Proceedings of the 2025 Winter Simulation Conference
E. Azar, A. Djanatliev, A. Harper, C. Kogler, V. Ramamohan, A. Anagnostou, and S. J. E. Taylor, eds.

GOAL-ORIENTED GENERATION OF SIMULATION EXPERIMENTS

Anja Wolpers[1], Pia Wilsdorf[1], Fiete Haack[1], and Adelinde M. Uhrmacher[1]

[1]Institute for Visual and Analytic Computing, University of Rostock, Rostock, GERMANY

ABSTRACT

Automatically generating and executing simulation experiments promises to make running simulation studies more efficient, less error-prone, and easier to document and replicate. However, during experiment generation, background knowledge is required regarding which experiments using which inputs and outputs are useful to the modeler. Therefore, we conducted an interview study to identify what types of experiments modelers perform during simulation studies. From the interview results, we defined four general goals for simulation experiments: *exploration*, *confirmation*, *answering the research question*, and *presentation*. Based on the goals, we outline and demonstrate an approach for automatically generating experiments by utilizing an explicit and thoroughly detailed conceptual model.

1 INTRODUCTION

Simulation experiments are essential to simulation studies. For example, validation experiments test if the simulation model correctly represents the real-world system, calibration experiments optimize the simulation model's input parameters to reproduce real-world data accurately, and parameter scan experiments show how the model's behavior changes under different conditions. In short, simulation experiments are crucial for ensuring that a simulation study is meaningful, accurate, and useful for decision-making and problem-solving. Therefore, automated support for experimenting with simulation models by generating experiments and exploiting explicit context information about the simulation study seems imperative for successful, efficient, rigorous, and credible simulation studies (Uhrmacher et al. 2024). In addition, automatically generated experiments provide a clear record of the experiment setups and results, allowing the provenance of the simulation study to be automatically documented without additional effort.

However, to create helpful simulation experiments for modelers, we need to understand what kinds of experiments are executed with which goals in non-automated simulation studies. To this end, we conducted an interview study asking modelers when they execute what experiments with which goals. In these interviews, we identified four distinctive goals that we used as a foundation for automated experiment generation, i.e., *exploration*, *confirmation*, *answering the research question*, and *presentation*, drawing on terminology from visual analytics (Schulz et al. 2013).

We make the background knowledge about the relationship between goals and experiment types explicit and demonstrate how it can be utilized to automatically generate simulation experiments from scratch. Our approach and prototypical implementation support modelers by recommending a list of experiments tailored to the current goal and the context in which the model is built and analyzed. From the suggestions, modelers can select the experiments they wish to run, and our prototype will automatically extract relevant context information and populate templates to produce an executable experiment in a target specification language. Context information about a simulation model includes the research questions, behavioral requirements, and input and output definitions – all of which belong to the *conceptual model* (Robinson 2008). Thus far, individual parts of the conceptual model have been used for experiment generation, e.g., formally specified requirements for generating hypothesis tests (Peng et al. 2016; Yilmaz et al. 2016; Lorig 2019; Ruscheinski et al. 2018) or input parameter tables for generating full factorial experiment designs (Ruscheinski et al. 2018). Here, we aim to exploit the entire conceptual model – not only to fully specify a simulation experiment (e.g., to correctly set initial values and time points for observation) but also to constrain the list of suggested experiment types. For instance, validation experiments may only be suggested if a requirement exists in the conceptual model that defines a validation data set to be reproduced or a temporal-logic formula to be satisfied by the model.

The paper is structured as follows: First, Section 2 presents the design and results of our interview study. This is followed by Section 3, which interprets the study results in goal-based phases for simulation studies. Section 4 then introduces the experiment generator and how it exploits the contents of the conceptual model. In Section 5, we apply our goal-based approach by automatically generating an exploratory experiment in the context of a cell biological simulation study and discuss how existing experiment generators could support the different phases. Finally, in Section 6, we discuss open questions and future research directions.

2 INTERVIEW STUDY

To get a realistic overview of the types of experiments and their goals in simulation studies, we conducted qualitative interviews with modelers, asking what experiments they use in their simulation studies and why. We decided to use semi-structured interviews to learn about modelers' work processes, which requires asking follow-up questions for clarity. We interviewed four modelers from the fields of epidemiology, biology, and ecology. Because the modelers' and our native language is German, we performed the interviews in German. We argue that the relaxed atmosphere sets the tone for the modelers to also talk about intermediate experiments and experiments that they might deem less important. In the following, we will summarize the interview results in English.

In the interviews, we talked with the modelers about their general approach to designing simulation models and conducting simulation studies. As the first step, we asked them to tell us what questions about their simulation model they answered using model/experiment executions. These questions will be our guide to *why* or with what *goal* modelers design and execute simulation experiments. Then, we asked them to explain how they answered these questions. This relates to the types and roles of experiments they used. Experiment types describe how to execute simulations, what inputs are used, and how the results are calculated. For example, parameter scans are a type of experiment in which multiple model executions are run with varying input parameter configurations, e.g., based on specific experiment designs. Another example are optimization experiments, which minimize (or maximize) a goal function relating to the simulation model's output parameters by trying out different parameter configurations. In contrast, an experiment's role refers to its purpose in the simulation study, such as validation, calibration, or analysis. As such, one role can be realized by different types of experiments, and one type of experiment can have different roles during a simulation study. For example, *validation* experiments can have the type *statistical model checking* if the simulation model is checked against a logical formula or the type *parameter scan* if various parameter configurations are executed and the modeler visually assesses the output data.

