Proceedings of the 2021 Winter Simulation Conference
S. Kim, B. Feng, K. Smith, S. Masoud, Z. Zheng, C. Szabo, and M. Loper, eds.

EXPLOITING PROVENANCE AND ONTOLOGIES IN SUPPORTING BEST PRACTICES FOR SIMULATION EXPERIMENTS: A CASE STUDY ON SENSITIVITY ANALYSIS

Pia Wilsdorf
Nadine Fischer
Fiete Haack
Adelinde M. Uhrmacher

Institute for Visual and Analytic Computing
University of Rostock
Albert-Einstein-Str. 22
Rostock, 18059, GERMANY

ABSTRACT

Simulation studies are intricate processes and user support for conducting more consistent, systematic, and efficient simulation studies is needed. Simulation experiments as one crucial part of a simulation study can benefit from semi-automatic method selection, parameterization, and execution. However, this largely depends on the context in which the experiment is conducted. Context information about a simulation study can be provided in form of provenance that documents which artifacts contributed in developing a simulation model. We present an approach that exploits provenance to support best practices for simulation experiments. The approach relies on 1) explicitly specified provenance information, 2) an ontology of methods, 3) best practices rules, and 4) integration with a previously developed experiment generation pipeline. We demonstrate our approach by conducting a sensitivity analysis experiment within a cell biological simulation study.

1 INTRODUCTION

Modeling and simulation has been established as a key tool in many sciences. As simulation studies are inherently complex, there is naturally a need for guidance, which has been addressed by life cycles (Balci 2012; Sargent 2013) and workflows (Ruscheinski et al. 2019). However, there is still a growing demand for support in more specific tasks of a simulation study, for example, in choosing the right agent-based model type for a given problem (Gray et al. 2017), in composing various simulation models (Neal et al. 2018), or in specifying simulation experiments in a compact way (Ewald and Uhrmacher 2014).

With regards to simulation experiments, a frequently asked question is which method to use (ten Broeke et al. 2016). The automatic selection of methods in modeling and simulation has been investigated with the goal of achieving more robust results or faster runtimes. E.g., in the case of steady state estimation, algorithm portfolios have been used (Leye et al. 2014). The idea of algorithm portfolios was to apply a set of algorithms to the same problem and to combine their results. How these results are combined is learned in relating problem features and algorithm features via decision trees. In the case of selecting an efficient simulation algorithm, different approaches exist. For instance, reinforcement learning could be successfully applied, with wallclock time as reward (Helms et al. 2015).

However, for other experiment types such as optimization, finding the best performing method proved difficult (Villaverde et al. 2018), due to the fact that properties of the response surface are typically "not known a priori". Another problem is that, with regards to the applicability of a method, there may not be one best method but rather various alternatives. Modelers may not be aware of them or their differences. Also, methods are often applied without considering the context. A recent review on sensitivity analysis

978-1-6654-3311-2/21/$31.00 ©2021 IEEE

(SA) found that approximately half of the conducted analyses were applied falsely (Saltelli et al. 2019). This makes it difficult to learn from previous cases.

Therefore, support for SA (as well as other critical experiment types) should rely on thoroughly collected knowledge based on theory instead of imitating previous applications and carefully consider the context of the simulation study. Providing a knowledge base that can guide the decision-making process by eliminating certain methods based on provided criteria, could help to bundle expertise and automate certain steps, which eventually prevents fundamental methodological mistakes and gives some guidance, especially for inexperienced users.

We present an approach for supporting best practices in simulation experiments that brings the following components together: context about the simulation study given explicitly by provenance, an ontology defining the general properties of the various analysis methods, rules describing the best practices (i.e., relating context knowledge with knowledge about methods), and a pipeline for automatic experiment generation and execution. As running example we will use the best practices for SA formulated by (Saltelli et al. 2019) to illustrate the steps of our approach. Thereafter, we demonstrate the suitability of our approach for a cell biological simulation study, for which provenance has been identified in (Budde et al. 2021).

2 PROVENANCE

The provenance of a simulation model describes all activities and artifacts that contributed to its generation (Ruscheinski and Uhrmacher 2017). The artifacts for a simulation study comprise first of all the simulation models `SM` and the simulation experiments `SE`, as well as the simulation data `SD`, i.e., the results of a simulation experiment. The typical activities in a simulation study include building or refining the simulation model `BSM`, calibrating simulation model `CSM`, validating simulation model `VSM`, and analyzing simulation model `ASM` (Budde et al. 2021). In addition to the three main artifacts, the activities can take into account further artifacts that concern the context of the simulation study. These artifacts include the research questions `RQ`, requirements `R`, assumptions `A`, qualitative model `QM`, and various data for example wet lab data `WD` used as input to the simulation model. These contextual artifacts together are also referred to as the conceptual model (Wilsdorf et al. 2020).

Figure 1 shows a small provenance example depicted as a directed acyclic graph using the PROV-DM notation (Belhajjame et al. 2013). Here, a simulation model was built based on a qualitative model and an assumption. Subsequently, a validation activity took the simulation model and data and produced a simulation experiment as well as simulation data.

Provenance is crucial for effectively supporting simulation studies (Wilsdorf et al. 2020). It provides valuable information for validating and calibrating the model as well as interpreting the results or selecting and parametrizing appropriate analysis methods.

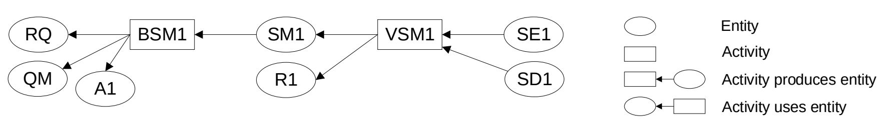

Figure 1: Example of a provenance with a building simulation model activity (`BSM`) and a validating simulation model activity (`VSM`). Activities are denoted by rectangles, and artifacts are denoted by ellipses.

