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.

ON THE USE OF GENERATIVE AI IN SIMULATION STUDIES: A REVIEW OF TECHNIQUES, APPLICATIONS AND OPPORTUNITIES

Niclas Feldkamp[1]

[1]Group for Information Technology in Production and Logistics, TU Ilmenau, Ilmenau, GERMANY

ABSTRACT

Although Large Language Models get a lot of attention, Generative Artificial Intelligence encompasses a variety of methods such as Generative Adversarial Networks, Variational Autoencoders or Diffusion Models, that all work very differently but are all capable of generating synthetic data. These methods have considerable potential to make simulation studies more efficient, especially through the creation of artificial data sets, automatic model parameterization and assisted result analysis. The aim of this study is to systematically classify generative methods and their applicability in the context of simulation studies. Based on a comprehensive literature review, applications, trends and challenges of generative methods that are used in combination with simulation are analyzed and structured. This is then summarized in a conceptual workflow that shows how and in which phase generative methods can be used advantageously in simulation studies.

1 INTRODUCTION

Generative Artificial Intelligence has gained enormous attention in recent years due to its rapid advances (Biever 2023). Contrary to the public perception that Generative AI revolves mostly around Large Language Models (LLMs), generative models and methods rather encompass a wide range of methods such as Generative Adversarial Networks (GANs) (Goodfellow et al. 2014), Variational Autoencoders (VAEs) (Kingma and Welling 2014), Diffusion Models (Song et al. 2021) and other modern and highly effective approaches, each suited for different tasks and applications. These methods have in common that they can generate new data that resemble existing patterns, which makes them attractive for many areas, including simulation studies. Simulation and modeling is a well-established method for planning, control, analysis, and optimization of systems of various types (Law 2015). Hybrid Systems Modelling describes an approach where methods and tools from other disciplines are used in one or more phases of a simulation study in a supportive function (Mustafee and Powell 2018). In this manner, generative methods could provide support in a number of ways, from generating input data up to assistance in the analysis and interpretation of results (Giabbanelli et al. 2024).

The aim of this paper is to provide a well-founded classification and evaluation of usage of existing generative methods in the context of simulation studies. A comprehensive literature review is conducted to explore the current state of research and to analyze work in which generative models have been used within simulation studies. Based on this evaluation, the potential and possible applications will be systematically structured in order to determine when and how generative methods can be used most effectively in the individual phases of a simulation study. The remainder of this paper is therefore as follows: Section 2 outlines the basics of generative models and gives an overview of simulation studies and hybrid systems modelling. Section 3 provides an extensive literature review as well as a conceptualization of the findings. This is followed by some concluding remarks and outlook in section 4.

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

2 RELATED WORK

2.1 Generative Models

Generative models are machine learning algorithms that extract key features from existing data and can generate new data on this basis. In contrast to discriminative models, which directly learn a decision boundary between input data X and target variable Y, generative models determine the probability distribution of the underlying data X and target variable Y (Ng and Jordan 2001). From this distribution, target values Y can then be inferred. Therefore, generative models can extract key features from data and even generate new data from the learned distribution function (Ng and Jordan 2001). They comprise many different approaches whose application context and relevance vary considerably. Generative models and algorithms have recently become increasingly important, especially in the field of Generative AI (Jovanovic and Campbell 2022). Recently, the focus here has been on methods for processing and generating natural language, in particular using so-called Large Language Models (LLMs). These can process texts and generate new text based on the recursive principle of predicting the next most probable word or token (Brown et al. 2020). LLMs are based on the principle of transformer architecture (Vaswani et al. 2017). This is a special architecture based on artificial neural networks that is very well suited for processing (usually very large) amounts of sequence data. These can be used to generate text in the sense of Large Language Models (Brown et al. 2020), but other tasks can also be efficiently accomplished using transformer networks, for example text classification, time series analysis and prediction or speech recognition (Vaswani et al. 2017). Transformer architectures are particularly helpful when converting from one sequence to another (Seq2Seq), e.g. for machine translation, text-to-speech or speech-to-text (Sutskever et al. 2014). Before the rise of transformer architectures, such sequence processing tasks were typically performed with so-called recurrent neural networks. A recurrent neural network (RNN) is a type of artificial neural network that processes information sequentially by using previous outputs as input for subsequent steps, allowing it to learn temporal dependencies, whereby the improved development of RNN-technology is called Long Short-Term Memory (LSTM) (Gers et al. 1999). RNNs can be used in a generative manner (Graves 2014), and while transformer networks are generally used for complex tasks that require training on very large amounts of data, generative recurrent networks (GRN) can be used efficiently for smaller tasks (Peng et al. 2023).

Apart from the dominance of transformer-based architectures, for the generation of rather static, non-sequential data, two other methods are relevant: Variational Autoencoder (VAE) (Kingma and Welling 2014) and Generative Adversarial Networks (GAN) (Goodfellow et al. 2014). A Variational Autoencoder (VAE) is a generative model that consists of an encoder and decoder. The encoder converts data into a compressed, probabilistic representation, so that realistic new data can be generated from this distribution (Kingma and Welling 2014). Generative Adversarial Networks (GANs), on the other hand, consist of two competing networks: a generator produces artificial data, while a discriminator attempts to distinguish real from generated data. Through mutual training, the generator produces increasingly realistic data until the discriminator can hardly differentiate between real and synthetic examples (Goodfellow et al. 2014). Both approaches were originally developed for image generation tasks but can also be used for other areas of application. In the field of image generation, Diffusion Models are currently among the most modern technologies (Song et al. 2021). Furthermore, Neural Radiance Fields (NeRFs) are the most recent development in this field and are used to generate 3D scenes from two-dimensional images (Mildenhall et al. 2020).

