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.

DIALECTIC MODELS FOR DOCUMENTING AND CONDUCTING SIMULATION STUDIES: EXPLORING FEASIBILITY

Steffen Zschaler[1], Pia Wilsdorf[2], Thomas Godfrey[3], and Adelinde M. Uhrmacher[2]

[1]Dept. of Informatics, King's College London, London, UK
[2]Institute for Visual and Analytic Computing, University of Rostock, Rostock, GERMANY
[3]Aerogility Ltd, London, UK

ABSTRACT

Validation and documentation of rationale are central to simulation studies. Most current approaches focus only on individual simulation artifacts—most typically simulation models—and their validity rather than their contribution to the overall simulation study. Approaches that aim to validate simulation studies as a whole either impose structured processes with the implicit assumption that this will ensure validity, or they rely on capturing provenance and rationale, most commonly in natural language, following accepted documentation guidelines. Inspired by dialectic approaches for developing mathematical proofs, we explore the feasibility of capturing validity and rationale information as a study unfolds through agent dialogs that also generate the overall simulation-study argument. We introduce a formal framework, an initial catalog of possible interactions, and a proof-of-concept tool to capture such information about a simulation study. We illustrate the ideas in the context of a cell biological simulation study.

1 INTRODUCTION

Given the increasing importance of simulation (Winsberg 2019) and its complexity, ensuring the credibility, reproducibility, and quality of simulation studies has been the subject of various research efforts over the last decades. For example, research has addressed the identification of crucial steps of simulation studies—in the form of modeling and simulation life cycles (Sargent 2010; Balci 2012; Robinson 2008)—supporting simulation studies by workflow systems (Al-Zoubi and Wainer 2010; Ruscheinski et al. 2020), and, last but not least, documentation guidelines and (semi-)automatic support for conducting simulation studies (Monks et al. 2019; Grimm et al. 2014; Ruscheinski and Uhrmacher 2017; Wilsdorf et al. 2024). Each of these approaches provides a specific view of simulation studies.

Scientific research rarely follows a linear progression (Rule and John 2015). Thus, capturing the reasoning (arguments) behind the steps and decisions made is crucial. In argumentation-based dialogs, participants (e.g., two colleagues in a simulation study) engage in an exchange of arguments, using mechanisms of argumentation to establish what facts/knowledge can be deemed acceptable (Black et al. 2021). Beyond dialogs, monologic discourses *conducted in a dialogic* way can also improve understanding (Ford and Wargo 2012). We consider this and view simulation studies through the lens of dialogs—to be more precise, argumentation-based dialogs rooted in dialectics (van Emeren et al. 1987, Page 63*ff.*).

Inspired by argumentation-based approaches for mathematical proofs (Pease et al. 2017; Corneli et al. 2019; Almpani et al. 2023; Almpani and Stefaneas 2023), we explore the feasibility of a dialectic approach to document and conduct simulation studies. We develop a formal dialectic framework where a "proponent" proposes artifacts of a simulation study and supports them (e.g., with theories or experiments), while an "opponent" critically evaluates and challenges their validity. We make the following contributions:

1. We propose a dialectic model based on an exchange of arguments and counter-arguments, providing an approach to the fundamental challenge of developing scientific understanding through the

979-8-3315-8726-0/25/$31.00 ©2025 IEEE

interactively evolving comprehension of meanings and implications (Khalifa 2017). This model consists of:

(a) An agent dialog protocol (McBurney and Parsons 2009) defining an initial catalog of *moves* that proponents and opponents may make as part of a simulation study;
(b) Rules describing how each move extends an *argument graph* describing the current state of the simulation study; and
(c) Analysis of argument graphs based on *abstract argumentation theory* to identify the set of arguments that are currently agreed by both proponent and opponent.

2. We provide a proof-of-concept implementation of the approach.
3. We demonstrate our approach using a biochemical simulation study.

2 BACKGROUND

In this section, we provide some background necessary for understanding the remainder of the paper. Specifically, we first discuss some key concepts from the field of agent dialog protocols and then some key ideas from the field of abstract argumentation as our approach combines concepts from these fields. We then briefly highlight some technical concepts from the field of model-driven engineering, which we have used in our proof-of-concept implementation.

2.1 Dialog protocols for agent argumentation

Inspired by Pease et al. (2017), we model simulation studies as a dialog between proposing and opposing agents, who exchange arguments with the purpose of negotiation, persuasion, and inquiry (Wooldridge 2012). McBurney and Parsons (2009) define five types of concepts and rules to govern agent argumentation:

1. *Locutions* or *moves* – that is statements that can be made by the agents involved;
2. *Commencement rules* defining the conditions under which dialogs can start;
3. *Combination rules* defining which moves are allowed in which dialog states;
4. *Commitments* defining what knowledge participants accept at any given point in a dialog; and
5. *Termination rules* indicating the circumstances under which a dialog may end.

The purpose of scrutinizing individual artifacts and activities of a simulation study by arguing is most closely related to the concept of persuasion, in particular, its conflict resolution subtype. This type of dialog is triggered by a difference in opinions, with the main goal of the conversation to resolve the conflict (Pease et al. 2017). In this paper, we will focus on describing specific moves and commitments, where the latter will be captured by monotonically extending an argument graph – thus giving the semantics of the moves. We, therefore, briefly introduce argument graphs next.

