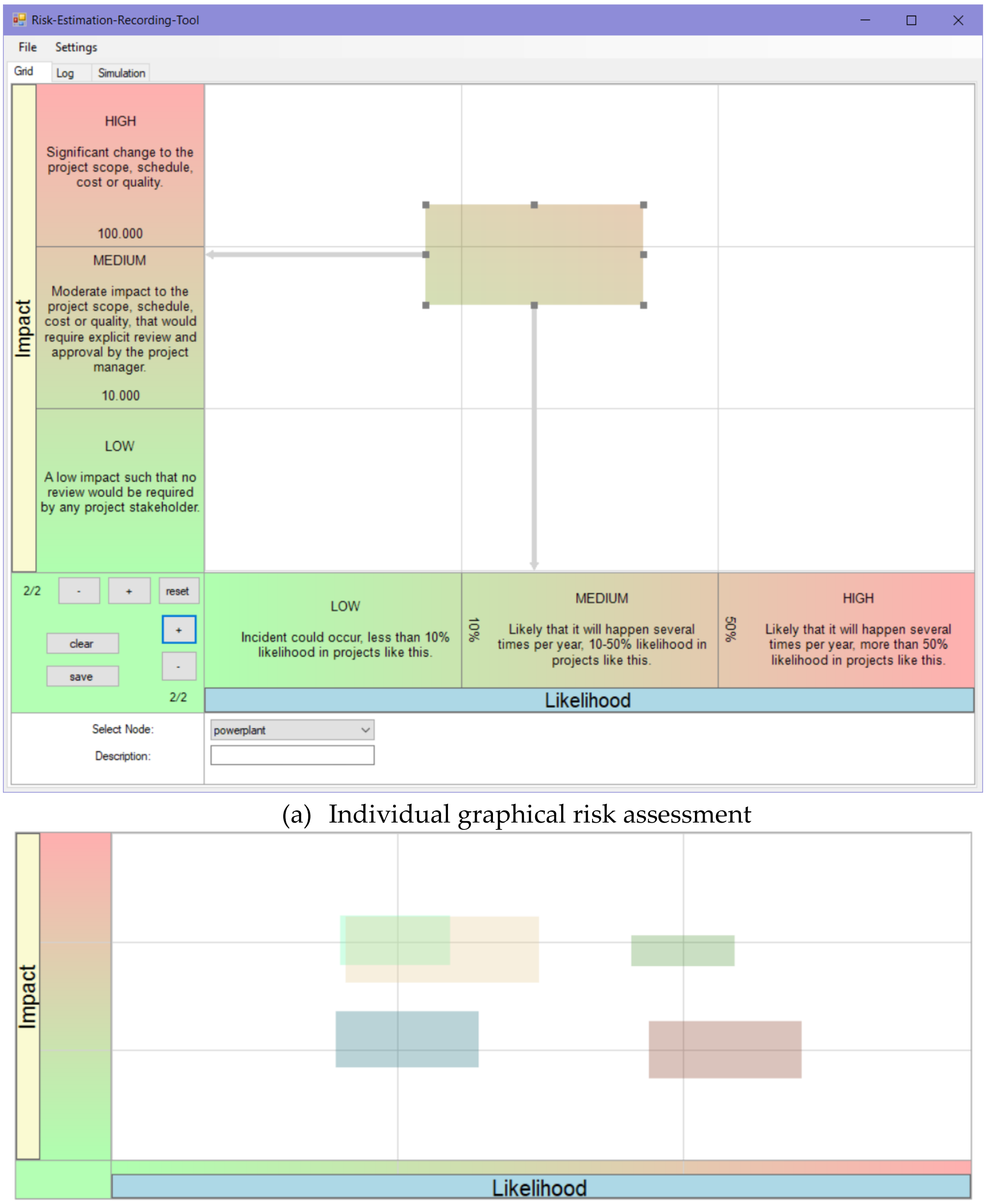
Figure 5 *ISO 31000 Risk Management Process [23]*

Following Peter Drucker's famous quote that "you can't manage what you can't measure", evaluating the risk formula eventually becomes necessary. Or doesn't it? The perhaps surprising (though well known) answer is no!