2.2 Hybrid Systems Modelling and Phases of a Simulation Study

According to the relevant literature in the field, a simulation study can typically be divided into multiple phases (Law 2003; Mustafee and Powell 2018; Wilsdorf et al. 2022): At the beginning there is a thorough investigation of the real problem, which serves as the starting point. This is followed by the transfer of this problem into a conceptual model that depicts the essential characteristics and relationships of elements in the underlying system. In the next step, this conceptual model is implemented in the form of a computer-based, executable simulation model, followed by a phase of verification and validation in which the correctness of the model is evaluated technically and to ensure that it actually and adequately represents its real-world counterpart. Finally, targeted experiments and analyses are carried out in order to find solutions for the formulated real-world problem. Some authors divide the phases of a simulation study into even more individual steps, while others combine various aspects into one phase. In this article, the phases of a simulation study are divided into the following seven categories. This categorization serves as a basis for the subsequent literature review in the next section and will be maintained consistently throughout the rest of the paper: *(1) problem definition and conceptual modelling, (2) simulation input modelling, (3) implementation, (4) verification and validation, (5) design of experiments, (6) execution of simulation runs, (7) analysis, presentation and documentation of results*. When combining multiple modeling approaches or simulation techniques within a single simulation, we refer to this as a hybrid simulation study (Mustafee and Powell 2018). This approach aims to enhance the functionality and accuracy of the simulation with respect to the system under investigation, or even to enable the development of models that would otherwise not be feasible. Hybrid systems modeling, in turn, describes the integration of different methods and techniques from various disciplines into one or more phases of the simulation study (Mustafee and Powell 2018). This also applies to the use of Generative AI within a simulation study, so when generative methods are employed in at least one phase of a simulation project, this can also be considered as a form of hybrid systems modeling.

3 THE USE OF GENERATIVE AI AND SIMULATION

3.1 Methodology

In order to evaluate the use of generative methods within simulation studies, a detailed literature review was carried out, based on the method of Webster and Watson (Webster and Watson 2002). For this purpose, the relevant databases for scientific literature were searched. The search terms used were the methods presented in Section 2.1 in combination with the keyword simulation. The literature identified was then filtered based on the following criteria, which is also illustrated in Figure 1:

- The simulation method used must correspond to the term simulation in the consensual sense of this conference, i.e. simulation is the replication of a real system by an artificial computer model in order to understand and analyze its behavior over time and derive insights from it (Banks 1999; Shannon 1998).
- The generative method must have been used within a simulation study. Although this did not have to be explicitly declared as a simulation study, it should be clear from the context that the phases of a simulation study presented in Section 2.2 were at least rudimentarily run through. The generative method used must provide a recognizable added value for the simulation study that would not have been achievable without its use. If the combination of simulation and generative method is only evaluated from a theoretical, potential-based perspective, the paper will only be used for analysis if the added value is sufficiently justified argumentatively and/or or with convincing prototypical efforts.
- If the same authors or group of authors deal with the (almost) identical topic in several papers, one paper was chosen to represent the corresponding approach.

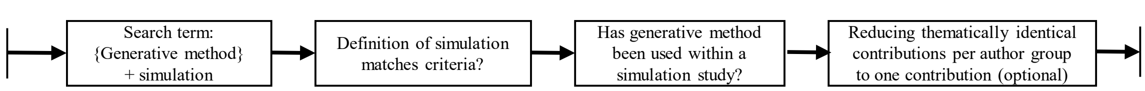

Figure 1: Summary of literature filtering process.

This left a total of 36 papers for in-depth analysis. It is evident that LLMs are by far the most widely used method, followed by GANs and generative recurrent networks. The extracted literature was then classified into two-dimensional concepts consisting of the generative method on the one hand and the phase (or phases) of the simulation study that the method was used in. Table 1 provides the comprehensive concept matrix summary of all the papers analyzed.

3.2 Literature Analysis

The first steps can be summarized as *Problem Definition and Conceptual Modeling*. However, there are only a few studies that use Generative AI in this phase. In addition, the use of Generative AI here mainly is limited to more theoretical analysis that points out future potential. For example, Giabbanelli et al. (2024) outline a concept in which end users pose problem questions in natural language and a LLM automatically assigns them to a suitable simulation model. According to this paper, this would mean a fundamental change in model selection and problem formulation, with language models taking over the mapping between analysis questions and existing simulation models. LLMs could also help to facilitate model conception. Akhavan and Jalali (2024) for example report that LLMs could be used to support the structuring of causal diagrams for system dynamics models, for example by suggesting variable relationships or questioning model assumptions in natural language.

A frequent use case for Generative AI according to the analyzed literature is during the phase of *Data Collection and Input Data Modeling*. In many simulation projects, preparing the input data is a critical step, and this is where generative methods are especially useful. For example, Kotnana et al. (2022) demonstrate how a GAN can help with the synthesis of artificial population data. Their method replaces or complements traditional demographic methods (such as iterative proportional fitting) and generates fine-grained synthetic populations as input for agent-based models. In addition, GANs have also been used to generate motion and trajectory data for simulations. Roy et al. (2022) use a GAN to generate realistic animal movement trajectories based on GPS tracking data of animal populations. These generated trajectories then serve as input data or scenarios in ecological simulations to investigate behavioral patterns with more variety than the limited available real-world data would provide. Montevechi et al. (2021) present a GAN-based method for modeling stochastic input distributions in discrete event simulation. Instead of using traditional statistical distribution fitting, their GAN learns complex distributions (e.g. arrival or process time distributions) directly from data, which can be advantageous for multivariate distributions or distributions that are difficult to parameterize. Sequence- and time-series-based generative models (LSTM, Seq2Seq) are frequently used in particular when temporal sequences need to be modeled or generated. Camargo et al. (2021) use a combination of GAN and LSTM for generating precise time-intervals for event-logs. They propose the use of a hybrid approach: A simulation model should first generate sequential event sequences and then a deep learning model can predict the time intervals between these events. A related approach is presented by Cen et al. (2020): here, a combination of LSTM and VAE was used to generate samples from a fitted random distribution for simulation input. They call this approach Neural Input Modeling (NIM). Jia et al. (2024) use a transformer network for modeling random distributions. They argue that this approach allows for a more realistic simulation of the distributions in highly complex stochastic systems, where traditional random distributions may fail to capture the underlying real-world distribution with sufficient accuracy. Borysov et al. (2019) use VAEs in the context of simulation input data generation for agent-based simulation of transportation systems. They use their VAE-based approach to generate scalable realistic agents with multiple attributes, which is particularly useful when empirical data is insufficient.