2.2 Abstract argumentation

In his seminal essays on the uses of argument, Toulmin discusses ‘field-independent’ and ‘field-specific’ (we would perhaps refer to these as ‘domain-independent’ (or ‘generic’) and ‘domain-specific’) properties of arguments constructed by humans across a wide spectrum of dialectic interactions. He identifies ‘field-independent’ layouts of arguments—essentially a template structure for arguments regardless of the domain of debate. In this paper, we build on the basic structure he identifies (Toulmin 2003, p. 94):

$$\begin{array}{ccc} \mathrm{D} & \longrightarrow & \textbf{So } \mathrm{Q, C} \\ & | & | \\ & \textbf{Since } \mathrm{W} & \textbf{Unless } \mathrm{R} \end{array}$$

where ‘D’ refers to some data (or evidence) allowing us to make a claim ‘C’. We can make this inference because we know and accept a warrant ‘W’ (a—possibly ‘field-specific’—rule of reasoning). Even so, we may wish to add a qualification 'Q' (e.g., 'presumably') and list potential rebuttal conditions 'R' (exceptions to the rule established by 'W'—cf. the critical questions in Walton (2013)'s work on presumptive reasoning).

Building on Toulmin's work, the field of abstract argumentation (Dung 1995) studies *abstract argumentation frameworks*: graphs of *abstract arguments* and *attack* relationships between them (more complex forms of argumentation frameworks (Modgil 2013) are out of the scope of this paper). Arguments are called 'abstract', because the formalisation is not interested in the specific contents or meaning of individual 'arguments', but only in their relationships. This formalisation enables automated analysis, identifying so-called *extensions*—subsets of arguments that can rationally be held to be true together (these arguments are called *acceptable*). Dung (1995) defines different types of extensions. In this paper, we use *complete* extensions to analyse argumentation frameworks. A subset S of the arguments in an argumentation framework is a complete extension if and only if every argument in S is acceptable with respect to S (there are no attacking arguments in S, or those arguments are themselves being attacked by an argument in S) and every argument that would be acceptable with respect to S is also included in S. Several software tools have been developed to automatically derive all complete extensions of a given abstract argumentation framework. We will use `plato` (the latest development of EQARGSOLVER (Rodrigues 2018)).

2.3 Model-driven engineering

We briefly review some concepts and technologies from model-driven engineering (MDE) (Brambilla et al. 2017), which we use to simplify the formal encoding of our ideas and in the development of a proof-of-concept prototype. MDE uses models expressed in formal languages, called modelling languages, to capture information about software systems under development from various stakeholders. In this paper, we will develop several domain-specific modelling languages (DSMLs)—modelling languages based on concepts specific to a particular problem domain—to capture key ideas. Language workbenches (Erdweg et al. 2015) are software development tools that simplify the creation of DSMLs by automating many processes based on declarative descriptions of the language's syntax and descriptions of its semantics. Language workbenches differentiate between the *concrete syntax* of a language—what the language feels like to the user—and *abstract syntax*—what concepts are available and how they can be related to each other. We will use Xtext (Eysholdt and Behrens 2010), which supports textual concrete syntax, and Sirius (Viyović et al. 2014), which supports graphical concrete syntax. The abstract syntax of our languages will be provided as a metamodel (essentially a class diagram) using the Eclipse Modelling Framework (Steinberg et al. 2009).

Finally, model-to-model transformations are used to translate information between DSMLs. We will use Henshin (Strüber et al. 2017), a graph-transformation-based model-to-model transformation tool, to describe how individual moves change an overlying argument graph for the simulation study. Individual Henshin rules describe a change to a model based on matching and preserving existing elements, creating new elements, and removing existing elements. They are written in a graphical format, directly referring to elements from the modelling language's abstract syntax.

3 AN INITIAL DIALECTIC MODEL OF SIMULATION STUDIES

Our goal is to explore the feasibility of a dialectic model of simulation studies. Hence, we focus on describing an initial set of locutions and corresponding commitments – expressed as changes to an argument graph in the spirit of Pease et al. (2017). We capture basic combination rules through parameter references in locutions, requiring that all objects referenced must have been explicitly introduced by an earlier locution. Other combination rules are conceivable but outside the scope of our initial feasibility study.

Equally, commencement and termination rules are not defined explicitly at this stage. We would naturally expect simulation studies to start with the proposal of a research question (`proposeResearchQuestion` move). Different termination rules can be considered depending on the degree of finality we expect. In line with thinking about proofs as socio-temporal sequences of 'proof events' in mathematics (Almpani and Stefaneas 2023), we can consider simulation studies to reach temporary closure with the publication of a paper or report, but the overall argument to remain open to future challenges from other members of the community. In such a case, the dialog may temporarily be considered terminated only to be re-opened at a later stage when a new challenge is identified.

3.1 Moves

Table 1 lists an initial set of moves that can be used to describe the process of a simulation study. Moves are described using a signature: a name and a list of parameters, where some parameters indicate data to be provided as part of the move and other parameters indicate references to data provided in previous moves. The table indicates whether a move is meant to be made by a proponent (P) or opponent (O) agent – generally, proponents make moves that claim validity of a particular simulation-study artifact while opponents aim to challenge the validity of an artifact (cf. Wilsdorf et al. (2024)).

3.2 Commitments

We use commitment rules to capture the semantics (McBurney and Parsons 2009) of the different moves that agents can make. Inspired by Pease et al. (2017), each move contributes elements to an *argument graph* representing what all parties agree to be true about the answer to the overall research question. No move removes any elements from the argument graph, but some moves will introduce elements that attack other existing elements. Argument graphs are not abstract argumentation frameworks: different to Dung (1995), we care about the specific contents of each statement in the overall argument. In fact, we aim to differentiate and catalog different types of statements that can be made.