3 SUPPORTING BEST PRACTICES

We present an approach for automatically supporting best practices for simulation experiments. The architecture is depicted in Figure 2. The approach consists of 1) an explicit representation of provenance, 2) an ontology attributing properties to methods of a specific experiment type, 3) best practices for an experiment type (i.e., rules relating provenance with general methodological knowledge of the ontology), and 4) automatic experiment generation based on experiment schemas.

We assume that we know which experiment type (e.g., SA, simulation-based optimization, or statistical model checking) the user wants to carry out next. This information can be obtained, e.g., by combining our approach with a declarative artifact-based workflow for simulation studies. When the user creates a new experimentation activity with an empty experiment (e.g., of type SA) our automatic support is initiated. In the first step of the approach, we check whether the selected experiment type is appropriate given the current state of the simulation study, and possibly other simulation experiments are suggested. In the second step, a set of suitable methods for the experiment type is selected. These are then automatically parametrized in the third step using an experiment schema. Schemas have been defined for the different experiment types to structure their ingredients, see (Wilsdorf et al. 2019). Next, the user can manually narrow down the selection and adapt the parametrization of the methods. Finally, executable code is generated and run for a target backend. This fills the new experiment and data entities and creates the necessary relationships in the provenance graph.

In this section, we will describe the steps and their ingredients further. As a running example (as well as later in the case study) we will use the best practices for SA presented in (Saltelli et al. 2019). In their paper, Saltelli et al. reviewed a number of studies that applied SA in various fields of science and engineering. They found that many SAs are false, and therefore formulated six best practices to prevent the most frequent mistakes. For each best practice, we will discuss how the knowledge provided by provenance and ontologies can be exploited.

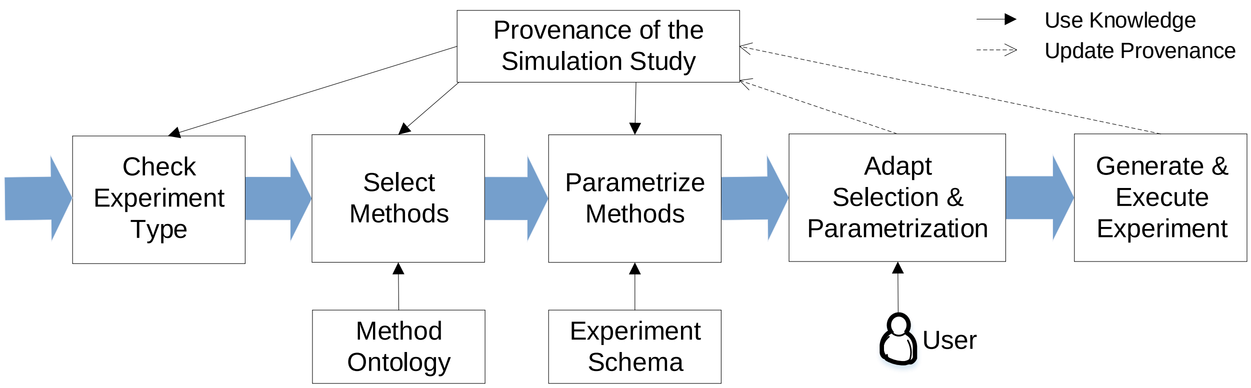

Figure 2: Approach for supporting best practices for automatically selecting, parametrizing, generating and executing simulation experiments.

3.1 Checking the Choice of Experiment Type

The first step is deciding whether it makes sense to conduct an experiment of the given type at the current point in the simulation study. Based on context knowledge, other experiment types may be suggested. The experiment type chosen by the user may be rejected, e.g., if already similar experiments have been done and the model has not significantly changed since. Moreover, some other experiment types may be considered a prerequisite for running the chosen analysis. For SA, Saltelli provides the following best practices:

Best practice #1 (combination of UQ and SA): *"With some exceptions, it is advisable to perform both uncertainty and sensitivity analysis. Once an analyst has performed an uncertainty analysis and is informed of the robustness of the inference, it would appear natural to ascertain where volatility/uncertainty is coming from. At the other extreme, a sensitivity analysis without uncertainty analysis is usually illogical [...]. However, there are cases – for instance, studies to identify the dominant effects on the output for a subsequent model reduction or calibration analysis – where the analyst may be satisfied with a pure SA."*
Exploiting the knowledge: The relationship between uncertainty quantification (UQ) and SA is described. UQ allows assessing of the uncertainty associated with the model response as a result of uncertainties in the

model input, whereas SA studies how the variation in the model output can be apportioned to the different sources of variation. To decide whether a UQ needs to precede the SA, we need to extract information about the role of the current activity since SA can serve different purposes such as analysis, calibration, or validation. If the current activity is an analyzing simulation model activity (`ASM`), a UQ experiment is seen as a precondition. Therefore, if no simulation experiment has been previously conducted with the model (i.e., no `SE` with type UQ exists in the provenance graph), a UQ experiment needs to be conducted first. Otherwise, if the role of the current activity is `CSM` or `VSM`, then the SA experiment can be conducted without a UQ.

Best practice #2 (validate and verify the model): *"Even an apparently perfect uncertainty and sensitivity analysis is no assurance against error. As noted by (Pilkey and Pilkey-Jarvis, 2009) 'It is important to recognize that the sensitivity of the parameter in the equation is what is being determined, not the sensitivity of the parameter in nature. [...] If the model is wrong or if it is a poor representation of reality, determining the sensitivity of an individual parameter in the model is a meaningless pursuit.'"*
Exploiting the knowledge: This best practice states that a SA is only as good as the model itself. Therefore, SA should never be the only experiment conducted with a model. Instead, it should be accompanied by experiments that can ascertain validity. A model can be considered validated if all behavioral requirements (`R`) and all wet lab data (`WD`) with role validation have been checked successfully. I.e., for each of the relevant requirement and data artifacts, there needs to exist a validation activity `VSM` that used the artifact and produced a corresponding simulation experiment (`SE`) with simulation data (`SD`) and their status is "validation successful". Otherwise, if unchecked requirements or unused validation data are found, further validation experiments are required. Therefore, the system suggests running also these validation experiments. E.g., statistical model checking is a common experiment type for testing the formally specified behavioral requirements of a stochastic model. However, these experiments are only recommendations and they do not have to precede the SA.