Regarding the phase of *Model Implementation*, two general sub-categories can be distinguished: The use of Generative AI to support model the implementation process (1) and the integration of AI directly into the simulation itself so that it can drive the model’s behavior (or certain parts of it) at runtime (2).

Table 1: Matrix showing all analyzed papers, mapped to generative method and corresponding phase within the simulation study. The color intensity is based on the number of papers in the respective cell. A paper can be assigned to multiple cells.

	Phase of Simulation Study						
	Problem Definition and Conceptual Modelling	**Data Collection and Simulation Input Modelling**	**Model Implementation**	**Validation and Verification**	**Design of Experiments**	**Execution of Runs**	**Analysis and Presentation of Results**
LLM	(Akhavan and Jalali 2024) (Giabbanelli et al. 2024)		(Akhavan and Jalali 2024) (Diamatopoulos et al. 2024) (Du Plooy and Oosthuizen 2023) (Ferraro et al. 2024) (Frydenlund et al. 2024) (Gao et al. 2024) (Ghaffarzadegan et al. 2024) (Jackson et al. 2024) (Park et al. 2023) (Vezhnevets et al. 2023)	(Akhavan and Jalali 2024) (Giabbanelli 2023)	(Giabbanelli et al. 2024)	(Giabbanelli et al. 2024) (Vezhnevets et al. 2023)	(Akhavan and Jalali 2024)) (Du Plooy and Oosthuizen 2023) (Giabbanelli 2023) (Giabbanelli et al. 2024)
VAE		(Borysov et al. 2019) (Cen and Haas 2022)				(Cen et al. 2020)	
GAN		(Camargo et al. 2021) (Kotnana et al. 2022) (Montevechi et al. 2022) (Roy et al. 2022)	(Bicher et al. 2024).	(Montevechi et al. 2022)	(Feldkamp et al. 2022)		
GRN		(Camargo et al. 2021) (Cen and Haas 2022) (Woerrlein and Strassburger 2020)				(Hajisharifi et al. 2024) (Cen et al. 2020)	
Transformer		(Jia et al. 2024)		(Maftouni et al. 2023)		(Chen et al. 2023) (Feng and Zhou 2024) (Geneva and Zabaras 2022) (Najafi and Lu 2023)	
Diffusion Model						(Chung et al. 2024) (Finn et al. 2024) (Shi et al. 2024)	
NeRF			(Chen et al. 2024) (Chen et al. 2025) (Li et al. 2023)				
Sum	*2*	*10*	*14*	*4*	*2*	*12*	*4*

The first sub-category for implementation primarily includes papers in which AI is used to generate code for creating the simulation model or to configure the simulation software based on the conceptual model. For example, Akhavan and Jalali (2024) emphasize that LLMs can generate code frameworks or derive simulation scripts from textual model descriptions for system dynamic models. According to this paper, this use of LLMs can speed up implementation, but requires human review for quality assurance of the generated code. Du Plooy and Oosthuizen (2023) similarly report that an LLM was able to reliably translate a given simple system dynamics model into working Python code. However, multiple papers agree that in principle, the LLM-based creation of simulation models in the sense of text-to-code is not yet mature enough for practical use, but promises great future potential (Akhavan and Jalali 2024; Du Plooy and Oosthuizen 2023; Frydenlund et al. 2024; Jackson et al. 2024).

When integrating Generative AI into the simulation model itself, a frequent use case in this context is agent-based modeling. Many papers focus on so-called generative agents. In this approach, LLMs are used directly as an integrated component within each agent in order to control its behavior. Park et al. (2023), for example, present an architecture for agents that integrates an LLM that is able to store, synthesize and retrieve agent's experiences in order to generate more dynamic behavior. Ferraro et al. (2024) and Vezhnevets et al. (2023) adopt similar approaches by integrating Large Language Models (LLMs) into their agents, allowing Generative AI to guide both the agents' decision-making and their interactions with one another. However, a challenge remains in managing the interaction between the LLM and the simulation environment. This includes, for example, converting the language model's text output into relevant actions within the simulation, as well as implementing processes to control computing time and ensure consistency (Ferraro et al. 2024; Vezhnevets et al. 2023). Ghaffarzadegan et al. (2024) introduce the term "*Generative Agent-Based Modeling*": Pre-trained language models are coupled with agent-based simulation in order to better simulate human decision. While this use case is dominated by LLMs, there is also one paper by Bicher et al. (2024) in which GANs are used to model the decision-making process of agents. Diamatopoulos et al. (2024) present a short concept for integrating LLMs into a discrete event simulation of block-chains. In this approach, LLMs can simulate malicious node behavior as well as cooperation attacks. Woerrlein and Strassburger (2020) use a generative seq2seq-LSTMs within a simulation model to add an additional dimensions of output data. In their application for production simulation, the generative method can provide a dynamic power consumption curve, which the discrete event-driven simulation technique itself cannot provide in this form. In the context of model implementation, it's also worth mentioning the potential of Generative AI in the field of image generation. Among the more modern generative methods are Neural Radiance Fields (NeRFs) and Diffusion Models. For example, in traffic and driving simulation scenarios, their capabilities are leveraged. Chen et al. (2024) emphasize in their review paper that realistic 3D scene generation and rendering using NeRFs is an emerging area of research in traffic simulation, with significant potential for future development. Li et al. (2023) present a platform for large-scale traffic simulation and discuss how NeRFs could be used to generate synthetic traffic scenes from driving videos without having to manually invest the effort to create 3D assets of vehicles or buildings. Chen et al. (2025) show a similar approach in which NeRFs are used to generate realistic driving simulation environments with high visual quality based on lidar and camera data. These can then be used as a simulated training environment for autonomous driving.