For the most part, the interviewed modelers used experiments to answer similar questions. The most prominent questions were: *Does the model run (without errors)? Does the model roughly behave as expected (after the recent edit)? Is the run time feasible? Can I simplify the model? Can the model reproduce lab data or data from another model?* To answer these questions, most modelers individually described using combinations of simple runs, parameter scans, and visualizations of the resulting simulation data. In contrast to the parameter scan, we understand a simple run to mean that the simulation model is executed with a single configuration of input parameters while one or a set of outputs is observed (the execution can include multiple replications for stochastic models). When assessing whether the simulation model behaves roughly as expected, the modelers reported using simple runs or parameter scans to obtain simulation data that they then visualize to form an opinion based on their expertise. Similarly, they compare the visualized results of simple runs to the plots of data obtained from a reference model or lab data to find out if their model can reproduce said data or if their previous model version behaves similarly enough to their new simplified model.

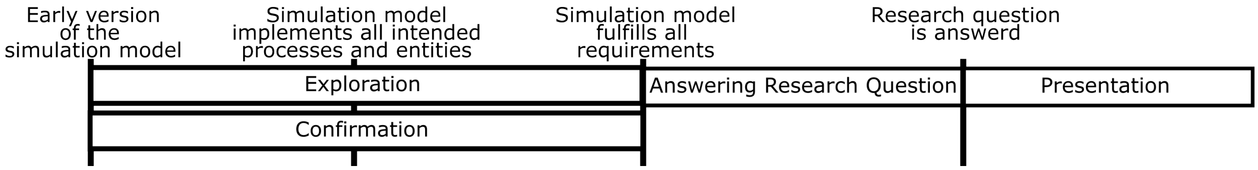

Figure 1: The experiments discussed in the interviews with modelers grouped by their general goals on a timeline and milestones of simulation studies.

Some modelers explicitly discussed using optimization experiments to calibrate their simulation models to real-world data. Others explained that in their domain, there is not enough fitting data available that they could use to calibrate their models. Instead, they use all available data for validating their modeled (sub-)processes. Also, one modeler said they run experiments explicitly designed to present their findings. They explained that their results, to be plotted and published, represent only a subset of trajectories from a larger parameter scan. So, it may be challenging for reviewers and readers to locate and reproduce the exact trajectories used from the original parameter scan. Therefore, they design and run additional, targeted experiments specifically configured for the plots and data to be presented in the publication.

The answers of the interviewed modelers differed for the experiments regarding the research question. In most interviews, the modelers remarked that these questions and the experiments to answer them depend on the nature of the research question itself. If the research question is about examining how a specific intervention affects the simulated process, then a so-called what-if analysis would be conducted consisting of a parameter scan (if the different scenarios can be reflected by changing input variables) or simple runs of, respectively, altered model versions that are compared (if different scenarios mean changing the model structure). For example, the research question could also be about examining what parameters a certain model output is sensitive to. This type of research question would be answered using a sensitivity analysis. These are only some examples of experiment types linked to the research question given by the interviewees. As the type of experiment depends on the research question, other types of experiments are also possible in this phase.

As the next step, we presented the modelers with a timeline marked by the following milestones: 1. “early version of simulation model”, 2. “simulation model implements all intended processes and entities”, 3. “simulation model fulfills all requirements”, and 4. “research question is answered”. We requested the modelers to put the experiments discussed in the previous questions on this timeline, showing when they performed them in their simulation studies. Based on where in the timeline modelers placed the experiments and based on the experiments’ questions, we found that experiments with similar questions can be grouped under four general goals (see Figure 1). All modelers considered experiments that *explore* the model’s approximate behavior to be relevant from milestone 1 through milestone 3. These experiments answer questions like *Does the model run (without errors)?, Does the model roughly behave as expected (after the recent edit)?, Can the model roughly replicate a reference model?, Is the run time feasible?* and *Can I simplify the model?*. As a next general goal, we summarize experiments used for *confirming* that the simulation model reflects the modeled system close enough – by comparing its outputs to real-world or simulation data – which were also placed between milestones 1 and 3. Experiments used to *answer research questions* were located between milestones 3 and 4, and experiments designed to single out trajectories to *present* their results in publications were placed after milestone 4. In the following, we will go into more detail about how these general goals relate to the experiments associated with them.

3 EXPERIMENT GOALS AND ASSOCIATED EXPERIMENT TYPES

Based on the general goals identified in the interview study above, we define four phases of simulation studies and associate them with the experiment roles and types discussed by the interviewed modelers. The phases based on the experiment goals are: *1. Exploration phase*, *2. Confirmation phase*, *3. Answering research question phase*, and *4. Presentation phase* (see Figure 2). In the following, we will describe the