3.2 Selecting the Suitable Methods

In the previous step, the applicability of the experiment type was checked and other crucial experiment types might have been suggested. Now, for the current experiment type, a set of suitable methods needs to be selected. Therefore, in addition to the context knowledge, we require knowledge that characterizes the various possible methods. We use ontologies to assign properties to methods, e.g., which types of models they can or cannot handle, how computationally expensive they are, or which kinds of measures can be calculated (e.g., first-order, second-order sensitivity indices). Figure 3 shows an excerpt of an ontology we developed specifically for SA methods. It is written in the Manchester OWL syntax (Horridge et al. 2006) and describes the properties of the Elementary Effects method, also known as the Morris method. Given an ontology, a list of suitable methods can be extracted by building the right description logic (DL) query. We use the following two best practices to give examples of such queries.

Best practice #3 (global vs local methods): *"Both uncertainty and sensitivity analysis should be based on a global exploration of the space of input factors, be it using an experimental design, Monte Carlo or other ad-hoc designs. [...] local/OAT methods do not adequately represent models with nonlinearities."*
Exploiting the knowledge: This best practice promotes the use of global SA methods over local SA methods, especially for nonlinear models. Therefore, first the provenance graph needs to be searched for an assumption artifact (`A`) that refers to the response of the model. Based on the assumption found, a DL query is built to extract the suitable methods. In the case where the model response is assumed nonlinear or no such information is found, only global SA methods are selected from the ontology: `SAMethod and GlobalMethod`. If the model is assumed to exhibit linear behavior, also local one-at-a-time (OAT) methods are selected: `SAMethod and (GlobalMethod or LocalMethod)`.

Best practice #4 (use versatile methods): *"When sensitivity analysis is performed, it should allow the relative importance of input factors and combinations of factors, to be assessed, either visually (scatterplots)*

or quantitatively (regression coefficients, sensitivity measures or other).”
Exploiting the knowledge: In contrast to the previous best practices, this one does not depend on the context information provided by provenance. Which methods allow calculating which measures and therefore allow which kinds of conclusions is contained in the ontology. Again, a DL query is developed. This time it filters all methods that can be used for factor ranking and can calculate the interaction effects between parameters: `has Purpose some Ranking and hasSpecialty Interactions`.

```
Class: Morris
SubClassOf:
SAMethod
GlobalMethod
hasComplexity some LowComplexity
hasPurpose some Ranking
failsWith some NonSmoothModelResponse
...
DisjointWith:
FAST, Sobol, SRCs, ...
```

Figure 3: Specification of the class “Morris” of the SA method ontology.

3.3 Parametrizing the Methods

To provide as much support as possible, we also aim to automatically parametrize the selected methods. In previous work, we have introduced experiment schemas (Wilsdorf et al. 2019) as a means to define the required and optional inputs of various types of simulation experiments and methods, including SA and statistical model checking. We try to at least partially fill these schemas with provenance information. The following best practices emphasize the importance of carefully choosing the interesting input and output variables for the analysis.

Best practice #5 (choice of output variable): *“Sensitivity and uncertainty analysis should be focused on a question. Most models have many outputs, and these outputs can be used to answer a range of different questions. The relationship (sensitivity) between the input factors and each different model output can be very different. For this reason, it is essential to focus the sensitivity analysis on the question addressed by the model rather than more generally on the model.”*
Exploiting the knowledge: Provenance can be used to find a suitable target for the SA. The qualitative model (`QM`) is an artifact that makes explicit the model components as well as the model inputs and outputs. We therefore will use the qualitative model as the primary source for setting the analysis target. If the `QM` contains more than one output variable, an SA should be created for each of them.

Best practice #6 (choice of factors): *“[...] modellers should be frank about how they arrived at the supposed uncertainties (Saltelli et al., 2013). This should be kept in mind and efforts made to capture the uncertainty of input assumptions as accurately as possible.”*
Exploiting the knowledge: This best practice emphasizes the importance of making the uncertainties of a model explicit. I.e., when factors are associated with some uncertainty, they become an assumption. Therefore, assumption artifacts can be exploited in selecting the relevant input factors for the SA. Especially for large models, it can be useful to center the analysis around the most uncertain factors. Concatenated with best practice #1 (combination of UQ and SA) this yields a kind of “workflow”: First UQ (e.g., via approximate Bayesian computation) is applied for estimating the posterior distributions of the parameters, and subsequently, the distributions are used as input for a global SA (Eriksson et al. 2018). In the case where no assumptions on parameters have been specified, either all model parameters could be used (typically provided as a table inside the qualitative model), or further best practice rules could be applied (whose definition is beyond the scope of this paper).

3.4 Adapting, Generating and Executing the Experiments

If the selected SA methods could be parametrized completely, the following steps (code generation and execution) can be carried out fully automatically. This is achieved by passing the method and its parameters into an existing pipeline for experiment generation which was introduced by (Ruscheinski et al. 2018) and extended by (Wilsdorf et al. 2019).

However, further input from the user may be required, or some of the extracted information needs to be adapted (semi-automatic step). This may be due to insufficient information if the modeler has not regularly updated the conceptual model. On the other hand, some decisions are highly domain- or even model-specific, which is currently not covered by our exemplary best practice rules. To later benefit from the added or adapted information, all changes such as modified parameter ranges are stored again in the respective artifacts. Finally, if multiple SA methods or output variables were selected, the user might want to further reduce the selection, e.g., due to runtime constraints.