The phase of *Verification and Validation* is traditionally one of the most demanding steps of a simulation study, but there are only a few noteworthy approaches that make use of Generative AI: Akhavan and Jalali (2024) outline that a LLM can provide hints for debugging system dynamics models, e.g. by suggesting possible implementation errors or logic errors based on a description of unexpected behavior. However, there are still hardly any real-world implementations and results available. In fact, the experiments by Du Plooy and Oosthuizen (2023) showed that although a LLM was capable of designing a model, it could not autonomously identify or correct existing errors (such as wrong parameter values). There is therefore still a methodological gap here: LLMs have so far seem to lack reliable mechanisms to understand the internal computational logics of simulation models to make targeted bug fixes. However, for validation purposes the use-cases are more mature. For example, Montevechi et al. (2022) use GANs in a very innovative way for the validation of simulation models. Here, the discriminator of a GAN is used to check the output of a simulation model against real-world data in order to assess whether the data generated by the simulation model is indistinguishable from real-world data. This method leverages the pattern recognition capabilities of the GAN's discriminator network very cleverly. However, the V&V with AI support still holds a lot of untapped potential. Giabbanelli (2023) outlines that future developments in LLMs could enable such applications, although they have so far remained largely of conceptual/theoretical nature. According to this paper, these models could potentially help identify hypotheses for possible simulation results or reveal errors in model assumptions that are otherwise difficult to explain. Maftouni et al. (2023) use transformers in the V&V phase for system dynamics models. Specifically, they present a concept for calibration in which a transformer-based deep learning model is used specifically for parameter identification, that works as follow: first, the system dynamics model itself is used to generate a large amount of synthetic data. This data then serves as a training basis for the transformer, which learns the relationships between the simulation model's outputs and the underlying parameters. After training, the transformer can then predict the appropriate model parameters for new input data, enabling an automated, data-driven calibration of the simulation model.

Generative AI has so far been used rather rarely for the *Design of Experiments*. Giabbanelli et al. (2024) address experiment design by proposing that an LLM could automatically derive suitable simulation runs from an end user's query. Feldkamp et al. (2022) present a concept that uses GANs to generate simulation experiment plans. Specifically, they use GANs in the context of robustness optimization of production systems, where two GANs alternately generate experiment plans for decision and noise factors in order to arrive at a solution that is as robust as possible, i.e. insensitive to variation from noise factors.

In the phase of *Execution of Runs*, the dominant use case for Generative AI clearly is metamodeling, also known as surrogate modeling. Deep learning-based methods have become indispensable in the field of metamodeling and generative models are also finding their way into this discipline. The aim of metamodeling is to create a surrogate using some initial data from the simulation model. This surrogate can then predict the simulation output faster in order to save computational effort, which is beneficial for simulation models with long runtimes (Barton 2015). A good example for using Generative AI in terms of metamodeling is the work of Cen and Haas (2022). They used graph neural networks in combination with VAE and LSTM, capable of imitating the dynamic outputs of a simulation model. Instead of predicting static single outputs, this metamodel generates entire time-series of output data. This enable rapid prediction of numerous output scenarios, making it especially valuable in the context of digital twins, where fast and extensive scenario analysis is essential. Another interesting publication is by Shi et al. (2024), which demonstrate an innovative approach for coupling physical simulations with a Diffusion Model. Diffusion Models belong to a newer class of generative models and are primarily intended for image generation tasks. They are therefore not yet widely used in simulation projects. In this paper, however, a Diffusion Model was used to create a metamodel. This approach is based on flow and heat simulations. Those simulations serve as input for a Diffusion Model, which then generates high-resolution flow fields. Occasionally, expensive, exact simulation results are used to specifically control the Diffusion Model. In this way, the Diffusion Model generates high-quality samples that come very close to the results of a complex numerical simulation and can be used in applications such as fluid dynamics and heat transport in additive manufacturing (Shi et al. 2024). Chung et al. (2024) use Diffusion Models to generate physically plausible initial conditions in fluid simulations, which significantly reduces simulation time through better initialization. Even if this is not metamodeling in the traditional way, it is included in this overview as metamodeling in the broader sense. Finn et al. (2024) use Diffusion Models to build simulation surrogates for the fast and efficient forecasting of sea-ice. Transformer networks are also frequently used as simulation surrogates in the context of metamodeling. This has been applied, for example, for discrete event models (Najafi and Lu 2023) or for physical model such as heat transfer simulation, fluid mechanics simulation (Chen et al. 2023; Geneva and Zabaras 2022) or solid mechanics simulation (Feng and Zhou 2024). This shows that transformer models are able to learn physical laws and make them available in the form of fast approximations (Geneva and Zabaras 2022). To surrogate simulation models of fluid-particle systems, Hajisharifi et al. (2024) present an approach based on LSTMs. With regard to further applications in the context of *Execution of Runs* besides metamodeling, Giabbanelli et al. (2024) propose the idea that an LLM could initiate the simulation run autonomously once the inputs have been identified, for example, by issuing a command to the simulation software as soon as all parameters have been defined.