Suppose that we ask a domain expert for an assessment of several risks at the same time, specifically, allowing for doing the ranking of impacts and likelihood relative to one another (and in an order of risks that is up to the expert's own choice). Figure 6(b) shows how the results of such an assessment may look like, with four five boxes being drawn on the grid, at positions that were dependent on one another.

This representation resembles that of a usual risk matrix, only offering new and interesting possibilities: first, we can visually inspect the picture for outliers, and remove them (manually) if necessary. Second, and more importantly, we could aggregate those values into a single representative value, which in the simplest case amounts to taking an average, or in a more sophisticated form, takes the variations, i.e. uncertainties (reflected in the height and width of the boxes) into account to weigh each value inverse proportionally to the "certainty" in the final average (see [18,24] for several proposals in this direction).

Our goal, however, is not on getting numbers from the image, but rather on decision making, for which numbers are an aid, but not a necessity. Game theory offers the answer on how to make decisions based on rankings between actions (only), if we recall the very fundamental starting points laid by von Neumann and Morgenstern themselves [25] (and later extended by Debreu [26]): The important insight of these pioneers was that certain ordering relations can be expressed by real-valued functions, which we commonly call *utility functions* in game theory.



HIGH
Significant change to the project scope, schedule, cost or quality.
100.000

MEDIUM
Moderate impact to the project scope, schedule, cost or quality, that would require explicit review and approval by the project manager.
10.000

LOW
A low impact such that no review would be required by any project stakeholder.

LOW
Incident could occur, less than 10% likelihood in projects like this.

MEDIUM
Likely that it will happen several times per year, 10-50% likelihood in projects like this.

HIGH
Likely that it will happen several times per year, more than 50% likelihood in projects like this.

Select Node: powerplant
Description:

(a) Individual graphical risk assessment

(b) Assessment of several risks relative to one another

Figure 6 *Subjective relative risk ratings*

The application of game theory to matters of decision making in risk management is simple, but instructive:

- Suppose that we have a status quo in a system, and several threats in question of which one is most urgent to address. This (simple) decision problem only asks for ranks, not numbers, and a graphical specification is all we need.
- Likewise, if there is a single threat to be addressed now by several possible countermeasures, their efficacy is equally well specifiable on a ranking scale, and does not require numbers per se. If we seek for a balance between investment and efficacy, a two-dimensional ranking such as in Figure 2, Figure 4 or Figure 6 is already sufficient.
- The combination of several threats and several countermeasures to address them is the nontrivial case, where randomized decisions are often unavoidable. Let us consider the

simplest example from game theory, Rock-Scissors-Paper, which despite its triviality, is nonetheless a valid "template" for a risk management decision making process (just think of the column and row labels to be replaced with threats and countermeasures).

Payoffs: (player 1, player 2)

		Player 2		
		Rock	Scissors	paper
Player 1	Rock	(0, 0)	(1, -1)	(-1, 1)
	Scissors	(-1, 1)	(0, 0)	(1, -1)
	Paper	(1, -1)	(-1, 1)	(0, 0)

This game is straightforward to specify in the sense that we just "assign" a payoff of +1 or -1 to a player depending on whether it wins or loses. For security risk management, and generally many security models, including probabilistic ones in particular, the assignment of risks, based on impacts and likelihoods, also requires numbers, but which are much more difficult to obtain or argue. The numbers +1 and -1 in the above game matrix are just a direct specification of a utility function, and many (if not most) game theoretic models do give a direct such specification.