Figure 1 shows the schema of such argument graphs expressed as a metamodel in standard class-diagram notation: each box represents a concept that can be instantiated in an argument graph, and edges represent relationships between the concepts. The left part of the diagram shows the structure of standard Toulmin (2003) arguments. Arguments are a relation (`ArgumentElementRelation`) between elements (`ArgumentElement`) playing the roles of claim, evidence, and warrant, respectively. This captures what Toulmin (2003), pp. 111ff. calls "warrant-using" and "warrant-establishing" arguments: an argument element that is the warrant in one argument relation may itself be the claim in a different relation. We also support Toulmin's notion of rebuttals, but prefer a positive framing as assumptions under which an overall argument can be considered valid. Different from Toulmin, we provide two types of relations to distinctly capture positive (`Support`) and negative (`Attack`) arguments. Finally, we add a list of argument element types (right half of Figure 1) that are specific to simulation studies. We lack the space to explain each of these in detail, so will pick only three examples:

1. A `SimulationMechanismWarrant` states that a simulation model that can be shown to replicate given `outputDataOverTime` (e.g., from a wet-lab experiment) may be considered to provide a mechanistic explanation of some real-world effect (described via the `explainedEffect` attribute). Proposing a requirement (`proposeRequirement` move) implicitly establishes such a warrant for a specific effect and data.
2. A `MechanismExplainsEffect` element states that a given `explainedEffect` can be explained by a given `mechanism`. This, thus, indicates we have found an answer to a research question of the form *'what mechanism best explains <effect>?'*
3. A `ModelMatchesDataOverTime` element claims that a given `model`, which implements a given `mechanism`, does indeed replicate some given `dataOverTime`. On its own, this is just a claim, which may or may not be justified. Proposing a model and claiming it satisfies a given requirement establishes a `ModelMatchesDataOverTime` element and uses it as evidence in

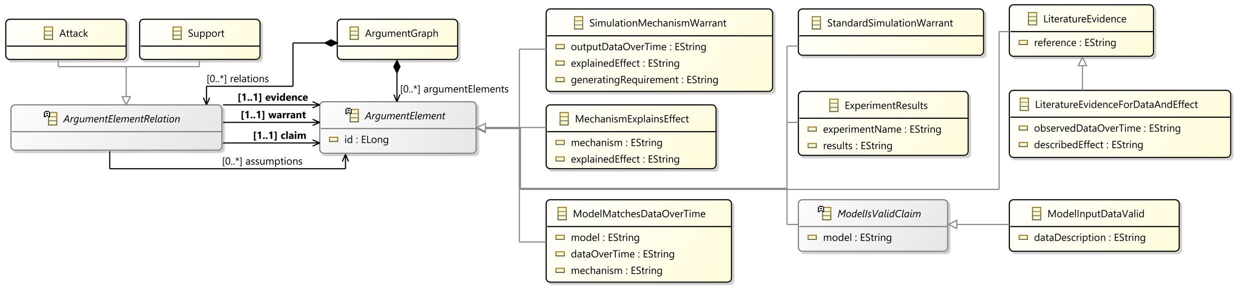


Figure 1: Argument-graph metamodel. The left half is a standard Toulmin argument structure. The right half introduces types of arguments specific to simulation studies.

Table 1: Initial set of moves for capturing parts of simulation studies. Arguments to evidence validity can use the argument patterns from Wilsdorf et al. (2024). Parameters in `[]` are references to objects established through earlier moves.

Move	Agent	Description
`proposeResearchQuestion (rq)`	P	Propose a specific research question for the simulation study. This lays the foundation for the central argument: the study needs to provide a contextualised answer to `rq`.
`proposeRequirement (req, [rq])`	P	Propose a requirement that a simulation model must satisfy to help answer the referenced research question `rq`.
`attackRequirement ([req], reason)`	O	Declare a requirement invalid.
`redefineRequirement ([req], newReq)`	P	Propose an alternative requirement formulation, typically in response to an attack on `req`.
`retractRequirement ([req])`	P	Retract a previously proposed requirement, typically in response to an attack on `req`.
`supportRequirement ([req], evidence)`	P	Provide (additional) support for the validity of `req`.
`reviseRequirement ([model], [req], newReq, experiment)`	P	Propose a refined requirement and claim that `model` satisfies it, supported by an experiment.
`proposeModel (model, [req])`	P	Propose a model, claiming that it satisfies requirement `req`.
`supportModel ([model], evidence)`	P	Provide evidence for the validity of `model`.
`counterModel ([model], experiment, [req])`	O	Provide a simulation experiment showing that `model` does not satisfy `req` as previously claimed.
`attackModel ([model], evidence)`	O	Provide a reason why `model` should not be considered well-constructed and valid.
`replaceModel ([model], newModel)`	P	Provide a new version of a model, usually in response to an attack or a counter to the original model.
`proposeExperiment ([model], experiment, [req])`	P	Support an earlier claim that `model` satisfies `req` by providing a simulation experiment.
`supportExperiment ([experiment], evidence)`	P	Provide evidence that `experiment` was well-designed, executed, or analysed.
`attackExperiment ([experiment], evidence)`	O	Provide evidence that `experiment` was *not* well-designed, executed, or analysed.
`retractExperiment ([experiment])`	P	Retract an experiment previously shown to be invalid.

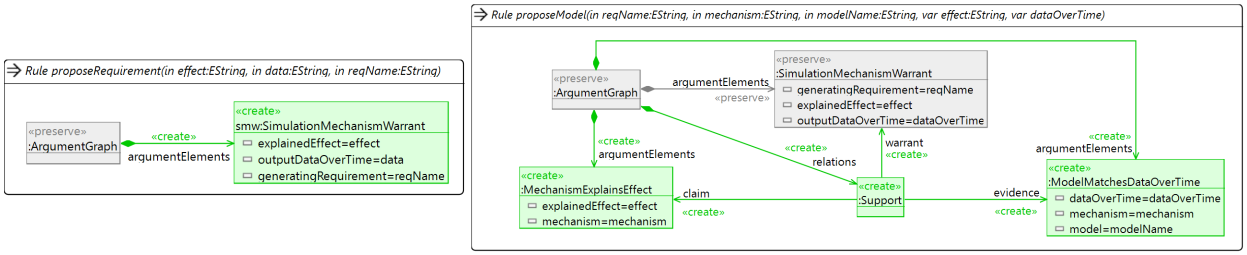


Figure 2: Example rules for commitments for `proposeRequirement` and `proposeModel` moves. The rules are expressed in Henshin (Strüber et al. 2017) syntax, where gray elements are expected to be found in a graph, and green elements will then be added to the graph.

an argument with a `MechanismExplainsEffect` claim. This argument is supported by a `SimulationMechanismWarrant` established when the requirement was originally proposed.