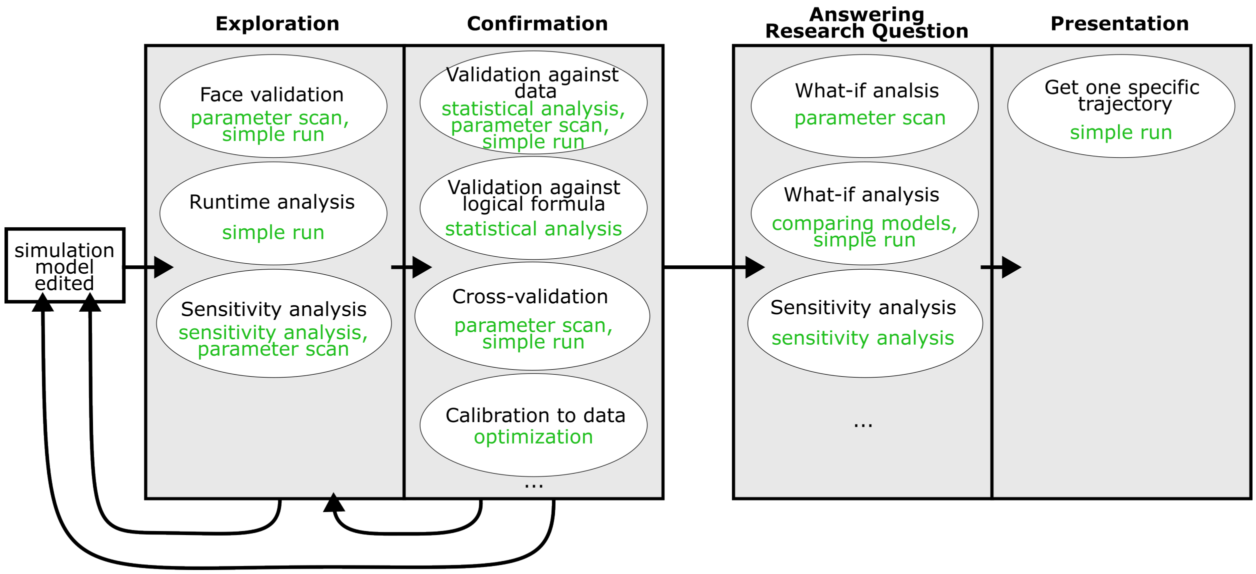

Figure 2: Goals of simulation experiments and their roles (black) and types (green, below) based on our interviews. The exploration and presentation phases include all experiments discussed in the interviews. For the confirmation and answering research question phases, we only included experiments that were discussed in detail.

properties defining these phases before going into more detail on how they can be utilized to generate simulation experiments in later sections.

1. Exploration Phase After editing the simulation model, modelers test whether their changes have roughly the intended result without unwanted effects on other simulated processes. Suppose executing the model shows behavior different than expected; in this case, the modeler will return to editing the simulation model and then test it again.

Because this first testing is highly iterative, the executions must finish fast to minimize waiting time between consecutive model edits. The model's behavior does not have to be tested for all edge cases and in detail, as rough results suffice to identify unexpected behavior.

We called this phase *exploration phase*, named after the visualization goal with the same name (Schulz et al. 2013). In visualization, exploratory analyses aim to discover patterns, trends, hypotheses, and insights within a data set. Users interactively explore large, complex data sets. In modeling and simulation, the modeler interactively explores their simulation model to get a sense of its behavior. They discover trends and patterns in the simulation output, which helps to refine hypotheses and generate new insights for refining the model (or executing further simulation experiments).

Experiments in the exploration phase play the roles of validation (face validation and cross-validation) and analysis (sensitivity analysis). Prominently, the experiment types simple runs and parameter scans are used for face validation (see section 2). A face validation means that experts for the modeled system (which often are the modelers themselves) assess if the model's "input-output relationships are reasonable" (Sargent 2010). So, modelers execute a simple run or parameter scan and then look at the visualized results. If the modeler suspects specific problems, they execute specific runs or more constrained parameter scans. Another application for simple runs in the exploration phase is measuring the approximate run time of model executions. If a simulation run takes too long, the modeler might choose to simplify the model. Alternatively, they might choose to work with a smaller (sub)model until they need more detailed analyses. Some modelers also reported conducting experiments with the role sensitivity analysis in the exploration phase. These may be realized as an actual sensitivity analysis (e.g., using the methods of Morris or Sobol (Pianosi et al. 2016)) or as a parameter scan with face validation.

2. Confirmation Phase After modelers finish a first check exploring their model's behavior and believe that their model does (coarsely) what they intend, they continue to examine their model (version) in more detail. In the confirmation phase, simulation experiments are executed with the roles of calibration (can the model be parameterized to fit the data or otherwise described expectations) and validation (does the model corroborate observations made, as a time series or otherwise). Central in this phase is information about input parameters (specific values or their range), the initial model state, and data about the expected output (e.g., as time series or logical formulas) defining various requirements on the simulation model and its behavior. This information is typically part of the conceptual model (Robinson 2008; Wilsdorf et al. 2020). In visualization, a confirmatory analysis aims to test the hypotheses or insights uncovered during exploration. In simulation studies, the requirements defined in the conceptual models are usually tested to be either confirmed or disproved. Some of these requirements can also be defined based on insights from the exploration phase. In the case of unsuccessful validation, further refinements of the simulation models may be required. It should be noted that calibration experiments might also be used to refute a simulation model as being valid, e.g., if a simulation model cannot be fitted to various data with plausible parameter values (Haack et al. 2020). Independent of this, validation requires independent data or information sources (not used during calibration) to test the simulation model against.

Of the several methods for model validation discussed in the literature (Leye et al. 2009; Sargent 2010), the interviewed modelers named validating the simulation model against data or a logical formula and cross-validation with other simulation models. In the latter case, comparing a model's new version to a previous one can tell the modeler if a simplification between the previous model version has any impact on the output, depending on the model's input configuration.

Confirmatory methods include statistical tests, simulation-based model checking, and corroboration with additional data sets (Leye et al. 2009). Some modelers explained that requirements from the conceptual model could refer to different aspects of model behaviors (e.g., whether its input uncertainty is required to be low), which would indicate specific analysis methods (e.g., Bayesian uncertainty quantification). We did not include these experiments in Figure 2, because we did not discuss them in detail. However, we understand that the list of experiments in this phase is incomplete. Similar to the exploration phase, most participating modelers emphasized the importance of suitable visualization support for analyzing the results (or the process) of their confirmatory analysis.

3. Answering Research Question Phase Once the simulation model meets all requirements defined in the conceptual model, the simulation model can be used to address the research question in the *answering research question* phase. All experiments discussed in the interviews associated with this phase have the role analysis. The experiment types conducted during this phase vary depending on the nature of the research question. A research question can be open (like "What happens if we raise the number of wolves in the initial model state?") or closed (like "Does a higher number of wolves in the model's initial state lead to the prey dying out?").