As expected, the final phase, *Analysis and Presentation of Results*, exclusively contains papers that propose the use of LLMs. However, there is still surprisingly little research in this area, and practical applications and case-studies remain rare. Much of the work in this field tends to be either a work-in-progress or based on theoretical/prototypical visions. However, this situation is likely to change rapidly in the near future, since LLMs seem to be one of the most trending topics in any data-science-related research. Giabbanelli was among the first to highlight the potential of LLMs for data analysis tasks within simulation studies for automatic summarization of simulation results. For example, Giabbanelli (2023) suggests using LLMs for documentation to automatically generate sections of a research report or project documentation based on model descriptions, input data, and results. Akhavan and Jalali (2024) mention the use of LLMs to write up or even interpret results from system dynamics models, and that LLMs could be used to help create the model documentation. Giabbanelli et al. (2024) expanded on their vision in a subsequent paper, proposing that LLMs could have the potential to streamline the entire simulation process for end users, from formulating questions to interpreting results, all through a conversational interface. Their vision is to have an end-to-end LLM-powered system that enables users without simulation expertise to access existing models, execute simulation runs, and get the results in an intuitive, dialogue-based format. Table 2 shows a summary of the technical properties of the generative methods considered in this review as well as their possible applications in the context of simulation studies.

Table 2: Technical details, strengths and weaknesses of relevant generative methods.

	Technical details	**Strengths in specific phases of the simulation study**	**Relative advantages and disadvantages compared to other methods**
LLM	Transformer-based neural networks, sequence data processing, text generation by predicting next tokens.	Particularly suitable for problem definition, model implementation (code generation), model logic substitution (generative agents), and for analysis / documentation of results.	+ Excellent for large amounts of data, text processing, interaction in natural language. - High computational requirements, difficult to interpret/validate, numerical precision must be considered.
VAE	Encoder-decoder architecture with probabilistic latent space. Generates data from learned distributions.	Well suited for generating agent attributes, for time series generation and as a surrogate model.	+ Stable for probabilistic models, advantageous for small amounts of data; more robust training than GANs. - Lower quality of detail than GANs or diffusion models, less suitable for sequential data.
GAN	Two competing neural networks (generator & discriminator), adversarial training.	Particularly strong for generating realistic input data (e.g. population data, movement data). Also suitable for validation using discriminator network and for generating experiment designs.	+ Very realistic data generation (e.g. movement profiles). - Instability in training; less robust than VAEs; inefficient for sequential tasks.
GRN	Sequential and recurrent processing through the neural network. LSTM-architecture for improved modeling of temporal dependencies in sequences.	Useful for generating time series in input data or for generating dynamic output as a surrogate model (e.g. energy curves).	+ Less resource-intensive than transformer for time series data. Efficient for small data sets. - Weaker for long contexts. Slower & less scalable than transformer for large data sets.
Trans-former	Specialized neural networks for large amounts of sequence data. Versatile applications (text classification, prediction, translation).	Particularly suitable for surrogate modeling in physical simulations, as well as for model calibration.	+ Very scalable & precise for complex tasks. Superior to GRNs for long-term dependencies. - High memory & computational requirements and long training times.
Diffusion Model	Iterative data reconstruction from noise. Primarily used for image generation.	Suitable for high-resolution surrogate models in physical simulations (e.g. flow fields).	+ Excellent quality & resolution for physical simulation scenarios. More stable than GANs. - Long inference times, high computational requirements and less established for structured numerical data.
NeRF	Generation of three-dimensional scenes from two-dimensional images using volumetric neural networks.	Realistic generation of 3D assets and environments for simulation models and virtual training environments.	+ Superior in realistic 3D scene generation from 2D views. - Limited flexibility outside of visual contexts. Very high computing & data requirements.

3.3 Conceptualization

The broad perspective of a Generative AI-based end-to-end system, as postulated in (Giabbanelli et al. 2024), is only partially a reality as of today. However, the overall synthesis across all the papers analyzed shows that Generative AI can be used in all phases of a simulation study. Figure 2 shows a summary of the key areas for integration of Generative AI along the phases of a simulation study according to the literature reviewed in the previous section.

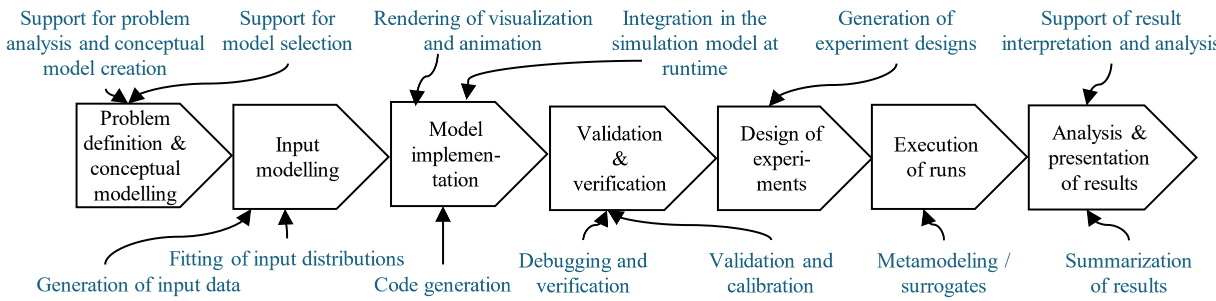

Figure 2: Key areas of application for Generative AI in simulation studies.

Generative AI is most frequently used in the phases of input modeling, model implementation and execution of runs (especially metamodeling). The phases that remain less explored are the initial problem definition and concept modeling and – surprisingly – the automated analysis and documentation of results. Especially in the analysis phase, the rising popularity and integration of Large Language Models suggest rapid and widespread future adoption.

In the context of problem definition and conceptual model creation, Generative AI can help to better grasp the problem to be simulated and to select a suitable model. In the field of input modeling, Generative AI is already frequently used, particularly in the generation of realistic synthetic data, for example, arrival times, movement patterns or population characteristics. It is also possible to generate synthetic scenarios or alternative system states, which helps when defining model boundaries or agent attributes. Methods such as GANs and VAEs are able to approximate complex distributions and generate realistic data from a small number of examples. This is particularly valuable when historical data is incomplete or difficult to access. Diffusion Models and NeRFs extend this potential by generating image or 3D data that can be used for realistic simulation environments, visualizations and animations.