In our proof-of-concept implementation (cf. section 3.4), we represent the changes to the argument graph affected by each move using graph transformation rules expressed in Henshin (Strüber et al. 2017). Figure 2 shows the rules for proposing a requirement and proposing a model:

- Rule `proposeRequirement` adds a newly established `SimulationMechanismWarrant`.It assumes requirements of a particular form, namely indicating a particular real-world data set to be replicated by a model. This newly established `SimulationMechanismWarrant` says that a model that can replicate that data may be assumed to explain a particular `explainedEffect`. This effect is taken from the research question that the requirement has been proposed for (note that `proposeRequirement` references a specific research question.) This assumes research questions that focus on identifying mechanisms that may explain a particular real-world effect.
- Rule `proposeModel` adds argument elements implied by the corresponding move. Specifically, it uses the previously established `SimulationMechanismWarrant` to add a `Support` argument element relation between a newly established piece of evidence about the proposed model being able to match the given data (`ModelMatchesDataOverTime`), and a claim that the particular `mechanism` incorporated in the model can explain the effect we are aiming to explain overall (`MechanismExplainsEffect`), thereby offering an answer to the research question.

The above assumptions are of course not true for all simulation studies; developing a more complete taxonomy of the structure of possible research questions, requirements, and models is left for future work.

3.3 Analyzing the argument graph

We translate the argument graph into an abstract argumentation framework (Dung 1995) and use a standard argument solver (Rodrigues 2018) to identify the set of argument elements currently accepted by both proponent and opponent. In particular, it becomes possible to identify whether the study currently offers an acceptable answer to the overall research question.

Abstract argumentation frameworks, in their most basic form, only support attack relations between arguments, whereas our argument graphs can include both attack and support relations between argument elements. At its most basic, a support relation can be translated to a sequence of two attack relations with an intermediary artificial argument element, which abstracts possible attacks on the supported argument element. Thus, $A \overset{support}{\rightarrow} B$ in our argument graphs can be translated to $A \overset{attack}{\rightarrow} \epsilon_{AB} \overset{attack}{\rightarrow} B$ in an abstract argumentation framework, where ϵ_{AB} is an artificial, abstract argument newly introduced as part of the translation. We use this interpretation in our translation (cf. Section 4). As a consequence an

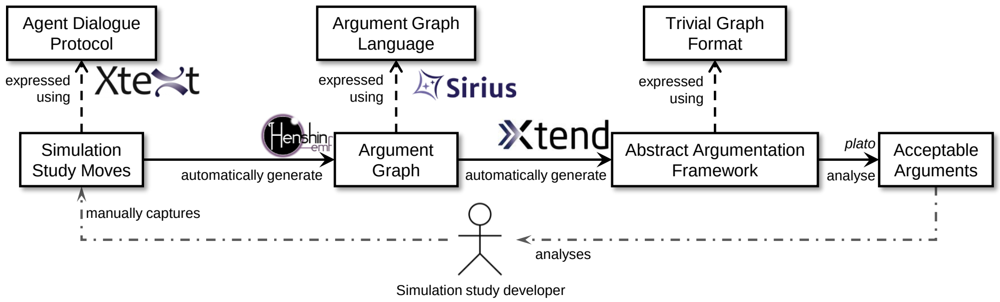


Figure 3: Tools and information flow in the proof-of-concept implementation.

attack on evidence, warrants, or assumptions in a support relation implicitly attacks the claim, too. Other interpretations are possible (Proietti et al. 2019), and we will explore these in future work.

3.4 Proof-of-Concept Implementation

To test our ideas, we have developed a proof-of-concept implementation, which is available open source on GitHub (https://github.com/cs4ec/ModelSpeak). A final tool will require better integration than this early prototype, perhaps in electronic lab notebooks (Higgins et al. 2022). Figure 3 gives an overview of the components and information flow:

1. Simulation study developers use a textual notation developed in Xtext (Eysholdt and Behrens 2010) to capture a sequence of moves describing activities that occurred as part of a simulation study.
2. From this sequence of moves, the software automatically generates an argument graph, using Henshin (Strüber et al. 2017) rules like the ones shown in Figure 2. We have implemented a simple argument-graph language using the Eclipse Modelling Framework (EMF) (Steinberg et al. 2009) for the abstract syntax (cf. Figure 1) and Sirius (Viyović et al. 2014) for a graphical concrete syntax.
3. The argument graph is then further automatically translated into an abstract argumentation framework. The transformation is currently fairly straightforward and has been implemented directly in Xtend (Bettini 2016). It produces an EMF-based model (for visualization purposes, we have also implemented a simple visual syntax with Sirius) and serializes this using the trivial graph format.
4. Finally, the abstract argumentation framework is processed by the `plato` argument solver (Rodrigues 2018) to identify the set of acceptable arguments. These can then be presented back to the simulation study developer for further analysis.

4 ILLUSTRATIVE CASE STUDY

In this section, we illustrate how the validity argument for an example simulation study can be captured using our approach. We use the study described by Haack et al. (2020), which explores different mechanisms of LRP6 receptor endocytosis and how they either promote or attenuate Wnt signalling. This study has previously been used in the evaluation of other validation and documentation approaches, such as provenance graphs (Budde et al. 2021; Wilsdorf et al. 2024), making it easier to access relevant information for our purposes. In the study, three successive model extensions and various experiments were necessary to build a simulation model that represents mechanisms of endocytosis of Wnt signalling, fitting to various in-vitro experiments. The model was then validated by using it to predict the dynamics of Wnt endocytosis induced by a different ligand. Information from references are used to constrain the structure and parameter space of the model and to compare the simulation outputs with.