Experiments investigating closed research questions can appear very similar to experiments from the confirmation phase, as they use the same experimental methods. Also, the hypothesis in a closed research question can be either *confirmed* or denied. However, experiments in the answering research question phase do not have to verify that this hypothesis is true. In the confirmation phase, in contrast, the experiment's results have to match the expected output (e.g., in the form of data) to confirm that the simulation model correctly reflects the modeled process.

Some research questions may call for a what-if analysis, where input parameters or parts of the simulation model are modified to analyze how changes impact the simulation output. These experiments often involve comparing simple simulation runs or parameter scans. Other research questions might necessitate a detailed sensitivity analysis to identify which parameters drive specific behaviors in the system. In some simulation studies (that are focused on explaining the underlying mechanisms), the insights gained during the simulation model's development may already provide sufficient understanding of the modeled process. So, the study's

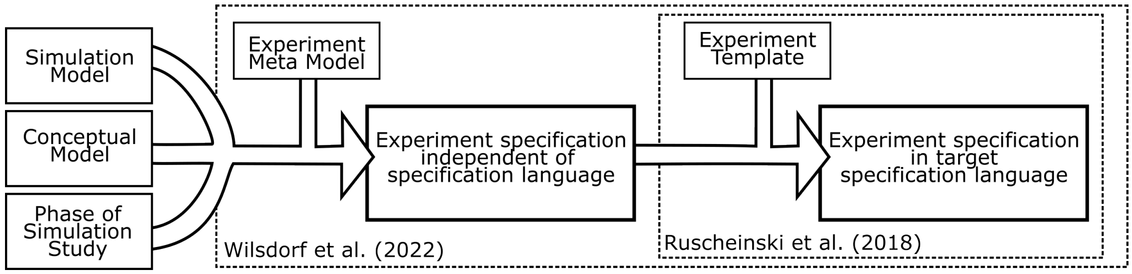

Figure 3: Structure of our experiment generator. Based on a simulation model, the simulation study's conceptual model and phase, our approach uses Wilsdorf et al. (2023)'s experiment meta-models to generate an experiment specification independent of a specification language. That specification is then translated into a target specification language using Ruscheinski et al. (2018)'s approach to generate experiment specifications using experiment templates.

research question has already been answered, allowing the modeler to skip the answering research question phase altogether.

4. Presentation Phase In the *presentation* phase, the goal is to communicate findings effectively to stakeholders. Thus, experiments are designed to reproduce specific trajectories from previous experiments that are used to convey the respective results, e.g., via dedicated reports, infographics, and storytelling techniques. Primarily, the results of the third phase (referring to the research questions) have to be presented in an accessible manner to the stakeholders (or for publication). As such, experiments in the presentation phase select subsets of model configurations of experiments in the third phase. These experiments only generate the trajectories used in the respective communication form and are thus clear and easy to reproduce. However, for communication purposes, other experiments (particularly of the confirmative phase) and their results may also be presented in a way that puts the final results into context (Uhrmacher et al. 2024).

4 USING EXPERIMENT GOALS FOR EXPERIMENT GENERATION

Based on the goals and associated experiments, the simulation model, and context information from the conceptual model, we automatically generate simulation experiments. The current goal defines what types of experiments are useful to the modeler in their current phase (see Figure 2). The next step is to identify what information regarding the model's inputs and outputs is needed to generate the respective experiment. We use a meta-model-driven approach for simulation experiments (Wilsdorf et al. 2022; Wilsdorf et al. 2019) (as shown in Figure 3). The meta-models define properties needed to generate experiments of various types (e.g., sensitivity analysis) and for various simulation approaches (e.g., stochastic discrete-event simulations). The combination of experiment type and simulation approach determines what properties are required (or optional) to generate an experiment. We then collect the information to fill the properties required by the experiments' meta-models from the simulation study's conceptual model, resulting in an experiment formulated independently of concrete specification languages. Finally, we use this language-independent experiment specification to generate an executable experiment in a target specification language using Ruscheinski et al. (2018)'s template-based approach.

In this section, we will explain how the current goal, the conceptual model, and the simulation model are used to collect and infer the information needed to generate a programming language-independent experiment specification as defined by meta-models. We first go through the conceptual model's elements that we use for the generation of simulation experiments and explain what information we expect to be part of the elements based on the definition provided by Robinson (2008) and a first formal data model specified by Wilsdorf et al. (2020). Similar contents of the conceptual model have also been discussed to be a central ingredient of documentation standards, such as TRACE (Grimm et al. 2014).