4 IMPLEMENTATION

We developed our SA ontology using Protegée (Musen 2015). We based it on the categorizations obtained from two review papers (Pianosi et al. 2016; Borgonovo and Plischke 2016). For the concrete methods, we pre-selected a few methods that we found readily available in popular SA packages (Herman and Usher 2017; Iooss et al. 2019). Our ontology therefore currently focuses on the following methods: a simple two-level full factorial analysis as a representative of local SA, and the following global methods: the Elementary Effects method by Morris, Delta, Sobol, RBD-FAST, FAST, SRC, and PRCC. Note that this list is not complete and that the ontology could be extended with many more methods. The ontology (OWL file) is imported and queried using the OWLApi (Horridge and Bechhofer 2011). We evaluate the DL queries using the Hermit reasoner (Glimm et al. 2014). The provenance artifacts are stored in a Neo4j graph database (Miller 2013). The rules are implemented in Java 1.8. For the future, we plan using a rule language to enable users without Java-programming skills to extend the rule set. The experiment generation steps based on schemas and the used technologies have been described in a previous publication (Wilsdorf et al. 2019). We have bindings to three different backends for SA: the Python library *SALib* (Herman and Usher 2017), the R library *sensitivity* (Iooss et al. 2019), and SESSL (Ewald and Uhrmacher 2014), each of which implements a subset of the mentioned methods. User inputs are requested via GUIs. The source code for our approach and the case study are available at https://git.informatik.uni-rostock.de/mosi/exp-generation/-/tree/master/src/main/java/org/mosi/grease/caseStudies/WSC2021.

5 CASE STUDY: Wnt/β-Catenin Simulation Study

We continue with the best practice rules by (Saltelli et al. 2019) and our ontology of SA methods based on (Pianosi et al. 2016; Borgonovo and Plischke 2016), and now aim to support a real simulation study by applying these rules. As an example, we use a simulation study of the Wnt/β-catenin signaling pathway (Haack et al. 2015), which is a central pathway that regulates proliferation as well as differentiation of cells. Deregulated forms of this pathway are involved in a number of human cancers and developmental disorders. In this simulation study, membrane-related processes are in the focus to study the effect of lipid rafts on the Wnt signaling mechanism. The model is implemented in the rule-based modeling language ML-Rules (Maus et al. 2011), and simulation runs are carried out using SESSL (Ewald and Uhrmacher 2014). Figure 4 depicts the model with its two submodels and 24 reactions.

The provenance of the simulation study has been recorded by (Budde et al. 2021). It is a compact representation where many smaller steps have been aggregated in order to reveal the essential activities regarding model building, model calibration, and model validation. In Figure 5 we show a snapshot of the provenance graph. For the purpose of this case study, we enriched the provenance information with further artifacts (highlighted). So far, a conceptual model with a research objective, five assumptions, a qualitative model, and various wet-lab data has been established. Based on this information a simulation

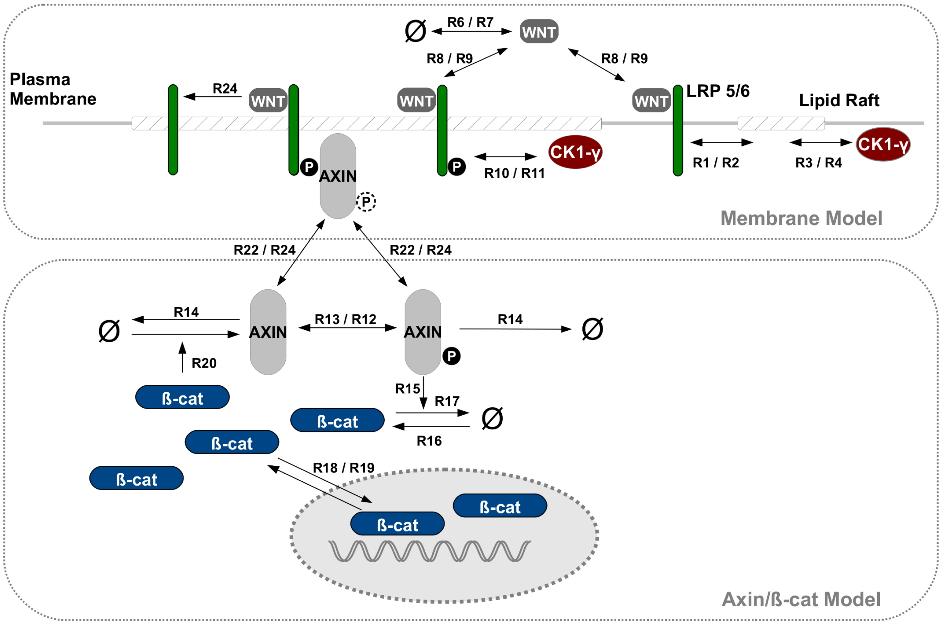

Figure 4: Sketch of the Wnt model illustrating the submodels, entities, and reactions.

model `SM1` was developed. The model was calibrated in the activity `CSM1` via two simulation experiments and wet-lab data `WD2`, which produced the calibrated model `SM2`. The updated model was used in two validation activities (`VSM1` and `VSM2`) which involved a number of simulation experiments and wet-lab data. The activity `ASM1` denotes the new analyzing simulation model activity that we want to fill with a concrete SA experiment `SE7` (and corresponding simulation data `SD7`). In the following, we will walk through the steps of our approach, and demonstrate how the provenance information of the Wnt model is central in making decisions about the SA experiment.

In the first step, the rules regarding the applicability of the experiment type “sensitivity analysis” are checked. Since the new activity is of type ASM, our approach checks whether a UQ has already been conducted. An experiment of type UQ is discovered, denoted by `SE1'`. As the UQ was conducted in a Master thesis (Sandmeir 2018), it was not part of the original study, and therefore introduced by us as an additional entity for this case study. The UQ focused on the parameters that were introduced by the model extension and fitted based on wet-lab data. All other parameters of the model were obtained from previous model versions as well as from literature and were thus considered established. The experiment results provided a posteriori distributions for six ($kLWntBind$, $kLWntUnbind$, $kLphos$, $kLdephos$, $kLAxinBind$, $kLAxinUnbind$) of the 18 model parameters. The obtained distributions, as well as upper and lower bounds, were recorded as additional assumptions in the provenance (`A6-10`) to make the uncertainties of the model explicit. Overall, the uncertainty was still relatively high for the six fitted parameters. One reason for the high uncertainty is the lack of experimental data and the variability of the experimental approach that was available for the parameter fitting. Another reason might be the robustness of the model, i.e., variations in the parameters chosen for the UQ may not significantly affect the model outcome.

