Journal of Telecommunication, Electronic and Computer Engineering
ISSN: 2180 – 1843 e-ISSN: 2289-8131 Vol. 16 No. 4 (2024) 57 – 64
jtec.utem.edu.my
DOI: https://doi.org/10.54554/jtec.2024.16.04.008

An Integrated Architecture for Validation of Simulation Models and Validation Methods Classification

Zainab Hassan Muhamad[1], Dhafer Abdulameer Abdulmonim[2]*
[1]*Department of Computer Science, Gifted Student School, Gifted Guardianship Committee, Ministry of Education, Najaf, Iraq.*
[2]*Department of Computer Science, Open Educational College, Ministry of Education, Najaf, Iraq.*

Article Info

Article history:
Received Jul 10th, 2024
Revised Sep 26th, 2024
Accepted Nov 24th, 2024
Published Dec 20th, 2024

Index Terms:
Verification and Validation
Simulation Modeling
Validation Process
Validation Architecture
Validation Methods

Abstract

Validation of simulation models is a crucial aspect of the software engineering simulation modelling process. It aims to reduce costs, save time, mitigate risks, and enhance understanding of the validation process's functions. This paper reviewed existing literature on simulation model validation, focusing on definitions, architectures, and methods. However, the study identified that it remains unclear which attributes should be prioritized when validating simulation models. Furthermore, the literature lacks a comprehensive architecture and classification to effectively describe the validation process. To address these gaps, this paper proposes an integrated architecture for the Simulation Model Validation Process (SMVP). Furthermore, it introduces a novel classification that incorporates three essentials elements: simulation validation phases, simulation models, and validation methods. The findings indicate that the primary focus of verification method is to verify the accuracy of model behavior. In contrast, structural validation emphasizes historical data validation, historical methods, and internal validation. Moreover, combining the structural and behavioral methods ensure a more robust and effective validation process. In conclusion, the proposed architecture and classification are expected to significantly contribute to future research by ensuring that simulation models are thoroughly validated during the validation processes.

*Corresponding Author: Dhafer985@gmail.com

I. INTRODUCTION

In recent years, Validation and Verification (V&V) of simulation Models has played an important role in ensuring low cost, reliability and safety in software engineering [1]. In addition, simulation models are significant when discussing the concept of verification and validation [2]. Besides, it is crucial to validate any simulation model before use [3]. Proper system modeling architecture and efficient V&V methods can improve simulation model efficiency and reusability [4]. Simulation models has become very important for solving problems and are used in many projects. They help reduce cost, time, and errors while providing a better understanding of organizational requirements [5]. Moreover, simulation models are critical for evaluating rapid system prototyping [6]. However, they are sometimes imprecise and becomes increasingly complex, which makes software simulators slower. While software simulators focus on evaluating small benchmarks, they often become very slow when simulating entire software [7].

The simulation model of a system is generally distinct from a dynamic model of the same system. Simulation software often employs computer simulation packages such as MATLAB and Stella to verify the dynamic behavior of the system. Modeling software provides validation, evaluation, and optimization to assist designers [8]. One of the challenges faced by simulation analysts is ensuring that the simulation model accurately represents the actual system. If the simulation model fails to represent the system correctly, the results derived from it may be inaccurate, leading to significant errors and increased costs in decision-making [9,10]. This highlights the importance of validation in any simulation project. For a simulation software to meet user needs effectively, the validation process must be executed thoroughly. However, simulation modelling software often suffered from credibility challenges. Detecting errors early in the simulation modelling process is crucial for developing successful software. Therefore, validating simulation models is a highly important issue [11].