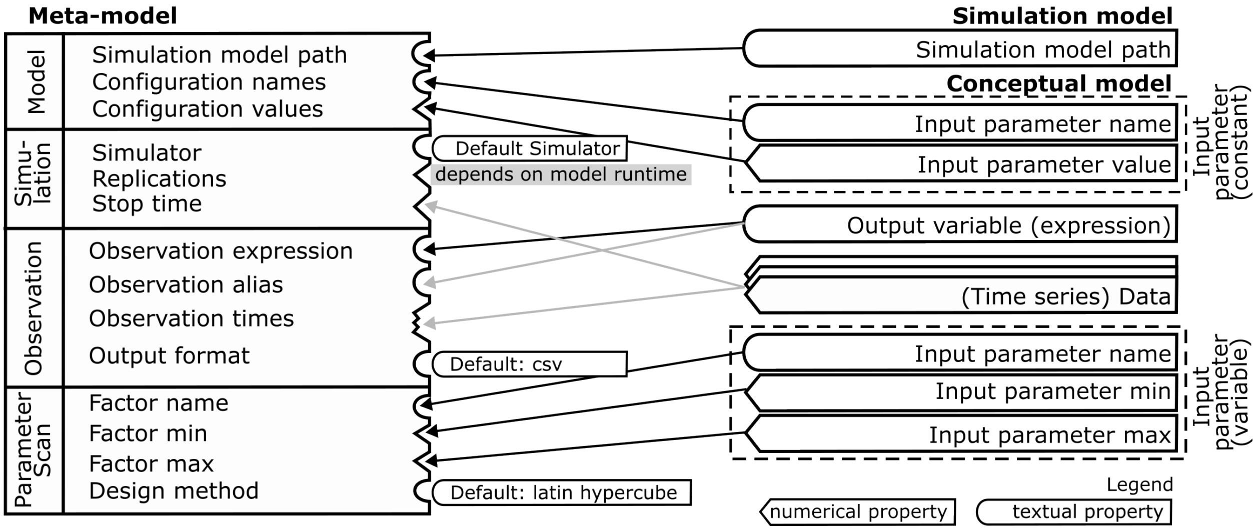

Figure 4: We use information from the conceptual model and simulation model to fill the experiment properties required by the meta-model for the target experiment's type. In this figure, the experiment's type is a parameter scan with the goal of exploration. For other types, the section labeled "parameter scan" in the meta-model is exchanged. Properties linked by black arrows can be imported from the conceptual model directly, while information from the conceptual model with gray arrows is processed first.

4.1 Contents of the Conceptual Model

The *research question* defines the scope of the simulation model and what questions about the modeled system shall be answered by the simulation model. Usually, the research question is formulated as text in natural language.

Requirements pose conditions on the simulation model's output, inputs, or structure. Requirements for the simulation model's outputs require the outputs to fulfill certain constraints or to reproduce (time-series) data. Either way, the requirement must specify which model output it refers to and how the simulation model's input parameters are configured. This type of requirement is usually used to validate the simulation model. Alternatively, requirements can define that a simulation model should be calibrated by providing a data set or logical formula (as long as a distance metric is available (Palaniappan et al. 2013)), the output variable that should fit the data, and the input parameter(s) to be calibrated. Finally, a requirement can define the modeling and simulation approach or other technical aspects of the simulation study (Balci 2012).

Data is central to simulation studies as it defines parameter values or the initial model state, is used to calibrate input parameters, and is used to validate the simulation model. Crucially, a data element in the conceptual model must contain a reference to the data source (file, database, etc.). In addition, the format of the data source and a tool suggesting how to read the data source can help to process the data automatically.

Output variables define what outputs of the simulation model and expressions thereof are relevant to the simulation study. The output variables can be used for calibration, validation or analysis, including answering the research question.

Input parameters may be defined by specific values, value ranges, or even random distributions. Some input parameters may also be unknown and subject to calibration as part of the simulation study.

4.2 Collecting Information from the Conceptual Model and Simulation Model

Our approach generates experiments containing all the properties required by the meta-models for stochastic discrete-event simulation using information extracted from both the conceptual model and the simulation model (see Figure 4). Most properties can be directly imported from the conceptual model as depicted in the figure. For example, the model's configuration names and values can be directly taken from the input parameter names and the corresponding values in the conceptual model.

For some experiment properties, we chose default values that work for most applications. So, we always use a default simulator for the respective model specification language, always use the output format CSV, and perform parameter scans using a Latin hypercube design.

Some information from the conceptual model has to be processed before being used to fill the experiment properties. For instance, the experiment specification has to define at which points in the simulation the observation expression(s) shall be evaluated and when the simulation should stop. We fill these properties using time series data from the conceptual model (that is used for calibration or validation). As we assume that this data captures the time frame relevant to the simulated process, the time points specified in the data can be singled out from the data as observation times for the simulation. Similarly, we set the simulation stop time to match the highest timestamp found in any time-series data within the conceptual model. Also, the observation alias for the observed expression is generated from the expression by removing all special characters and numerals from it to obtain a viable variable name.

Figure 4 shows an example of a parameter scan used for exploration. Thus, all model inputs and outputs are used in the experiment generation, and the number of replications depends on the model runtime (see Section 5 for a more detailed explanation). Other types of experiments with different goals use the same principle of importing information from the conceptual model. The main difference is that experiments with the goal of confirmation focus on the requirements defined as part of the conceptual model. As explained above, requirements for calibration or validation (which the confirmation phase focuses on) specify the model's input configuration, output variables, further constraints, and, in particular, the data (or logical formula) to fit to or compare with.

5 APPLICATION

As a proof of concept, we implemented a prototype of our approach, which can be accessed in this git repository (Wolpers et al. 2025). In the following, we will briefly describe how the prototype generates a parameter scan in the exploration phase for a simulation model of the WNT signaling pathway (Haack et al. 2015) specified in the rule-based multilevel modeling language ML-Rules (Helms et al. 2017). After that, we will more generally put our approach into the context of experiment generation.

5.1 Generating a Parameter Scan for the Exploration Phase

The WNT signaling pathway model represents the dynamics of key regulatory elements of the pathway, such as WNT, LRP6, Axin, and Beta-Catenin. It combines regulatory processes at the membrane with those in the cytosol and the nucleus. As part of the model's construction, a membrane sub-model simulated the distribution of specific proteins (LRP6 and CK1γ) within or outside so-called lipid rafts domains.

The first experiments generated by our tool are always designed to support exploration. Based on the simulation model (an ML-Rules file) and the simulation study's conceptual model (a JSON file), it automatically generated a parameter scan. In addition to multiple model configurations, multiple replications need to be calculated as the model is stochastic. As explorative simulation experiments should give fast feedback to the user, our tool internally measured the execution time of one replication and, based on this (and some default) information, determined the number of replications and the number of model configurations to be calculated. Automatically, all available processing cores, but one, are used. Runtime information is also important in selecting methods for exploration as well as confirming specific requirements, e.g., to decide whether variance-based and density-based methods can be used for sensitivity analysis (or will be too computationally challenging (Pianosi et al. 2016)).