The axiomatic roots of game theory, however, start with a proof of existence of such functions, which merely demands a ranking of actions, and from this starting point, constructs utility values to represent this ordering. The important point is that these utility functions are constructed explicitly by the theory, so we can repeat the steps of the proof to get utility values. Irrespectively of whether we are seeking values for impacts, likelihoods or other (probabilistic) variables, all we need is a specification of values relative to each other. Formally, let us thus generically consider a space $(R, \leq)$, where R can be a set of impact, likelihood, or other values that we ought to specify. The simple ingredients enabling us to assign a value only concern the ordering on R; more generally, on its convex hull, with the ordering naturally extended; following the exposition of [27]: the ordering relation should be *total* and *transitive*. Furthermore, we require *conservation of the order under indifferent alternatives*, meaning that

$$r_1 \leq r_2 \text{ and } \alpha r_1 + (1-\alpha) r_2 \leq \alpha r_2 + (1-\alpha) r \text{ should hold for all } r \in R,$$

and *connectedness* (or closedness), meaning that for every $r_1 \leq r_2 \leq r_3$, there are two values $\alpha, \beta \in (0,1)$ such that $\alpha r_1 + (1-\alpha) r_3 \leq r_2 \leq \beta r_1 + (1-\beta) r_3$. The last assumption implies that for any two $r_1 < r_2$ that enclose some r as $r_1 \leq r \leq r_2$, there is a unique value $v \in [0,1]$ with $r \sim v \cdot r_1 + (1-v) \cdot r_2$, where the $\sim$-relation is induced by $\leq$ in the canonic way ($r \sim s \Leftrightarrow [r \leq s] \wedge [s \leq r]$). If the ordering satisfies these axioms, we can define a utility value as

(1) $U(r) = v$ if $r_2 \leq r \leq r_1$ and $r \sim v r_1 + (1-v) r_2$
(2) $U(r) = -\frac{v}{1-v}$ if $r \leq r_2$ and $r_2 \sim v r_1 + (1-v) r$
(3) $U(r) = \frac{1}{v}$ if $r_1 \leq r$ and $r_1 \sim v r + (1-v) r_2$

In particular, $U(r_1) = 1$ and $U(r_2) = 0$, and U preserves the ordering $\leq$ on R. Indeed, U is a linear function, since if $r_2 = \alpha r_1 + (1-\alpha) r_3$, we have $U(r_2) = \alpha U(r_1) + (1-\alpha) U(r_3)$. Extended to the convex hull of R, we need one more assumption to admit *linearization*: fix the range on which we need to specify our parameters as the interval $[r_1, r_2]$ and let P be a probability measure on this range with $P([r_1, r_2]) = 1$. With $\alpha(r) := \big(U(r) - U(r_1)\big) / \big(U(r_2) - U(r_1)\big)$ and $\beta = \int_{r_1}^{r_2} \alpha(r) dP(r)$, we need to assume that $P \sim \beta \delta_{r_2} + (1-\beta) \delta_{r_1}$ (where δ is the Dirac mass), i.e., if some value $r \sim \alpha(r) r_1 + \big(1 - \alpha(r)\big) r_2$, then this equivalence holds on average. This is nothing else than the assumption of "linearity" on the scale between two ratings (ranks) r_1 and r_2, and we have $\beta =$

$\frac{E_P[U(r)]-U(r_1)}{U(r_2)-U(r_1)}$ Mapping this back to our graphical specification, all this just formalizes that the axes defining the 2D area on which the rectangles are drawn are *linearly scaled*. Under (all these) assumptions, we can use the so-constructed function U to express the ordering (ranking) for us, since for any two $P_1, P_2 \in conv(R)$ (where $conv$ is the convex hull, or equivalently, set of randomized decisions), we have $P_1 \leq P_2$ if and only if $E_{P_1}[U(r)] \leq E_{P_2}[U(r)]$. Moreover, this function U is unique up to affine transformations, i.e., any alternative valuation U' would take the form $U'(r) = a \cdot U(r) + b$ for some real values $a > 0$ and b.