The second rule regarding the choice of experiment type checks whether the model is already validated. In the provenance graph, we see that all wet-lab data have been used either for calibration or validation. By looking closer at the simulation data artifacts, we find that all validation experiments were indeed successful. As the simulation study contains no requirement artifacts, no further validation experiments are needed.

Since the precondition for a SA as analysis activity is fulfilled, we continue with the second step. Here, the rules regarding the analysis methods are evaluated. First, it is checked whether there is an assumption regarding the linearity of the model. No such assumption was specified by the modeler, as the Wnt model contains a negative feedback loop. Consequently, only global SA methods are selected.

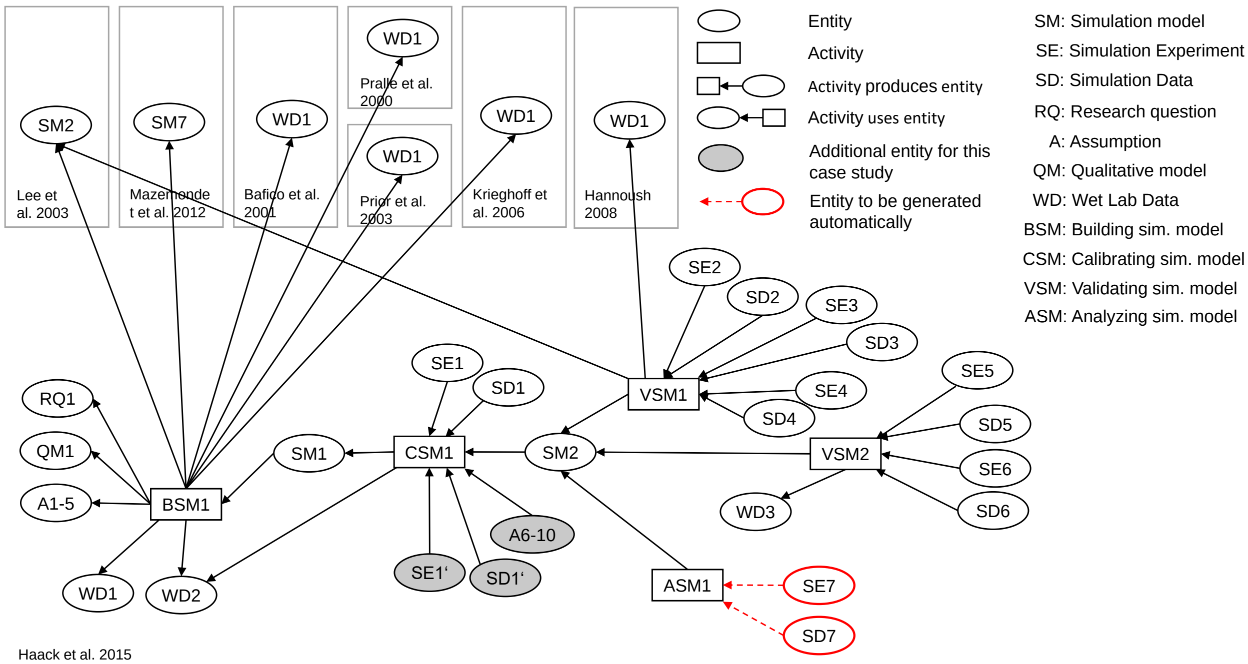

Figure 5: Provenance graph of the Wnt/β-catenin simulation study by (Haack et al. 2015). The Figure was taken from (Budde et al. 2021) and modified for the purpose of this paper.

Furthermore, according to the next rule (best practice #4), only methods that can determine interaction effects, and allow factor ranking shall be selected. Based on the evaluation of these rules, a query is generated to extract all suitable methods from the ontology of SA methods. Figure 6 shows the generated DL query and the query result.

```
Query: GlobalMethod and hasPurpose some Ranking and
         hasSpecialty some Interactions
Result: FAST, Morris, Sobol
```

Figure 6: DL query for finding all suitable SA methods for the current simulation model.

In the third step, the rules regarding the parametrization of the selected methods are evaluated. For the Wnt simulation model, the qualitative model tells us that the observed model quantity is the amount of β-catenin accumulated inside the cell nucleus $Cell/Nuc/Bcat$.

For the parametrization of the global SA methods, the parameter names, ranges as well as distributions are required. Due to the existing assumptions (`A6-10`) on the reaction rate constants $kLWntBind$, $kLWntUnbind$, $kLphos$, $kLdephos$, $kLAxinBind$, and $kLAxinUnbind$, these parameters are considered relevant for a SA according to best practice rule #6, and their information is extracted from the assumption artifacts. As we do not focus on runtime performance here, for the sample sizes we automatically set default values of the used tools. Future work could aim at including rules about the performance which is an often studied question for different types of models (see, e.g., (Nguyen and Reiter 2015)).

For this case study, the user however manually reduces the runtime to 500 samples as the model is computationally expensive. However, the selected inputs and outputs are not adapted further, and the choice of three SA methods (FAST, Morris, Sobol) is kept since the user is interested in comparing different sensitivity indices. Finally, from the collected information executable experiment code is generated by using the pipeline of (Wilsdorf et al. 2019), and the experiments are executed. Figure 7 shows the generated Python code for the Sobol method. The function `run_wnt_model` starts simulation runs with SESSL for the configuration samples created in `X`. This last step of our pipeline automatically fills the new entities `SE7` and `SD7` in the provenance graph, and also establishes a link between the assumptions `A6-10` and `ASM1` to make explicit which assumptions went into the analysis and where they came from. The completed provenance activity is shown in Figure 8.

```python
from SALib.sample import saltelli
from SALib.analyze import sobol

problem = {
  'num_vars': 6,
  'names': ['kLWntBind', 'kLWntUnbind', 'kLphos', 'kLdephos', 'kLAxinBind',
            'kLAxinUnbind'],
  'bounds': [
    [50, 180], [0.05, 0.5], [0.075, 0.9], [1.5E-2, 9.5E-2], [2, 8], [1E-4, 5E-3]]
}
X = saltelli.sample(problem, 500, calc_second_order=True)
Y = run_wnt_model(X, problem)
S = sobol.analyze(problem, Y, calc_second_order=True)
```

Figure 7: Generated code for applying the Sobol method to the Wnt model.