During implementation, generative models support the technical design of the simulation. Either by supporting the creation of the model or through integration into the model itself, for example to control model logic using neural networks or to generate additional output dimensions. In agent-based models, for example, the internal decision logic can be replaced by generative methods, which is beneficial for implementing dynamic and realistic behavior. In the area of verification and validation, automation through Generative AI is not very common, but initial approaches do exist, both for debugging support and through automated validation and calibration.

In the design of experiments phase, generative methods can make valuable contributions to the generation of experiment plans, for example for the automated definition of new or extreme scenarios, which are required for robustness and sensitivity analyses. Their ability to generate parameter configurations beyond traditional sampling methods enables the targeted generation of useful scenarios, cases and experiments. This creates a bridge to the next phase: the execution of simulation runs. This phase is dominated by the approach of using generative models of all kinds in the form of metamodeling, i.e. to approximate the simulation model or sub-processes of it in order to reduce the computational effort for complex simulation models with long runtimes.

The phase of analysis, presentation and documentation of results shows promising potential applications for Generative AI. Especially LLM-based methods can be leveraged to summarize results or automatically generate management reports. Comprehensible, sophisticated texts can be generated from model descriptions, code fragments and result logs. Although this use case is currently still rather visionary, it is foreseeable that such automated analysis and documentation processes will become increasingly important in the future, both in a practice and scientific context.

4 CONCLUSION AND OUTLOOK

Generative AI enriches all aspects of simulation studies, from data collection and model implementation to the interpretation of results. This work provides a comprehensive overview of the current state of research and demonstrates the extent to which Generative AI can support and enhance the complex process of a simulation study. The broad spectrum of approaches enables a deeper understanding of how simulation and AI can grow together in the future. However, despite the variety of the papers analyzed, there are also noticeable gaps and underrepresented topics. There is currently still a lack of holistic approaches that cover the entire simulation life cycle using Generative AI. Although there are already visions of an end-to-end use of Large Language Models, these have so far remained largely conceptual and focus on one specific method. In practical implementations and applications, the focus is usually on one isolated phase of the simulation study. A cross-system framework that automatically integrates a variety of Generative AI methods from problem formulation through implementation to the presentation of results does not yet exist. This represents both a methodological challenge (integration of different methods) and an organizational one (lack of standards and interfaces). Although especially LLMs enable enormous progress in interaction using natural language, they also raise methodological questions for the simulation community, particularly regarding the validity and credibility of models. For example, hallucinations or training biases in Generative AIs could affect the validity of simulation results. Although such problems have already been discussed and theoretically considered in some papers, there has so far been a lack of structured studies on quality assurance measures when using Generative AI in simulation. Methodological guidelines or benchmarks for evaluating generative simulation elements have not yet been established. In addition, issues relating to the integration of Generative AI itself into simulation models must also be investigated more intensively in the future.

REFERENCES

Akhavan, A., and M. S. Jalali. 2024. “Generative AI and Simulation Modeling: How Should You (Not) Use Large Language Models like ChatGPT”. *System Dynamics Review* 40(3):1–27.

Banks, J. 1999. “Introduction to Simulation”. In *Proceedings of the 31st Conference on Winter Simulation Simulation,* 7–13 https://doi.org/10.1145/324138.324142.

Barton, R. R. 2015. “Tutorial: Simulation Metamodeling”. In *Proceedings of the 2015 Winter Simulation Conference,* 1765–1779 https://doi.org/10.1109/WSC.2015.7408294.

Bicher, M., D. Brunmeir, and N. Popper. 2024. “Modeling of Agent Decisions Using Conditional Generative Adversarial Networks”. In *2024 Winter Simulation Conference (WSC),* 2643–2654 https://doi.org/10.1109/WSC63780.2024.10838996.

Biever, C. 2023. “ChatGPT Broke the Turing Test - the Race Is on for New Ways to Assess AI”. *Nature* 619(7971):686–689.

Borysov, S. S., J. Rich, and F. C. Pereira. 2019. “How to Generate Micro-Agents? A Deep Generative Modeling Approach to Population Synthesis”. *Transportation Research Part C: Emerging Technologies* 106:73–97.

Brown, T. B., B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, *et al*. 2020. "Language Models are Few-Shot Learners". In *Proceedings of the 34th International Conference on Neural Information Processing Systems,* edited by H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin. Red Hook, NY, USA: Curran Associates Inc.

Camargo, M., M. Dumas, and O. González-Rojas. 2021. “Discovering Generative Models from Event Logs: Data-Driven Simulation Vs Deep Learning”. *PeerJ Computer Science* 7(4):e577.

Cen, W., and P. J. Haas. 2022. “Enhanced Simulation Metamodeling via Graph and Generative Neural Networks”. In *Proceedings of the 2022 Winter Simulation Conference,* 2748–2759 https://doi.org/10.1109/WSC57314.2022.10015361.

Cen, W., E. A. Herbert, and P. J. Haas. 2020. “NIM: Modeling and Generation of Simulation Inputs Via Generative Neural Networks”. In *Proceedings of the 2020 Winter Simulation Conference,* 584–595 https://doi.org/10.1109/WSC48552.2020.9383966.

Chen, D., M. Zhu, H. Yang, X. Wang, and Y. Wang. 2024. “Data-Driven Traffic Simulation: A Comprehensive Review”. *IEEE Transactions on Intelligent Vehicles* 9(4):4730–4748.

Chen, Q., L. Kong, F. Dugast, and A. To. 2023. "Using the Transformer Model for Physical Simulation: An Application on Transient Thermal Analysis for 3D Printing Process Simulation". In *NeurIPS 2023 AI for Science Workshop,* December 16th, New Orleans, USA.

Chen, Y., J. Zhang, Z. Xie, W. Li, F. Zhang, and J. Lu *et al.* 2025. "S-NeRF++: Autonomous Driving Simulation via Neural Reconstruction and Generation". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 47(6):4358–4376.

Chung, J., A. Marcato, E. J. Guiltinan, T. Mukerji, H. Viswanathan, and Y. T. Lin *et al.* 2024. "Accelerating Multiphase Simulations With Denoising Diffusion Model Driven Initializations". *Journal of Geophysical Research: Machine Learning and Computation* 1(4):1–13.