Our case study focuses on the start of this simulation study: the first iteration of model building and the failed attempt at model output verification (Grimm et al. 2014). We are starting at a point where we no longer question the validity of the background knowledge given in other references. Figure 4 shows the moves made during this stage of the simulation study and the argument graph that is automatically generated from these moves. The proposal of a research question (`rq1`) and refinement to a requirement (`r1`) on the model (Lines 1–8) generates the warrant ①. The requirement is further underpinned by providing various literature references (`ref3`, `ref6`, and `ref7`, see Lines 10–12) that identify the wet-lab data that needs to be matched, generating the argument structure labeled ②. The simulation experiments try to identify a model with plausible parameter values that is able to reproduce the data given in the references. When comparing the argumentation-based dialogue with the provenance graph of the study, presented in Figure 6 of Haack et al. (2020), it becomes evident that the research question and requirement are not provided in the provenance graph. Instead, the three data sources (`ref3`, `ref6`, `ref7`) are directly used as input to the model analysis activity without further clarifying their roles, i.e., output verification. Continuing with the agent argumentation dialog, a first model `m1` is proposed, claiming that it satisfies the requirement (Lines 14–17). This generates the orange claim of an initial overall answer to the research question, supported by the argument labeled ③. Note that this argument makes use of the warrant previously established by accepting the requirement `r1`. At this point no experiments have been performed yet; the argument graph only represents the claims currently accepted by the team undertaking the simulation study. An experiment (`s1`) is then undertaken and turns out to show that the model does not satisfy the requirement; the experiment thus counters the model (move shown on Lines 19–25). This adds the argument ④, which attacks the claim that `m1` matches the wet-lab data. In the provenance graph, the attack is denoted less explicitly by a model analysis activity that produced an experiment specification and a "findings" entity, which contains simulation data from the experiment execution as well as a remark whether the output verification was successful. Overall, the provenance graph in Figure 6 of Haack et al. (2020) delineates sequences of steps with their inputs and outputs, whereas the argument graph in Figure 4 comprises detailed claims, evidence, and warrants about the artifacts, which enables an analysis of what statements the simulation developers currently accept about the system being simulated.

To undertake such analysis, we first automatically translate the argument graph in Figure 4 into the abstract argumentation framework shown in Figure 5(a). This makes it amenable to analysis with standard tools developed by the abstract-argumentation community, such as `plato` (Rodrigues 2018)—for example, determining the set of acceptable arguments in the argumentation framework. Analysis with `plato` produces exactly the set $\{14, 7, 4, 3, 2, 1, 0\}$ (indicated by green ticks in Figure 5(a)). Crucially, this excludes 5, the claim that we have found a mechanism explaining the wet-lab data. This correctly reflects the fact that we currently do not have a model that replicates the data, because Experiment `s1` attacks Model `m1`.

We can do this analysis at any point during the simulation study. For example, one move earlier (that is, when the `counterModel` move on Lines 19–25 has not been made yet), argument ④ would not be part of the argument graph yet and, we would, thus, obtain the abstract argumentation framework in Figure 5(b). Using `plato` to obtain the acceptable arguments produces $\{6, 5, 4, 3, 2, 1, 0\}$ (see, again, the green ticks in Figure 5(b)), reflecting the fact that at this point the simulation-study team still believe (albeit without explicit evidence) that they may have found a mechanism explaining the wet-lab data.

As we discuss at the start of Sect. 3, we do not define any explicit termination rules for our dialogs. This is because, similar to the ideas of 'proof-events' and 'fluents' (sequences of proof-events) from Almpani and Stefaneas (2023), we think of a simulation study as a socio-temporal phenomenon, where new moves may be introduced by different agents at different points in time. This is where argument graphs and their automated analysis can be particularly useful. For example, assume we have produced a complete sequence of moves for our study and have generated the corresponding argument graph. At this point, we publish the paper (Haack et al. 2020), where we make a claim about the mechanism underpinning LRP6 receptor internalization. This corresponds approximately to Almpani's notion of a 'proof-event': a public