Figure 9 shows the sensitivity indices obtained by the different methods. The first-order and μ^* indices provide a measure of sensitivity when each model parameter is tested individually, whereas the total-order and σ indices provide a measure for the interaction effects between the parameters. Our results show that neither method could identify a clear primary cause for changes in the model output, with all individual effects near zero. Only when changing combinations of multiple parameters, effects are observed. This result is in line with the previous UQ as both experiments indicate high robustness of the model with respect to the tested parameters. Indeed, there are other parameters that have a much stronger impact on the model outcome, as shown in another SA experiment with a different set of parameters, given in the accompanying Git repository.

6 RELATED WORK

Our work builds on surveys about SA which provide guidelines to the reader as to when to apply which methods (Pianosi et al. 2016; Borgonovo and Plischke 2016). This information we used to develop an ontology about SA, as we needed the information to be accessible and queryable by our approach. The definition of ontologies is not new in the area of modeling and simulation, e.g., for categorizing different approaches of discrete-event simulation (Silver et al. 2011).

Another central ingredient of our approach are accessible and well-structured experiment descriptions. Templates or schemas can be used to generate executable simulation experiments of various types (Ruscheinski et al. 2018) for different targets (Wilsdorf et al. 2019). The importance of having explicit specifications of experiments to facilitate reproducibility and reuse is reflected by the number of developments over the last decade: developing standardized formats (Waltemath et al. 2011), domain-specific languages (Ewald and Uhrmacher 2014), or model-based approaches for specifying simulation experiments (Teran-Somohano et al. 2015).

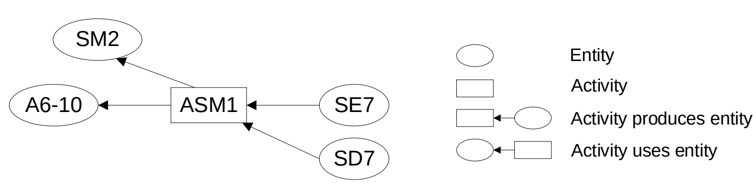

Figure 8: Automatically completed activity. Compare with provenance graph of Figure 5.

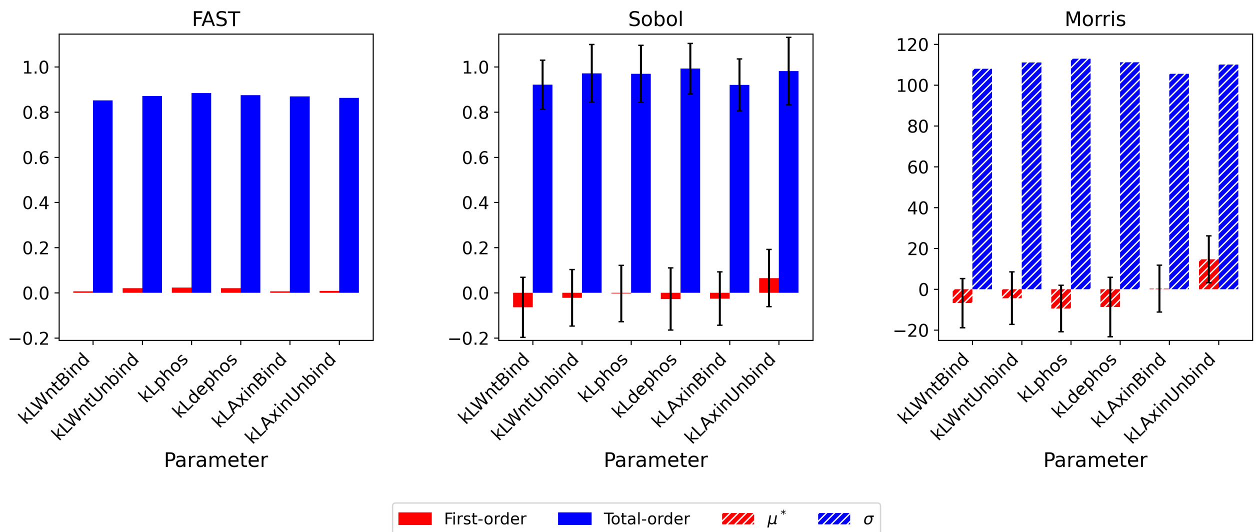

Figure 9: First- and total-order sensitivity indices obtained by the FAST and Sobol method, and mean elementary effects (μ^*) and their standard deviations (σ) obtained by the Morris method. Error bars are given if confidence intervals are computed by the SALib implementations.

Our approach takes context information about the simulation study into account. Several previous works focus on reusing explicit behavioral requirements. E.g., requirements defined as temporal logic formulas have been used to automatically generate and execute statistical model checking experiments (Ruscheinski et al. 2019). Also, hypotheses can automatically be tested (Lorig 2019; Yilmaz et al. 2016). Furthermore, statistical model checking experiments with explicit behavioral properties have been reused automatically for extended or composed models (Peng et al. 2017).

Provenance information has been used before to reveal the relations between various simulation models to support their interpretation or to develop new, refined models (Budde et al. 2021). Furthermore, provenance about executed simulation experiments can reduce the effort of computing platforms by avoiding duplicate simulations (Ma et al. 2019). These are only a few possibilities to exploit provenance for conducting simulation studies (Suh and Lee 2018). Rather recent is its use to support in combination with other knowledge sources to automatically generate simulation experiments (Wilsdorf et al. 2020).

7 CONCLUSIONS AND FUTURE WORK

Supporting users by considering context information as well as methodological knowledge is crucial for conducting more consistent, systematic, and efficient simulation studies. Therefore, we exploited provenance and an ontology of methods for supporting best practices for simulation experiments, to automatically generate and execute simulation experiments. As a case study, we used best practices for SA which referred to the applicability of the experiment type, the choice of SA methods, and their parametrization. We demonstrated the approach using a simulation study from cell biology and showed that UQ and SA could effectively be combined, and executable SA experiments could be automatically generated.