How do we make use of all these (old and well known) facts for our actual challenge of specifying numbers in absence of precise knowledge about them? The idea is to use precisely the "three-case" definition of U above based on the ranking, not values, of the actions by making use of "linear interpolation" between them. More concretely, assuming that the axes are linearly scaled, we can just go ahead and take the graphical (visual) coordinates of the graphical range, mapped to the utility values U and letting us compute optimal decisions by standard methods and algorithms from game theory. The graphical specification is herein an aid to get the utility function for the decision making, and backed up by the axiomatic foundation of game theory. The crucial point, however, is that all of this, namely

1. The graphical ranking of actions, risks, etc.
2. The retrieval of (graphical) coordinates of those to play the role of the utility function
3. And the decision making as a matter of optimization (over finite sets in our example case even),

works without ever asking an expert for any number!

It is not surprising that this theoretical possibility has a number of caveats. First of all, our thoughts cannot be taken as formal argument or are mathematically rigorous here; the axiomatic approach to the existence of utilities hereby only plays the role of making our heuristic plausible, but do not lend themselves to proving any correctness. Numeric values obtained in this way do not necessarily have any particularly better semantic or accuracy than any other educated guess, but the main point of all this is to *ease guessing*, but without claiming to improve it.

Essentially, all of the axioms to define a utility function as such can be put to question, and the whole field of bounded rationality [28,29] deals with observations on human decision making to violate one or more of these assumptions (an excellent essay about this is that of Starmer [30]). Propsect theory [31] for example, accounts for phenomena of over- and underrating values near the end of the scale. For probabilities, this amounts to the effect that, subjectively, low probabilities are overrated, while large probabilities are underrated potentially. Probability weighting functions like that of Prelec [32] try to annihilate this effect. The research prototype shown in Figure 6(a) allows for a similar such correction by letting subjects individually adjust the grid towards smaller or wider ranges of the scales (visually).

A theoretical limitation concerns the use of multiple goals, as often occur in risk management applications. The existence of a utility function like the above is known under a variety of alternative conditions, often summarized as *Debreu representation theorems*. The common denominator therein is the *continuity* of the ordering, meaning that whenever a sequence $(r_n)_{n\in\mathbb{N}}$ with limit $r = \lim_{n\to\infty} r_n$ satisfies $s \leq r_n$, then the limit r should also satisfy $s \leq r$. This holds for orders on real values, and is indeed a major reason for game theory to be done mostly within $\mathbb{R}$. In higher dimensions, we can resort to weighted sums of utilities for different goals, which leads to Pareto-optimal decisions. However, if the ordering among the goals is "more explicit" in the sense of being lexicographic, then continuous utility functions no longer exist. The lexicographic order is indeed not continuous in the sense just stated. To see this, consider any sequence $a_n \to 0$, and take the limit of $(0, a_n)$ as $n \to \infty$. Then, obviously, $(a_n, 0) >_{lex} (0,1)$, but $\lim_{n\to\infty}(a_n, 0) = (0,0) \leq_{lex} (0,1)$, so the ordering is discontinuous. This leads to variations of decision theory, based on non-standard calculus and extension fields to $\mathbb{R}$, such as has been done in [19,33], in which some discontinuities naturally "disappear" by virtue of the richer algebraic structures. Those methods are applicable when multiple goals are relevant for a simultaneous optimization, or if the optimization shall be w.r.t. a lexicographic order (see Figure 4, where the lexicographic order would first consider "impact" and break ties using the "likelihood"). See [34] for an implementation of such methods in the **R** software [35].

VI. Conclusion

The ideas laid out here are applicable whenever a probabilistic parameter describes an observable event, so that data for a Bayesian update is collectible. A practical issue can indeed be the speed of convergence, since the above argument is nonetheless asymptotic, and the true value is reached only after a hypothetic infinitude of updates. Therefore, we may need to update upon every incoming ticket at the IT administration office, or as often as we can, in practice.