Diamatopoulos, G., G. Theodoropoulos, N. Tziritas, and R. Bahsoon. 2024. "Towards LLM Augmented Discrete Event Simulation of Blockchain Systems". In *Proceedings of the 38th ACM SIGSIM Conference on Principles of Advanced Discrete Simulation,* edited by M. Loper and A. Pellegrini, 75–76. New York, NY, USA: ACM.

Du Plooy, C., and R. Oosthuizen. 2023. "AI Usefulness in Systems Modelling and Simulation: GPT-4 Application". *South African Journal of Industrial Engineering* 34(3):286–303.

Feldkamp, N., S. Bergmann, F. Conrad, and S. Strassburger. 2022. "A Method Using Generative Adversarial Networks for Robustness Optimization". *ACM Transactions on Modeling and Computer Simulation.* 32(2):1–22.

Feng, B., and X.-P. Zhou. 2024. "The Novel Graph Transformer-Based Surrogate Model for Learning Physical Systems". *Computer Methods in Applied Mechanics and Engineering* 432:117410.

Ferraro, A., A. Galli, V. La Gatta, M. Postiglione, G. M. Orlando, D. Russo, *et al*. 2024. "Agent-Based Modelling Meets Generative AI in Social Network Simulations". In *Social Networks Analysis and Mining: 16th International Conference,* edited by L. M. Aiello, T. Chakraborty, and S. Gaito, 155–170. Berlin, Heidelberg: Springer.

Finn, T. S., C. Durand, A. Farchi, M. Bocquet, P. Rampal, and A. Carrassi. 2024. "Generative Diffusion for Regional Surrogate Models From Sea‐Ice Simulations". *Journal of Advances in Modeling Earth Systems* 16(10):1–38.

Frydenlund, E., J. Martínez, J. J. Padilla, K. Palacio, and D. Shuttleworth. 2024. "Modeler in a Box: How Can Large Language Models Aid in the Simulation Modeling Process?". *SIMULATION* 100(7):727–749.

Gao, C., X. Lan, N. Li, Y. Yuan, J. Ding, and Z. Zhou *et al.* 2024. "Large Language Models Empowered Agent-Based Modeling and Simulation: A Survey and Perspectives". *Humanities and Social Sciences Communications* 11(1):1–24.

Geneva, N., and N. Zabaras. 2022. "Transformers for Modeling Physical Systems". *Neural Networks* 146(2022):272–289.

Gers, F. A., J. Schmidhuber, and F. Cummins. 1999. "Learning to Forget: Continual Prediction with LSTM". In: *ICANN 99: Ninth International Conference on Artificial Neural Networks*, 850–855. London, UK: Inst. of Electrical Engineers.

Ghaffarzadegan, N., A. Majumdar, R. Williams, and N. Hosseinichimeh. 2024. "Generative Agent-Based Modeling: An Introduction and Tutorial". *System Dynamics Review* 40(1):e1761.

Giabbanelli, P. J. 2023. "GPT-Based Models Meet Simulation: How to Efficiently use Large-Scale Pre-Trained Language Models Across Simulation Tasks". In *2023 Winter Simulation Conference (WSC),* 2920–2931 https://doi.org/10.1109/WSC60868.2023.10408017.

Giabbanelli, P. J., J. J. Padilla, and A. Agrawal. 2024. "Broadening Access to Simulations for End-Users via Large Language Models: Challenges and Opportunities". In *2024 Winter Simulation Conference (WSC),* 2535–2546 https://doi.org/10.1109/WSC63780.2024.10838754.

Goodfellow, I., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair *et al*. 2014. "Generative Adversarial Nets". In Advances in Neural Information Processing Systems 27 (NIPS 2024), edited by Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, 2672–2680. Red Hook, NY, USA: Curran Associates, Inc.

Graves, A. 2014. "Generating Sequences With Recurrent Neural Networks", Arxiv preprint. https://doi.org/10.48550/arXiv.1308.0850.

Hajisharifi, A., R. Halder, M. Girfoglio, A. Beccari, D. Bonanni, and G. Rozza. 2024. "An LSTM-Enhanced Surrogate Model to Simulate the Dynamics of Particle-Laden Fluid Systems". *Computers & Fluids* 280(2024):106361.

Jackson, I., M. Jesus Saenz, and D. Ivanov. 2024. "From Natural Language to Simulations: Applying AI to Automate Simulation Modelling of Logistics Systems". *International Journal of Production Research* 62(4):1434–1457.

Jia, F., H. Zhu, F. Jia, X. Ren, H. Tan, and S. Chen *et al.* 2024. "SIMDIFF: Modeling and Generation of Stochastic Discrete-Event Simulation Inputs via Denoising Probabilistic Diffusion Process". In *2024 Winter Simulation Conference (WSC),* 2691–2702 https://doi.org/10.1109/WSC63780.2024.10838756.

Jovanovic, M., and M. Campbell. 2022. "Generative Artificial Intelligence: Trends and Prospects". *Computer* 55(10):107–112.

Kingma, D. P., and M. Welling. 2014. "Auto-Encoding Variational Bayes". In *2nd International Conference on Learning Representations (ICLR2014)*, April 14th-16th, Banff, AB, Canada.

Kotnana, S., D. Han, T. Anderson, A. Zufle, and H. Kavak. 2022. "Using Generative Adversarial Networks to Assist Synthetic Population Creation for Simulations". In *2022 Annual Modeling and Simulation Conference (ANNSIM),* edited by C. Ruiz Martin, 199–210. Piscataway, N.J., USA: IEEE inc.

Law, A. M. 2015. *Simulation Modeling and Analysis.* 5th ed. New York, NY: McGraw-Hill Education.

Law, A. M. 2003. "How to Conduct a Successful Simulation Study". In *Proceedings of the 2003 Winter Simulation Conference,* 66–70 https://doi.org/10.1109/WSC.2003.1261409.