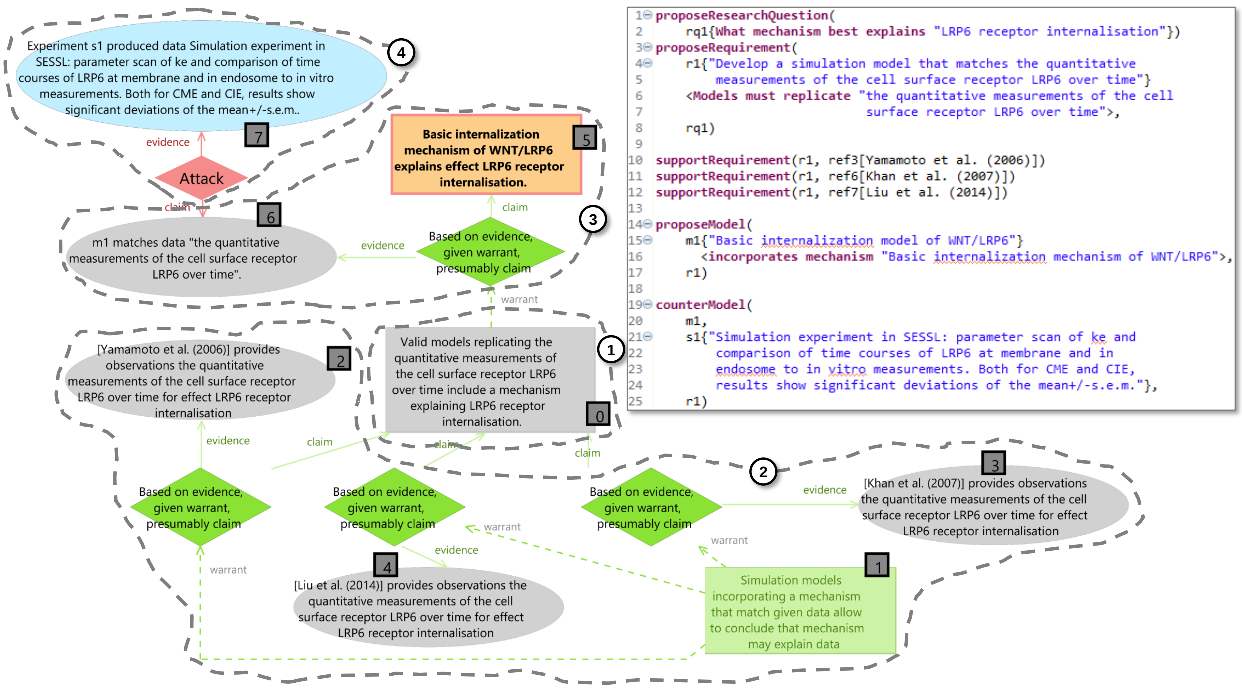
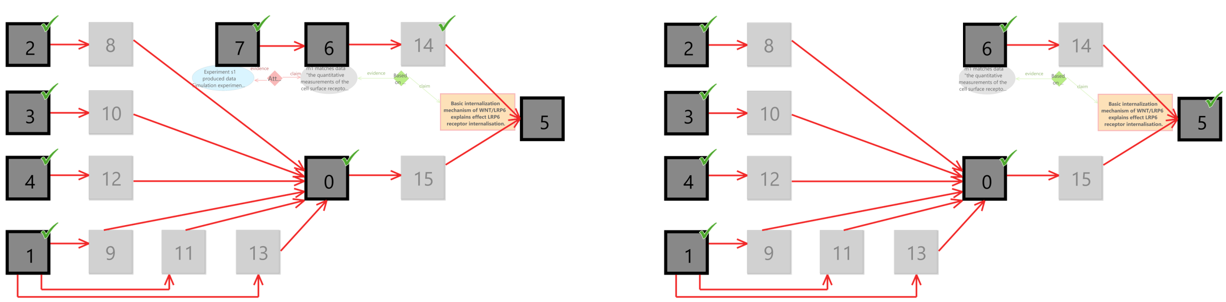


Figure 4: Moves (top right box) and argument graph generated for the beginning of the illustrative case study, also in Zschaler et al. (2025), Figure 1. Different colors and shapes correspond to different subtypes of `ArgumentElement` in the metamodel (Figure 1): orange rectangle – `MechanismExplainsEffect`, light blue ellipse – `ExperimentResult`, light gray rectangle – `SimulationMechanismWarrant`, light gray ellipses – `ModelMatchesDataOverTime` or `LiteratureEvidenceForDataAndEffect`.



(a) Abstract argumentation framework generated for the argument graph in Figure 4.

(b) Abstract argumentation framework generated without the `counterModel` move present.

Figure 5: Generated abstract argumentation frameworks with or without the experiment `s1`. Dark gray boxes map to arguments in the argument graph (numbers match the small gray boxes in Figure 4). Light gray boxes represent virtual arguments artificially generated to express `Support` relations. Red arrows represent attack relationships. Green ticks indicate the acceptable arguments in each case. For arguments 7, 6, 14, and 5 we have underlaid the corresponding elements from Figure 4. Note how argument 14 corresponds to the `Support` relationship and not an individual argument.

announcement of a specific claim with an underpinning argument (Almpani et al. 2023). Assume that a year later, someone finds a problem with the experiments described in a reference. How would this affect the validity of the claims by Haack et al. (2020)? Adding a corresponding move to the documentation of our study and re-evaluating with `plato` would allow us to analyze which arguments remain acceptable and, in particular, whether the acceptability of our overall claim has been affected. This is impossible to do from the provenance graph alone because it misses key information about how the information in the reference affects the overall study argument. Further automated analysis would require the formalization of artifacts, such as specifying requirements in temporal logic (Ruscheinski et al. 2020).

5 CONCLUSIONS AND OUTLOOK

We introduced a dialectic model based on an exchange of arguments and counterarguments to zoom in on the execution of simulation studies. Compared to checklists for conducting and documenting simulation studies (Grimm et al. 2014), this offers a complementary and more detailed view of the reasoning behind conducting simulation studies. The approach, based on arguments and counterarguments, promotes interaction, the generation of claims, their examination, refutation, and reasoning with others, with the final goal of converging on a solution. As stated by Almpani and Stefaneas (2023), such a dialectic approach may enhance creativity. In addition, the evolutionary process of weighting between different options, arguments and counterarguments in iteratively revising and conducting simulation studies is moved into the focus in conducting and documenting simulation studies.

Complementing our rather in-depth view with more coarse-grained approaches like checklists such as TRACE is a natural next step. Following Schneiderman's mantra of visualization—"overview first, zoom and filter, details on demand"—will help manage the complexity of conducting and documenting simulation studies, including the potentially large size of monotonically growing argument graphs. An integration with TRACE—for example, realized in Jupyter notebooks (Ayllón et al. 2021)—would allow users to zoom in and filter on critical aspects and, subsequently, based on the presented approach to hone in on arguments and counterarguments to ensure validity. Due to its interactive nature, the approach could easily be extended to be played by different types of users—for example, including stakeholders questioning specific parts of the simulation study. Also due to the reasoning support, changes within the validity of specific arguments would be propagated throughout the network of arguments and counterarguments, thus providing a living documentation of a simulation study and its validity. This might even be valuable in simulations integrated in digital twins, even though these may require subtly different moves and argument structures.

Careful scrutiny of simulation studies is particularly valuable in domains of high societal relevance and political dispute—for example, climate change (Randall et al. 2007), epidemics (Edeling et al. 2021), or studies leading to the development of medical devices (Center for Devices and Radiological Health 2023; Viceconti and Emili 2024). Although we took only the first step, we believe that our feasibility study shows the potential of dialectic models for documenting and conducting simulation studies. When integrated with other approaches and combined with visual support, we believe this approach has great potential for interactively and more effectively communicating simulation studies (Uhrmacher et al. 2024).