Figure 5 shows how, based on the different types of information in the conceptual model, our tool generates the simulation experiment to support the exploration of the model.

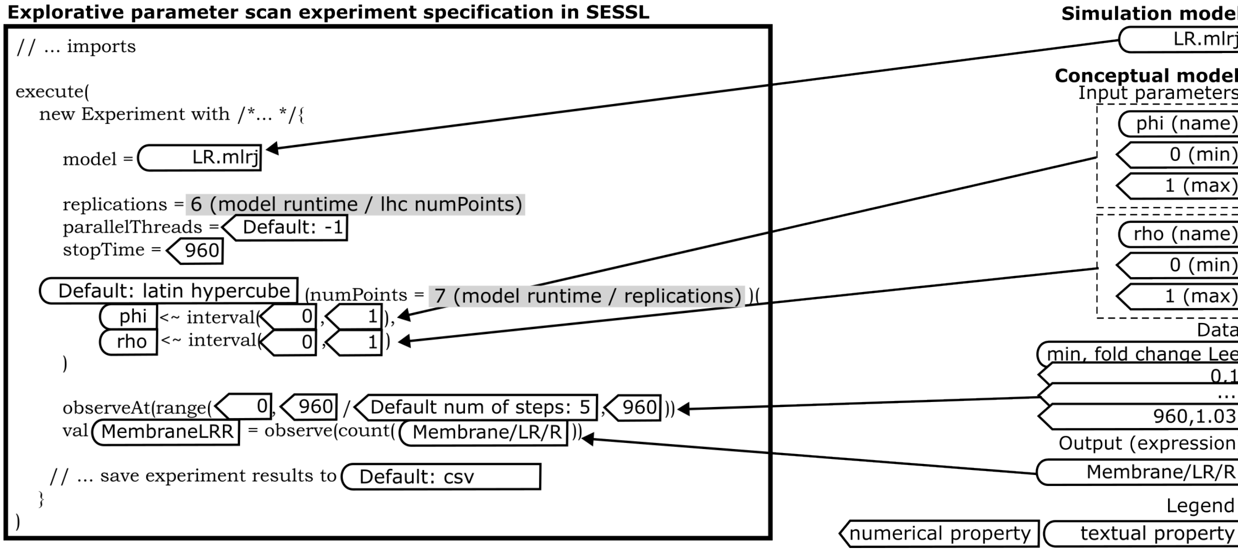

Figure 5: How our tool generated a parameter scan experiment specification in the exploration phase in SESSL by importing information from the conceptual model.

5.2 Generating Simulation Experiments for the Different Goals

The generation of simulation experiments for the different goals typically follows a pattern similar to related experiment generation approaches: The key challenge lies in identifying and integrating suitable sources of information. In this regard, the conceptual model can play a central role by providing tables of parameters (with their ranges, meaning, and names in the simulation model), logic formulas as requirements, etc.

Exploration Generally, for exploration, information about parameter values and ranges is important, e.g., for conducting parameter scan experiments and sensitivity analyses. In Ruscheinski et al. (2018), input parameter tables were used to generate sensitivity analysis experiments. Once generated and executed, these experiments might be reused to explore the behavior of refined simulation models (Wilsdorf et al. 2023). For exploration, visualizations will be central. Therefore, spontaneous simulation experiments, i.e., toy duck experiments in which you simply "wind it up and let it run" (Uhrmacher 2012) can be generated and executed, illustrating the behavior of relevant output variables. Also, for exploration, it would be interesting to explore specific underlying assumptions of the simulation model. For example, if we assume that a specific cognitive theory can be applied (Haase et al. 2022), what would it mean for the simulation results if another cognitive theory is used? So experiments with the goal of exploration could support exploratory modeling, where not one single model but a set of alternative simulation models is explored. To automate this process would imply transforming a simulation model based on one theory to another cognitive theory.

Confirmation Confirmation experiments require testable hypotheses. In addition to the generation of parameter scans and the comparison to *data* as explained in Section 3, Yilmaz et al. (2016) and Lorig (2019) formally specified *hypotheses* to generate model-checking experiments. The hypotheses may describe mechanisms, input-output relations, or constraints in a logic-based domain-specific language (Yilmaz et al. 2016) or as stylized facts (Wilsdorf et al. 2023). Ruscheinski et al. (2018) have used metric-interval temporal logic formulas to determine if a model behaves similarly to experimental wet-lab results (e.g., if a peak occurs in a specific time interval). This approach may be successively applied for re-validation after each model extension (Peng et al. 2016). But not only measured data, but also previous simulation results produced with a related simulation model may be used for generating (cross-)validations (Cooper et al. 2016). These, once an initial specification exists, may again may be generated repeatedly when new model versions become available (Wilsdorf et al. 2023). Our approach can contribute by integrating these methods for confirmation and automatically checking when which method applies to a given simulation study based on the study's context information and the methods' required information. We also aim to integrate a variety of calibration methods into the list of possible suggestions, which so far have not been in the focus of experiment generation.

Answering Research Question Some of the confirmatory experiments may be considered as part of the answering research question phase, depending on how close the hypotheses are to the original research question. Typically, a set of hypotheses will need to be evaluated to answer the research question (Yilmaz et al. 2016). Additionally, if the conceptual model explicitly and formally specifies target functions, such as "Maximise the storage capacity" in a manufacturing application (Lattner et al. 2011), parameter scans or optimization experiments may automatically be generated to obtain the answer to the overall research question. If the research question, like in most cases, is specified in natural language, new approaches for natural language processing (Hadi et al. 2023) may be exploited to extract suitable what-if scenarios.