Li, Q., Z. Peng, L. Feng, Z. Liu, C. Duan, W. Mo *et al*. 2023. "ScenarioNet: Open-Source Platform for Large-Scale Traffic Scenario Simulation and Modeling". In *Advances in Neural Information Processing Systems*, edited by A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, 3894–3920. Red Hook, NY, USA: Curran Associates, Inc.

Maftouni, M., N. Ghaffarzadegan, and Z. Kong. 2023. "A Deep Learning Technique for Parameter Estimation in System Dynamics Models". *SSRN ID 4440489 available at https://dx.doi.org/10.2139/ssrn.4440489*.

Mildenhall, B., P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. 2020. "NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis". In *Computer Vision – ECCV 2020: 16th European Conference*, edited by A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, 405–421. Cham, Switzerland: Springer International Publishing.

Montevechi, J. A. B., G. T. Gabriel, A. T. Campos, C. H. dos Santos, F. Leal, and M. E. F. H. S. Machado. 2022. "Using Generative Adversarial Networks to Validate Discrete Event Simulation Models". In *Proceedings of the 2022 Winter Simulation Conference,* 2772–2783 https://doi.org/10.1109/WSC57314.2022.10015375.

Montevechi, J. A. B., A. T. Campos, G. T. Gabriel, and C. H. dos Santos. 2021. "Input Data Modeling: An Approach Using Generative Adversarial Networks". In *2021 Winter Simulation Conference, 1–12* https://doi.org/10.1109/WSC52266.2021.9715407.

Mustafee, N., and J. H. Powell. 2018. "From Hybrid Simulation to Hybrid Systems Modelling". In *Proceedings of the 2018 Winter Simulation Conference,* 1430–1439 https://doi.org/10.1109/WSC.2018.8632528.

Najafi, H., and X. Lu. 2023. "DeepSim: A Transformer Based Model for Fast Simulation and Exploring Computer System Design Space". In *ACM SIGSIM Conference on Principles of Advanced Discrete Simulation,* edited by M. Loper, D. Jin, and C. D. Carothers, 54–55. New York, NY, USA: ACM.

Ng, A. Y., and M. I. Jordan. 2001. "On Discriminative vs. Generative Classifiers: A Comparison of Logistic Regression and Naive Bayes". In *Proceedings of the 15th International Conference on Neural Information Processing Systems: Natural and Synthetic (NIPS'01),* edited by T. G. Dietterich, S. Becker, and Z. Ghahramani, 841–848. Cambridge, MA, USA: MIT Press.

Park, J. S., J. O'Brien, C. J. Cai, M. R. Morris, P. Liang, and M. S. Bernstein. 2023. "Generative Agents: Interactive Simulacra of Human Behavior". In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology,* edited by S. Follmer, J. Han, J. Steimle, and N. H. Riche, 1–22. New York, NY, USA: ACM.

Peng, B., E. Alcaide, Q. Anthony, A. Albalak, S. Arcadinho, S. Biderman, *et al*. 2023. "RWKV: Reinventing RNNs for the Transformer Era". In *Findings of the Association for Computational Linguistics: EMNLP 2023,* edited by H. Bouamor, J. Pino, and K. Bali, 14048–14077. Stroudsburg, PA, USA: Association for Computational Linguistics.

Roy, A., R. Fablet, and S. L. Bertrand. 2022. "Using Generative Adversarial Networks (GAN) to Simulate Central-Place Foraging Trajectories". *Methods in Ecology and Evolution* 13(6):1275–1287.

Shannon, R. E. 1998. "Introduction to the Art and Science of Simulation". In *Proceedings of the 1998 Winter Simulation Conference,* 7–14 https://doi.org/10.5555/293172.293175.

Shi, N., H. Yan, S. Guo, and R. A. Kontar. 2024. Multi-physics Simulation Guided Generative Diffusion Models with Applications in Fluid and Heat Dynamics. arxiv preprint, https://doi.org/10.48550/arXiv.2407.17720.

Song, Y., J. N. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole. 2021. "Score-Based Generative Modeling through Stochastic Differential Equations". In *9th International Conference on Learning Representations, May* 3rd-7th, Virtual only.

Sutskever, I., O. Vinyals, and Q. V. Le. 2014. "Sequence to Sequence Learning with Neural Networks". In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*, edited by Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, 3104–3112. Cambridge, MA, USA: MIT Press.

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, *et al*. 2017. "Attention Is All You Need". In *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, edited by I. Guyon, U. von Luxburg, S. Bengio, H. Wallach, and R. Fergus, 5998–6008. Red Hook, NY, USA: Curran Associates Inc.

Vezhnevets, A. S., J. P. Agapiou, A. Aharon, R. Ziv, J. Matyas, E. A. Duéñez-Guzmán *et al*. 2023. Generative Agent-Based Modeling with Actions Grounded in Physical, Social, or Digital Space Using Concordia. arxiv preprint, https://doi.org/10.48550/arXiv.2312.03664.

Webster, J., and R. T. Watson. 2002. "Analyzing the Past to Prepare for the Future: Writing a Literature Review". *MIS Quarterly* 26(2):xiii–xxiii.

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

Woerrlein, B., and S. Strassburger. 2020. "A Method for Predicting High-Resolution Time Series Using Sequence-to-Sequence Models". In *Proceedings of the 2020 Winter Simulation Conference,* 1075–1086 https://doi.org/10.1109/WSC48552.2020.9383969.

AUTHOR BIOGRAPHIES

NICLAS FELDKAMP is a Research Scientist in the Group for Information Technology in Production and Logistics at Technische Universität Ilmenau, Germany. He holds a PhD in Information Systems and focuses on modeling and simulation, with an emphasis on integrating machine learning, data mining, and AI methods. His research includes the development of data farming approaches and the analysis of large-scale simulation datasets to support complex decision-making processes.