ACKNOWLEDGEMENTS

This work was partly supported by a seed-corn fund awarded through the EPSRC-funded MDENet network (grant reference EP/T030747/1) and by the DFG (grant 320435134). We thank Peter McBurney, Odinaldo Rodrigues, and Sanjay Modgil for helpful discussions and feedback about agent dialog and argumentation.

REFERENCES

Al-Zoubi, K., and G. Wainer. 2010. "Managing simulation workflow patterns using dynamic service-oriented compositions". In *Proceedings of the 2010 Winter Simulation Conference*, 765–777 https://doi.org/10.1109/WSC.2010.5679111.

Almpani, S., and P. Stefaneas. 2023. "Bridging informal reasoning and formal proving: The role of argumentation in proof-events". *Foundations of Science*:1–25 https://doi.org/10.1007/s10699-023-09926-9.

Almpani, S., P. Stefaneas, and I. Vandoulakis. 2023. "Formalization of mathematical proof practice through an argumentation-based model". *Global Philosophy* 33(3):33 https://doi.org/doi.org/10.1007/s10516-023-09685-z.

Ayllón, D., S. F. Railsback, C. Gallagher, J. Augusiak, H. Baveco, U. Berger, *et al*. 2021. "Keeping modelling notebooks with TRACE: Good for you and good for environmental research and management support". *Environmental Modelling & Software* 136:104932 https://doi.org/https://doi.org/10.1016/j.envsoft.2020.104932.

Balci, O. 2012. "A life cycle for modeling and simulation". *Simulation* 88(7):870–883 https://doi.org/10.1177/0037549712438469.

Bettini, L. 2016. *Implementing domain-specific languages with Xtext and Xtend*. Packt Publishing Ltd.

Black, E., N. Maudet, and S. Parsons. 2021. "Argumentation-based Dialogue". In *Handbook of Formal Argumentation, Volume 2*, edited by D. Gabbay, M. Giacomin, G. R. Simari, and M. Thimm. College Publications.

Brambilla, M., J. Cabot, M. Wimmer, and L. Baresi. 2017. *Model-driven software engineering in practice*. Second ed. Morgan & Claypool.

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* 17(8):1–27 https://doi.org/10.1371/journal.pcbi.1009227.

Center for Devices and Radiological Health 2023, November. "Assessing the credibility of computational modeling and simulation in medical device submissions: Guidance for industry and Food and Drug Administration staff". Technical report, U.S. Department of Health and Human Services; Food and Drug Administration.

Corneli, J., U. Martin, D. Murray-Rust, G. Rino Nesin, and A. Pease. 2019. "Argumentation theory for mathematical argument". *Argumentation* 33(2):173–214 https://doi.org/10.1007/s10503-018-9474-x.

Dung, P. M. 1995. "On the acceptability of arguments and its fundamental role in non-monotonic reasoning, logic programming and n-person games". *Artificial Intelligence* 77(2):321–357 https://doi.org/10.1016/0004-3702(94)00041-X.

Edeling, W., H. Arabnejad, R. Sinclair, D. Suleimenova, K. Gopalakrishnan, B. Bosak, *et al*. 2021. "The impact of uncertainty on predictions of the CovidSim epidemiological code". *Nature Computational Science* 1(2):128–135 https://doi.org/10.1038/s43588-021-00028-9.

Erdweg, S., T. van der Storm, M. Völter, L. Tratt, R. Bosman, W. R. Cook, *et al*. 2015. "Evaluating and comparing language workbenches: Existing results and benchmarks for the future". *Computer Languages, Systems & Structures* 44:24–47 https://doi.org/10.1016/j.cl.2015.08.007. Special issue on the 6th and 7th Int'l Conf Software Language Engineering (SLE 2013 and SLE 2014).

Eysholdt, M., and H. Behrens. 2010, October. "Xtext: implement your language faster than the quick and dirty way". In *Companion Proc. ACM Int'l Conf. Object Oriented Programming Systems Languages and Applications*, OOSPLA'10: ACM https://doi.org/10.1145/1869542.1869625.

Ford, M. J., and B. M. Wargo. 2012. "Dialogic framing of scientific content for conceptual and epistemic understanding". *Science Education* 96(3):369–391 https://doi.org/https://doi.org/10.1002/sce.20482.

Grimm, V., J. Augusiak, A. Focks, B. M. Frank, F. Gabsi, A. S. Johnston, *et al*. 2014. "Towards better modelling and decision support: Documenting model development, testing, and analysis using TRACE". *Ecological modelling* 280:129–139 https://doi.org/10.1016/j.ecolmodel.2014.01.018.

Haack, F., K. Budde, and A. M. Uhrmacher. 2020, August. "Exploring the mechanistic and temporal regulation of LRP6 endocytosis in canonical WNT signaling". *Journal of Cell Science* 133(15) https://doi.org/10.1242/jcs.243675.

Higgins, S. G., A. A. Nogiwa-Valdez, and M. M. Stevens. 2022, January. "Considerations for implementing electronic laboratory notebooks in an academic research environment". *Nature Protocols* 17(2):179–189 https://doi.org/10.1038/s41596-021-00645-8.

Khalifa, K. 2017. *Understanding, explanation, and scientific knowledge*. Cambridge University Press.

McBurney, P., and S. Parsons. 2009. "Dialogue games for agent argumentation". In *Argumentation in Artificial Intelligence*, 261–280. Springer US https://doi.org/10.1007/978-0-387-98197-0_13.

Modgil, S. 2013. "Revisiting abstract argumentation frameworks". In *International Workshop on Theory and Applications of Formal Argumentation*, 1–15. Springer https://doi.org/10.1007/978-3-642-54373-9_1.