As next steps, we will complement the knowledge about SA collected so far, e.g., using further literature reviews or expert interviews. Moreover, we want to extend our work by best practice rules for other experiment types such as optimization, statistical model checking, etc. There, different implications might exist between the provenance knowledge and the choice of analysis, and thus different rules have to be defined. Furthermore, our approach for supporting best practices with provenance and ontologies can be expanded from simulation experiments to other steps in the modeling and simulation life cycle.

ACKNOWLEDGMENTS

This work was funded by the research grants DFG UH-66/18 'GrEASE', DFG UH-66/15 'MoSiLLDe', and DFG CRC 1270 'Elaine'.

REFERENCES

Balci, O. 2012. "A Life Cycle for Modeling and Simulation". *SIMULATION* 88(7):870–883.

Belhajjame, K., R. B'Far, J. Cheney, S. Coppens, S. Cresswell, Y. Gil, P. Groth, G. Klyne, T. Lebo, J. McCusker et al. 2013. "PROV-DM: The PROV Data Model". *W3C Recommendation*.

Borgonovo, E., and E. Plischke. 2016. "Sensitivity Analysis: A Review of Recent Advances". *European Journal of Operational Research* 248(3):869–887.

Budde, K., J. Smith, P. Wilsdorf, F. Haack, and A. M. Uhrmacher. 2021. "Relating Simulation Studies by Provenance—Developing a Family of Wnt Signaling Models". *PLOS Computational Biology*. https://doi.org/10.1371/journal.pcbi.1009227. In Press.

Eriksson, O., A. Jauhiainen, S. Maad Sasane, A. Kramer, A. G. Nair, C. Sartorius, and J. Hellgren Kotaleski. 2018. "Uncertainty Quantification, Propagation and Characterization by Bayesian Analysis Combined with Global Sensitivity Analysis Applied to Dynamical Intracellular Pathway Models". *Bioinformatics* 35(2):284–292.

Ewald, R., and A. M. Uhrmacher. 2014. "SESSL: A Domain-specific Language for Simulation Experiments". *ACM Transactions on Modeling and Computer Simulation* 24(2):11:1–11:25.

Glimm, B., I. Horrocks, B. Motik, G. Stoilos, and Z. Wang. 2014. "HermiT: an OWL 2 Reasoner". *Journal of Automated Reasoning* 53(3):245–269.

Gray, J., J. Hilton, and J. Bijak. 2017. "Choosing the Choice: Reflections on Modelling Decisions and Behaviour in Demographic Agent-Based Models". *Population Studies* 71(sup1):85–97. PMID: 29061095.

Haack, F., H. Lemcke, R. Ewald, T. Rharass, and A. M. Uhrmacher. 2015. "Spatio-Temporal Model of Endogenous ROS and Raft-Dependent WNT/Beta-Catenin Signaling Driving Cell Fate Commitment in Human Neural Progenitor Cells". *PLOS Computational Biology* 11(3).

Helms, T., R. Ewald, S. Rybacki, and A. M. Uhrmacher. 2015, October. "Automatic Runtime Adaptation for Component-Based Simulation Algorithms". *ACM Transactions on Modeling and Computer Simulation* 26(1).

Herman, J., and W. Usher. 2017. "SALib: an Open-Source Python Library for Sensitivity Analysis". *Journal of Open Source Software* 2(9):97.

Horridge, M., and S. Bechhofer. 2011. "The OWL API: A Java API for OWL Ontologies". *Semantic Web* 2(1):11–21.

Horridge, M., N. Drummond, J. Goodwin, A. L. Rector, R. Stevens, and H. Wang. 2006. "The Manchester OWL Syntax.". In *OWLed*, Volume 216.

Iooss, B., A. Janon, G. Pujol, with contributions from Baptiste Broto, K. Boumhaout, S. D. Veiga, T. Delage, J. Fruth, L. Gilquin, J. Guillaume, L. Le Gratiet, P. Lemaitre, B. L. Nelson, F. Monari, R. Oomen, O. Rakovec, B. Ramos, O. Roustant, E. Song, J. Staum, R. Sueur, T. Touati, and F. Weber. 2019. *Sensitivity: Global Sensitivity Analysis of Model Outputs*. R package version 1.16.1.

Leye, S., R. Ewald, and A. M. Uhrmacher. 2014, 04. "Composing Problem Solvers for Simulation Experimentation: A Case Study on Steady State Estimation". *PLOS ONE* 9(4):1–13.

Lorig, F. 2019. *Hypothesis-Driven Simulation Studies: Assistance for the Systematic Design and Conducting of Computer Simulation Experiments*. Springer.

Ma, J., S. Lee, K. W. Cho, and Y.-K. Suh. 2019, March. "A Simulation Provenance Data Management System for Efficient Job Execution on an Online Computational Science Engineering Platform". *Cluster Computing* 22(1):147–159.

Maus, C., S. Rybacki, and A. M. Uhrmacher. 2011. "Rule-based Multi-level Modeling of Cell Biological Systems". *BMC Systems Biology* 5(1):166.

Miller, J. J. 2013. "Graph Database Applications and Concepts with Neo4j". In *Proceedings of the Southern Association for Information Systems Conference, Atlanta, GA, USA*, Volume 2324.

Musen, M. A. 2015. "The Protégé Project: a Look Back and a Look Forward". *AI Matters* 1(4):4–12.

Neal, M. L., C. T. Thompson, K. G. Kim, R. C. James, D. L. Cook, B. E. Carlson, and J. H. Gennari. 2018, 09. "SemGen: a Tool for Semantics-based Annotation and Composition of Biosimulation Models". *Bioinformatics* 35(9):1600–1602.

Nguyen, A.-T., and S. Reiter. 2015. "A Performance Comparison of Sensitivity Analysis Methods for Building Energy Models". In *Building Simulation*, Volume 8, 651–664. Springer.