We also stress that the above model does not serve too well as a model of human trust: the updating is in some sense "symmetric" and "self-stabilizing", meaning that (i) the likelihood changes eventually become smaller as more updates come in (self-stabilization), and the likelihoods will update with roughly comparable magnitudes in both directions. The latter is contrary to human subjective changes to trust, since confidence in an event to occur may substantially change upon recent experience and differently in the direction towards zero or towards one. In other words, if the probabilistic parameter is interpreted as a "trust value", say, if we take it as the expectation of some event (that we rely on) to occur, then subjective trust may be lost upon a single incident, but may be regained only over a much longer period of positive experience. On the contrary, the above model would not reflect such asymmetry due to human pessimism. This leads to the advice of applying the above model only for the estimation of parameters that describe *physical* processes, and *not* subjective human factors. The latter are subject to much deeper psychological mechanisms for whose capture the above model may be overly simplistic.

If the parameter in question, however, relates to a physical event that can be observed, then the Bayesian updating as described above offers a computationally efficient and elegant way of online learning parameters in absence of reliable domain expertise to specify a (more) accurate model or prior guess.

Finally, the methods outlined here are so far conceptual and lack an empirical study on accuracy, subjective comfort felt in the specification methods as such and similar. While they are certainly viable to make a start for a Bayesian updating, open questions relate to the accuracy of any such "guess", which on the one hand determines the speed of convergence as further Bayes updates come in, and on the other hand, have a direct influence on the accuracy for decision making in risk management.

References

[1] Schauer, S, König, S, Latzenhofer, M, Rass, S and Grafenauer, T (2018). Analyzing Cascading Effects among Critical Infrastructures : The CERBERUS Approach

[2] Rahnamay-Naeini, M and Hayat, M M (2016). Cascading Failures in Interdependent Infrastructures: An Interdependent Markov-Chain Approach. *IEEE Transactions on Smart Grid*. **7** 1997–2006

[3] Rahnamay-Naeini, M, Wang, Z, Ghani, N, Mammoli, A and Hayat, M M (2014). Stochastic Analysis of Cascading-Failure Dynamics in Power Grids. *IEEE Transactions on Power Systems*. **29** 1767–79

[4] Koenig, S, Schauer, S and Rass, S (2016). A Stochastic Framework for Prediction of Malware Spreading in Heterogeneous Networks. *NordSec 2016*. Springer. 67–81

[5] König, S and Rass, S (2018). Investigating Stochastic Dependencies Between Critical Infrastructures. **11** 250–8

[6] Dudenhoeffer, D D and Permann, M R und M M (2006). CIMS: A Framework For Infrastructure Interdependency Modeling And Analysis. http://dx.doi.org/10.1109/wsc.2006.323119

[7] Buldyrev, S V, Parshani, R, Paul, G, Stanley, H E and Havlin, S (2010). Catastrophic cascade of failures in interdependent networks. *Nature*. **464** 1025

[8] Wang, Z, Scaglione, A and Thomas, R J (2012). A Markov-Transition Model for Cascading Failures in Power Grids. *2012 45th Hawaii International Conference on System Sciences*. IEEE, Maui, HI, USA. 2115–24. http://ieeexplore.ieee.org/document/6149269/

[9] Dobson, I, Carreras, B A and Newman, D E (2004). A branching process approximation to cascading load-dependent system failure. *37th Annual Hawaii International Conference on System Sciences, 2004. Proceedings of the*. IEEE, Big Island, HI, USA. 10 pp. http://ieeexplore.ieee.org/document/1265185/

[10] Dobson, I (2012). Estimating the Propagation and Extent of Cascading Line Outages From Utility Data With a Branching Process. *IEEE Transactions on Power Systems*. **27** 2146–55

[11] Qi, J, Sun, K and Mei, S (2015). An Interaction Model for Simulation and Mitigation of Cascading Failures. *IEEE Transactions on Power Systems*. **30** 804–19

[12] Kelic, A, Warren, D E and Phillips, L R (2008). *Cyber and Physical Infrastructure Interdependencies.* http://www.osti.gov/servlets/purl/945905-sJflYi/

[13] Rinaldi, S, Peerenboom, J and Kelly, T (2001). Identifying, Understanding, and Analyzing Critical Infrastructure Interdependencies. 11–25