Monks, T., C. S. Currie, B. S. Onggo, S. Robinson, M. Kunc, and S. J. Taylor. 2019. "Strengthening the reporting of empirical simulation studies: Introducing the STRESS guidelines". *Journal of Simulation* 13(1):55–67 https://doi.org/10.1080/17477778.2018.1442155.

Pease, A., J. Lawrence, K. Budzynska, J. Corneli, and C. Reed. 2017, May. "Lakatos-style collaborative mathematics through dialectical, structured and abstract argumentation". *Artificial Intelligence* 246:181–219 https://doi.org/10.1016/j.artint.2017.02.006.

Proietti, C., D. Grossi, S. Smets, and F. R. Velázquez-Quesada. 2019. *Bipolar Argumentation Frameworks, Modal Logic and Semantic Paradoxes*, 214–229. Springer Berlin Heidelberg https://doi.org/10.1007/978-3-662-60292-8_16.

Randall, D. A., R. A. Wood, S. Bony, R. Colman, T. Fichefet, J. Fyfe, *et al*. 2007. "Climate models and their evaluation". In *Climate change 2007: The physical science basis. Contribution of Working Group I to the Fourth Assessment Report of the IPCC (FAR)*, 589–662. Cambridge University Press.

Robinson, S. 2008, March. "Conceptual modelling for simulation Part I: definition and requirements". *Journal of the Operational Research Society* 59(3):278–290 https://doi.org/10.1057/palgrave.jors.2602368. Publisher: Taylor & Francis.

Rodrigues, O. 2018. "EqArgSolver – System Description". In *Proceedings of the 4th Intl. Conference on Theory and Applications of Formal Argumentation*, 150–158 https://doi.org/https://doi.org/10.1007/978-3-319-75553-3_11.

Rule, P., and V. M. John. 2015. "A necessary dialogue: Theory in case study research". *International Journal of Qualitative Methods* 14(4):1609406915611575 https://doi.org/10.1177/1609406915611575.

Ruscheinski, A., and A. Uhrmacher. 2017. "Provenance in modeling and simulation studies — Bridging gaps". In *2017 Winter Simulation Conference (WSC)*, 872–883 https://doi.org/10.1109/WSC.2017.8247839.

Ruscheinski, A., T. Warnke, and A. M. Uhrmacher. 2020. "Artifact-based workflows for supporting simulation studies". *IEEE Transactions on Knowledge and Data Engineering* 32(6):1064–1078 https://doi.org/10.1109/TKDE.2019.2899840.

Sargent, R. G. 2010, December. "Verification and validation of simulation models". In *Proceedings of the 2010 Winter Simulation Conference*, 166–183 https://doi.org/10.1109/WSC.2010.5679166. ISSN: 1558-4305.

Steinberg, D., F. Budinsky, M. Paternostro, and E. Merks. 2009. *EMF: Eclipse Modeling Framework*. 2nd ed. Addison-Wesley Professional.

Strüber, D., K. Born, K. D. Gill, R. Groner, T. Kehrer, M. Ohrndorf *et al*. 2017. "Henshin: A usability-focused framework for EMF model transformation development". In *Proc. 10th Int'l Conf on Graph Transformations (ICGT'17)*, edited by J. de Lara and D. Plump, 196–208: Springer https://doi.org/10.1007/978-3-319-61470-0_12.

Toulmin, S. E. 2003. "The Layout of Arguments". In *The Uses of Argument* (updated ed.)., 87–134. Cambridge University Press https://doi.org/10.1017/CBO9780511840005.007.

Uhrmacher, A. M., P. Frazier, R. Hähnle, F. Klügl, F. Lorig, B. Ludäscher, *et al*. 2024, August. "Context, composition, automation, and communication: The C2AC roadmap for modeling and simulation". *ACM Trans. Model. Comput. Simul.* 34(4) https://doi.org/10.1145/3673226.

van Emeren, F. H., R. Grootendorst, and T. Kruiger. 1987. *Handbook of Argumentation Theory*. Foris Publications.

Viceconti, M., and L. Emili. (Eds.) 2024. *Toward Good Simulation Practice*. Springer Nature https://doi.org/10.1007/978-3-031-48284-7.

Viyović, V., M. Maksimović, and B. Perisić. 2014. "Sirius: A rapid development of DSM graphical editor". In *IEEE 18th Int'l Conf Intelligent Engineering Systems (INES'14)*, 233–238 https://doi.org/10.1109/INES.2014.6909375.

Walton, D. 2013, November. *Argumentation Schemes for Presumptive Reasoning*. Routledge https://doi.org/10.4324/9780203811160.

Wilsdorf, P., O. Reinhardt, T. Prike, M. Hinsch, J. Bijak, and A. M. Uhrmacher. 2024. "Simulation studies of social systems: telling the story based on provenance patterns". *Royal Society Open Science* 11(8):240258 https://doi.org/10.1098/rsos.240258.

Wilsdorf, P., S. Zschaler, F. Haack, and A. M. Uhrmacher. 2024. "Potential and Challenges of Assurance Cases for Simulation Validation". In *2024 Winter Simulation Conference (WSC)*, 2166–2177 https://doi.org/10.1109/WSC63780.2024.10838818.

Winsberg, E. 2019. *Science in the age of computer simulation*. University of Chicago Press.

Wooldridge, M. J. 2012. *An introduction to multiagent systems*. 2. ed., repr. ed. Chichester: Wiley.

Zschaler, S., P. Wilsdorf, T. Godfrey, and A. M. Uhrmacher. 2025. "Towards dialectic models for documenting and conducting simulation studies: a vision". In *SIGSIM-PADS'25*.

AUTHOR BIOGRAPHIES

STEFFEN ZSCHALER is Reader in Software Engineering in the Department of Informatics at King's College London. His e-mail address is szschaler@acm.org.

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

THOMAS GODFREY is a core model-team member at Aerogility Ltd. His e-mail address is tom.godfrey@aerogility.com.

ADELINDE M. UHRMACHER is 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.