Presentation So far, experiment generation has not focused on the presentation of simulation studies' results. Experiment generation for this phase will work similarly to the example shown above, and values of the conceptual model will need to be mapped to the properties required by experiment meta-models, following the goal and purpose of the respective experiments. A new challenge in this phase is the selection of which trajectories are of interest to present a study's results. Here, for being included in a publication, providing part of the caption of a figure might suffice for generating a simulation experiment and outputs fitting the caption. Additionally, a new aspect in the presentation phase will be the integration of existing knowledge and tooling from visual analytics and conceptualizing how to intertwine visualization (i.e, typically a postprocessing step with separate scripts) with experiment design.

6 DISCUSSION AND FUTURE RESEARCH

We extended existing approaches for (semi-)automatic experiment generation (Ruscheinski et al. 2018; Wilsdorf et al. 2022) with a goal-oriented mechanism. Four central goals of a simulation study were identified and used to steer the suggestion and generation of executable simulation experiments. The experiment specifications are generated from scratch with additional context information provided by the conceptual model. Our case study, in which we generated an exploration experiment, demonstrated the feasibility of this approach. We successfully extracted parameter values and ranges from the conceptual model, determined observation times and simulation stop times from the associated data, and generated a complete experiment specification that was executable without any additional user input. Further, we discussed how the application of our goal-oriented mechanism within existing experiment generators may support the different phases of a simulation study.

Beyond automation, our approach can effortlessly be expanded to automatically record provenance and thus support study documentation and reproducibility (Ruscheinski et al. 2019). Since all experiment executions, along with their corresponding goals and roles, are explicitly known to the tool, provenance recording can be seamlessly integrated.

Nevertheless, our automation requires users to define necessary information within a conceptual model that adheres to a well-defined structure with unambiguous semantics. While this may introduce initial effort, we argue that keeping explicit and well-structured conceptual models pays off by making the simulation workflow more efficient. Previously, the practical benefits of explicit conceptual models for improving a simulation study's documentation, understandability, reproducibility, and reusability have been highlighted (Wilsdorf et al. 2020; Robinson 2008; Uhrmacher et al. 2024). Thus, modelers are already encouraged to document their conceptual models for these reasons. Our only additional requirement is to structure this documentation in a machine-readable format (Wilsdorf et al. 2020). However, the new developments in large language models and their applications to text summarization, translation, and information retrieval (Hadi et al. 2023) may enable us to handle less rigid formats in the future.

In addition, we plan to integrate support for visualizing the results of simulation experiments into our tool, as our interviews have shown that visualizations are central to simulation studies. Modelers emphasized that their workflow involves iteratively designing experiments based on the visualized results of previous runs while refining their models during the exploration and confirmation phases. Allowing users to interact with visualized results and modify the generated experiments accordingly would support the modelers' workflows even more naturally.

Currently, our concept has been developed and tested primarily for rule-based stochastic discrete-event simulations and three specific application domains, with a limited number of participants. Thus, we can only be confident that similar simulation studies fit our classifications – even though we assume that they can be applied to most other simulation studies analogously. Future research could broaden the scope by interviewing more modelers from diverse research domains with different levels of expertise and exploring other modeling paradigms to assess and improve the generalizability of our approach. Particularly, we expect the definition of goals and the mapping of experiment types to these goals to evolve as we conduct more extensive interview studies. For instance, additional experiment types may arise during the discussions (e.g., using Bayesian methods for parameter estimation and uncertainty quantification (Mitra and Hlavacek 2019)). Moreover, based on a larger empirical data set, it may become possible to quantify the occurrence of different experiment types across the various phases of a simulation study, which may allow us to provide more targeted support for modelers.

ACKNOWLEDGMENTS

This research was funded by the German Research Foundation (DFG), grant no. 320435134.

REFERENCES

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

Cooper, J., M. Scharm, and G. R. Mirams. 2016. "The Cardiac Electrophysiology Web Lab". *Biophysical Journal* 110(2):292–300.

Grimm, V., J. Augusiak, A. Focks, B. M. Frank, F. Gabsi, A. S. A. Johnston, *et al*. 2014. "Towards Better Modelling and Decision Support: Documenting Model Development, Testing, and Analysis Using TRACE". *Ecological Modelling* 280:129–139.

Haack, F., K. Budde, and A. M. Uhrmacher. 2020. "Exploring the Mechanistic and Temporal Regulation of LRP6 Endocytosis in Canonical WNT Signaling". *Journal of Cell Science* 133(15).

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):e1004106.

Haase, K., H. V. Strehlow, W.-C. Lewin, O. Reinhardt, and A. M. Uhrmacher. 2022. "Site Choice in Recreational Fisheries - Towards an Agent-Based Approach". In *2022 Winter Simulation Conference (WSC)*, 819–830 https://doi.org/10.1109/WSC57314.2022.10015240.

Hadi, M. U., R. Qureshi, A. Shah, M. Irfan, A. Zafar, M. B. Shaikh, *et al*. 2023. "Large language models: a comprehensive survey of its applications, challenges, limitations, and future prospects". *Authorea Preprints* 1:1–26 https://doi.org/10.36227/techrxiv.23589741.v8.

Helms, T., T. Warnke, C. Maus, and A. M. Uhrmacher. 2017. "Semantics and Efficient Simulation Algorithms of an Expressive Multilevel Modeling Language". *ACM Transactions on Modeling and Computer Simulation* 27(2):8:1–8:25.