[14] Dudenhoeffer, D D and Permann, M und B R L R (2006). Decision consequence in complex environments: Visualizing decision impact

[15] Pederson, P, Dudenhoeffer, D D, Hartley, S and Permann, M R (2006). Critical Infrastructure Interdependency Modeling: A Survey of U.S. and International Research

[16] Laugé, A, Hernantes, J and Sarriegi, J M (2015). Critical infrastructure dependencies: A holistic, dynamic and quantitative approach. *International Journal of Critical Infrastructure Protection*. **8** 16–23

[17] Schaberreiter, T, Bouvry, P, Röning, J and Khadraoui, D (2013). A Bayesian Network Based Critical Infrastructure Risk Model. *EVOLVE - A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation II*. Springer Berlin Heidelberg, Berlin, Heidelberg. **175** 207–18. http://link.springer.com/10.1007/978-3-642-31519-0_13

[18] Carvalho, A and Larson, K (2013). A Consensual Linear Opinion Pool. *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*. AAAI Press. 2518–2524. http://dl.acm.org/citation.cfm?id=2540128.2540491

[19] Rass, S, Konig, S and Schauer, S (2016). Decisions with Uncertain Consequences-A Total Ordering on Loss-Distributions. *PLoS ONE*. **11** e0168583

[20] Rass, S, Wachter, J, Schauer, S and König, S (2017). Subjektive Risikobewertung – Über Datenerhebung und Opinion Pooling. *D-A-CH Security 2017*. syssec. 225–237

[21] Rass, S and Kurowski, S (2013). On Bayesian Trust and Risk Forecasting for Compound Systems. *Proceedings of the 7th International Conference on IT Security Incident Management & IT Forensics (IMF)*. IEEE Computer Society. 69–82

[22] Kubiak, P and Rass, S (2018). An Overview of Data-Driven Techniques for IT-Service-Management. *IEEE Access*. **6** 63664–63688

[23] Rass, S and Schauer, S (2018). *Game Theory for Security and Risk Management: From Theory to Practice*. Springer

[24] Dietrich, F and List, C (2017). Probabilistic opinion pooling generalized. Part one. *Soc Choice Welf*. **48** 747–86

[25] von Neumann, J and Morgenstern, O (1944). *Theory of Games and Economic Behavior*. Princeton University Press

[26] Debreu, G (1987). *Theory of Value: An Axiomatic Analysis of Economic Equilibrium*, 19. Dr. Yale Univ. Press, New Haven

[27] Robert, C P (2001). *The Bayesian Choice*. Springer, New York

[28] (2002). *Bounded Rationality*, 1st MIT Press pbk. ed. MIT Press, Cambridge, Mass. http://gso.gbv.de/DB=2.1/PPNSET?PPN=801088364

[29] Rubinstein, A (2002). *Modeling Bounded Rationality*, 3. print. MIT Press, Cambridge, Mass. http://gso.gbv.de/DB=2.1/PPNSET?PPN=358990017

[30] Starmer, C (2000). Developments in Non-Expected Utility Theory: The Hunt for a Descriptive Theory of Choice under Risk. *Journal of Economic Literature*. **38** 332–382

[31] Tversky, A and Kahneman, D (1992). Advances in prospect theory: Cumulative representation of uncertainty. *Journal of Risk and Uncertainty*. **5** 297–323

[32] Prelec, D (1998). The Probability Weighting Function. *Econometrica*. **66** 497

[33] Rass, S, Konig, S and Schauer, S (2017). Defending Against Advanced Persistent Threats Using Game-Theory. *PLoS ONE*. **12** e0168675

[34] Rass, S and König, S (2018). *R Package 'HyRiM': Multicriteria Risk Management Using Zero-Sum Games with Vector-Valued Payoffs That Are Probability Distributions*. Austrian Institute of Technology (AIT). https://cran.r-project.org/web/packages/HyRiM/index.html

[35] R Core Team (2018). R: A Language and Environment for Statistical Computing. www.r-project.org