Peng, D., T. Warnke, F. Haack, and A. M. Uhrmacher. 2017. "Reusing Simulation Experiment Specifications in Developing Models by Successive Composition — a Case Study of the Wnt/β-catenin Signaling Pathway". *SIMULATION* 93(8):659–677.

Pianosi, F., K. Beven, J. Freer, J. W. Hall, J. Rougier, D. B. Stephenson, and T. Wagener. 2016. "Sensitivity Analysis of Environmental Models: A Systematic Review with Practical Workflow". *Environmental Modelling & Software* 79:214–232.

Ruscheinski, A., K. Budde, T. Warnke, P. Wilsdorf, B. C. Hiller, M. Dombrowsky, and A. M. Uhrmacher. 2018. “Generating Simulation Experiments based on Model Documentations and Templates”. In *Proceedings of the 2018 Winter Simulation Conference*, edited by M. Rabe, A. A. Juan, N. Mustafee, A. Skoogh, S. Jain, and B. Johansson, 715–726. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Ruscheinski, A., and A. Uhrmacher. 2017. “Provenance in Modeling and Simulation Studies — Bridging Gaps”. In *Proceedings of the 2017 Winter Simulation Conference*, edited by W. K. V. Chan, A. D’Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, 872–883. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Ruscheinski, A., T. Warnke, and A. M. Uhrmacher. 2019. “Artifact-based Workflows for Supporting Simulation Studies”. *IEEE Transactions on Knowledge and Data Engineering* 32(6):1064–1078.

Saltelli, A., K. Aleksankina, W. Becker, P. Fennell, F. Ferretti, N. Holst, S. Li, and Q. Wu. 2019. “Why so Many Published Sensitivity Analyses are False: A Systematic Review of Sensitivity Analysis Practices”. *Environmental Modelling & Software* 114:29–39.

Sandmeir, L. 2018. “Parameter Estimation for Rule-Based Models with Approximate Bayesian Computation”. Master’s thesis, Department of Mathematics, Technische Universität München.

Sargent, R. G. 2013. “Verification and Validation of Simulation Models”. *Journal of Simulation* 7(1):12–24.

Silver, G. A., J. A. Miller, M. Hybinette, G. Baramidze, and W. S. York. 2011. “An Ontology for Discrete-Event Modeling and Simulation”. *SIMULATION* 87(9):747–773.

Suh, Y.-K., and K. Y. Lee. 2018, September. “A Survey of Simulation Provenance Systems: Modeling, Capturing, Querying, Visualization, and Advanced Utilization”. *Human-centric Computing and Information Sciences* 8(1):27.

ten Broeke, G., G. van Voorn, and A. Ligtenberg. 2016. “Which Sensitivity Analysis Method Should I Use for My Agent-Based Model?”. *Journal of Artificial Societies and Social Simulation* 19(1):5.

Teran-Somohano, A., A. E. Smith, J. Ledet, L. Yilmaz, and H. Oğuztüzün. 2015. “A Model-Driven Engineering Approach to Simulation Experiment Design and Execution”. In *Proceedings of the 2015 Winter Simulation Conference*, edited by L. Yilmaz, W. K. V. Chan, I. Moon, T. M. K. Roeder, C. Macal, and M. D. Rossetti, 2632–2643. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Villaverde, A. F., F. Fröhlich, D. Weindl, J. Hasenauer, and J. R. Banga. 2018, 08. “Benchmarking Optimization Methods for Parameter Estimation in Large Kinetic Models”. *Bioinformatics* 35(5):830–838.

Waltemath, D., R. Adams, F. T. Bergmann, M. Hucka, F. Kolpakov, A. K. Miller, I. I. Moraru, D. Nickerson, S. Sahle, J. L. Snoep et al. 2011. “Reproducible Computational Biology Experiments with SED-ML-the Simulation Experiment Description Markup Language”. *BMC Systems Biology* 5(1):198.

Wilsdorf, P., M. Dombrowsky, A. M. Uhrmacher, J. Zimmermann, and U. v. Rienen. 2019. “Simulation Experiment Schemas –Beyond Tools and Simulation Approaches”. In *Proceedings of the 2019 Winter Simulation Conference*, edited by blah, 2783–2794. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Wilsdorf, P., F. Haack, K. Budde, A. Ruscheinski, and A. M. Uhrmacher. 2020. “Conducting Systematic, Partly Automated Simulation Studies – Unde Venis et Quo Vadis”. *AIP Conference Proceedings* 2293(1):020001.

Wilsdorf, P., F. Haack, and A. M. Uhrmacher. 2020. “Conceptual Models in Simulation Studies: Making it Explicit”. In *Proceedings of the 2020 Winter Simulation Conference*, edited by K.-H. Bae, B. Feng, S. Kim, S. Lazarova-Molnar, Z. Zheng, T. Roeder, and R. Thiesing. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Yilmaz, L., S. Chakladar, and K. Doud. 2016. “The Goal-Hypothesis-Experiment framework: A generative cognitive domain architecture for simulation experiment management”. In *Proceedings of the 2016 Winter Simulation Conference*, edited by T. M. K. Roeder, P. I. Frazier, R. Szechtman, E. Zhou, T. Huschka, and S. E. Chick, 1001–1012. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

AUTHOR BIOGRAPHIES

PIA WILSDORF is a Ph.D. candidate in the Modeling and Simulation group at the University of Rostock. Her e-mail address is pia.wilsdorf@uni-rostock.de.

NADINE FISCHER is a Master’s student of Mathematics and a student assistant in the Modeling and Simulation group at the University of Rostock. The e-mail address is nadine.fischer@uni-rostock.de.

FIETE HAACK is a Postdoctoral researcher in the Modeling and Simulation group at the University of Rostock. His e-mail address is fiete.haack@uni-rostock.de.

ADELINDE M. UHRMACHER is a Professor at the Institute for Visual and Analytic Computing, University of Rostock, and head of the Modeling and Simulation group. Her e-mail address is adelinde.uhrmacher@uni-rostock.de.