Lattner, A., H. Pitsch, I. Timm, S. Spieckermann, and S. Wenzel. 2011. "AssistSim - Towards Automation of Simulation Studies in Logistics". *Simulation Notes Europe* 21(3-4):119–128.

Leye, S., J. Himmelspach, and A. M. Uhrmacher. 2009. "A Discussion on Experimental Model Validation". In *2009 11th International Conference on Computer Modelling and Simulation*. March 25th-27th, Cambridge, UK, 161-167.

Lorig, F. 2019. "Hypothesis-Driven Simulation Studies". In *Hypothesis-Driven Simulation Studies: Assistance for the Systematic Design and Conducting of Computer Simulation Experiments*, edited by F. Lorig, 137–170. Wiesbaden: Springer Fachmedien.

Mitra, E. D., and W. S. Hlavacek. 2019. "Parameter Estimation and Uncertainty Quantification for Systems Biology Models". *Current Opinion in Systems Biology* 18:9–18.

Palaniappan, S. K., B. M. Gyori, B. Liu, D. Hsu, and P. S. Thiagarajan. 2013. "Statistical Model Checking Based Calibration and Analysis of Bio-pathway Models". In *Computational Methods in Systems Biology*, edited by A. Gupta and T. A. Henzinger, 120–134. Berlin, Heidelberg: Springer.

Peng, D., T. Warnke, F. Haack, and A. M. Uhrmacher. 2016. "Reusing Simulation Experiment Specifications to Support Developing Models by Successive Extension". *Simulation Modelling Practice and Theory* 68:33–53.

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

Robinson, S. 2008. "Conceptual Modelling for Simulation Part I: Definition and Requirements". *Journal of the Operational Research Society* 59(3):278–290.

Ruscheinski, A., K. Budde, T. Warnke, P. Wilsdorf, B. C. Hiller, M. Dombrowsky *et al*. 2018. "Generating Simulation Experiments Based on Model Documentations and Templates". In *2018 Winter Simulation Conference (WSC)*, 715–726 https://doi.org/10.1109/WSC.2018.8632515.

Ruscheinski, A., P. Wilsdorf, M. Dombrowsky, and A. M. Uhrmacher. 2019. "Capturing and Reporting Provenance Information of Simulation Studies Based on an Artifact-Based Workflow Approach". In *Proceedings of the 2019 ACM SIGSIM Conference on Principles of Advanced Discrete Simulation*. June 3th-5th, Chicago, IL, USA, 185-196.

Sargent, R. G. 2010. "Verification and validation of simulation models". In *2010 Winter Simulation Conference (WSC)*, 166–183 https://doi.org/10.1109/WSC.2010.5679166.

Schulz, H.-J., T. Nocke, M. Heitzler, and H. Schumann. 2013. "A Design Space of Visualization Tasks". *IEEE Transactions on Visualization and Computer Graphics* 19(12):2366–2375.

Uhrmacher, A. M. 2012. "Seven pitfalls in modeling and simulation research". In *2012 Winter Simulation Conference (WSC)*, 1–12 https://doi.org/10.1109/WSC.2012.6465321.

Uhrmacher, A. M., P. Frazier, R. Hähnle, F. Klügl, F. Lorig, B. Ludäscher, *et al*. 2024. "Context, Composition, Automation, and Communication: The C2AC Roadmap for Modeling and Simulation". *ACM Trans. Model. Comput. Simul.* 34(4):23:1–23:51.

Wilsdorf, P., M. Dombrowsky, A. M. Uhrmacher, J. Zimmermann, and U. v. Rienen. 2019. "Simulation Experiment Schemas –Beyond Tools and Simulation Approaches". In *2019 Winter Simulation Conference (WSC)*, 2783–2794 https://doi.org/10.1109/WSC40007.2019.9004710.

Wilsdorf, P., F. Haack, and A. M. Uhrmacher. 2020. "Conceptual Models in Simulation Studies: Making it Explicit". In *2020 Winter Simulation Conference (WSC)*, 2353–2364 https://doi.org/10.1109/WSC48552.2020.9383984.

Wilsdorf, P., J. Heller, K. Budde, J. Zimmermann, T. Warnke, C. Haubelt, *et al*. 2022. "A Model-Driven Approach for Conducting Simulation Experiments". *Applied Sciences* 12(16):7977.

Wilsdorf, P., A. Wolpers, J. Hilton, F. Haack, and A. Uhrmacher. 2023. "Automatic Reuse, Adaption, and Execution of Simulation Experiments via Provenance Patterns". *ACM Transactions on Modeling and Computer Simulation* 33(1-2):4:1–4:27.

Wilsdorf, P., M. Zuska, P. Andelfinger, A. M. Uhrmacher, and F. Peters. 2023. "Validation Without Data - Formalizing Stylized Facts Of Time Series". In *2023 Winter Simulation Conference (WSC)*, 2674–2685 https://doi.org/10.1109/WSC60868.2023.10408388.

Wolpers, A., P. Wilsdorf, F. Haack, and U. Adelinde M. 2025. "The Git Repository Containing a Prototype of our Approach". https://git.informatik.uni-rostock.de/mosi/GREASE/experimentgenerationfromcm, Accessed 11th August 2025.

Yilmaz, L., S. Chakladar, and K. Doud. 2016. "The Goal-hypothesis-experiment Framework: A Generative Cognitive Domain Architecture for Simulation Experiment Management". In *2016 Winter Simulation Conference (WSC)*, 1001–1012 https://doi.org/10.1109/WSC.2016.78221600.

AUTHOR BIOGRAPHIES

ANJA WOLPERS is a Ph.D. candidate in the Modeling and Simulation group at the University of Rostock. Her e-mail address is anja.wolpers@uni-rostock.de.

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

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

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