In all types of software development, determining whether the software complies with its requirements is a necessary step. In addition, the software must align with organizational goals and user requirements. As a result, many studies have investigated the definitions of validation and verification concepts, presenting a variety of critical perspectives. For example, in 2001, [12] suggested that validation refers to ensuring the model's accuracy is sufficient to fulfill its functions under specified experimental conditions. This perspective was supported by [13], who emphasized that validation determines whether the simulation model represents the actual model with high accuracy and sufficiency. In 2002, [14] defined validation as the process of determining the extent to which a model matches the real system from a functional perspective. Nevertheless, in 2003 [15] proposed that the computerized model should be satisfactory in terms of accuracy relative to its intended application. By 2007, [16] described validation as comparing a simulation model's output with the real-world system.

More recent definitions include [17] in 2012, who highlighted that validation ensures cost-effectiveness and usability during the development process, while [18] introduced the idea of integrating validation directly with the model-building process. This integration allows validation to occur simultaneously with development, improving efficiency and usability. Additionally, in the same year, [19] emphasized evaluating whether the system or model meets specified requirements as part of the validation process. Lastly, in 2017, [20] defined validation as the process of proving that a product meets user expectations and ensuring that the system performs as specified.

The literature discusses various methods of validating simulation models, with authors describing these methods differently. Validation methods are generally classified as either subjective or objective. Objective methods include mathematical procedures or statistical tests, such as hypothesis testing. Subjective methods validate sub-models and the overall simulation model. Additionally, validation methods are categorized into informal, formal, symbolic, constraint-based, static and dynamic approaches. A range of methods has been detailed in contributions on simulation model validation methods provided by [21] – [26]. While previous studies have presented several classifications of simulation model validation, these classifications often overlap in concept, purpose or stage within the simulation process. Some previous studies provided classifications that were not clear, often based on the authors' interpretation while using similar terms for validation. Due to the significant number of validation methods presented in the literature, as well as their similarity and overlaps, distinguishing between these methods can be challenging. Commonly used validation methods for simulation models include comparison to models, animation, event validity, extreme condition, face validity, degenerate tests, historical data validation, historical methods, multistage validation, internal validity, operational graphics, parameter variability, traces, predictive validation, and Turing tests [27] - [40]. These methods are listed along with their explanation and purpose, as shown in Table 1.

Table 1
Validation Methods

Method	Reference	Explanation	Validation Method Purpose	
			Structural	Behavioral
Animation	[27]	Determine the model's behavior by use graphs to through time		✓
Comparison To Models	[28]	Comparing the model's results with other valid models' results		✓
Degenerate Tests	[29]	Testing the behavior of model by using a defined value		✓
Event Validity	[30]	Comparing the model events to real system in order to determine the similarities		✓
Extreme Condition	[31]	Checking validity of model in maximum and implausible levels of the system		✓
Face Validity	[32]	Take the opinion of expert persons on the reasonableness of the model		✓
Historical Data Validation	[33]	Part of validation data uses to create models, and the other of validation data for testing	✓	
Historical Methods	[33]	Using three methods is empiricism, rationalism, and positive economics	✓	
Internal Validity	[34]	Identifying of the model's variability by executing some of model functions	✓	
Multistage Validation	[35]	Integration three historical methods to a high level of the validation process		✓
Operational Graphics	[36]	Displaying various performance measures values		✓
Parameter Variability	[37]	Using sensitivity validation to identify the parameters' effect		✓
Predictive Validation	[38]	Check the model's prediction with the system behavior		✓
Traces	[39]	Trace the model's entities to determine the model logic is correct		✓
Turing Tests	[40]	Ask persons to discriminate the outputs of the model and real system		✓

Based on the above discussion of how literature defines the validation process and relevant concepts, this paper observes differences in interpretations used within the same context and terminology, which have been addressed by varied descriptions. The main issue is not only the differences in terminology, but also the confusion and vagueness raised and discussed in the literature. Therefore, this paper defines modelling validation as a synchronize assessment process aligned with the modelling simulation process, encompassing all model stages, including design, implementation, and testing. This definition is based on the proposed architecture for model validation, which explains the validation process in parallel with each phase of the required modelling simulation process.

Additionally, the validation methods of simulation modelling play a crucial role by providing an important validation framework, as highlighted in the literature. However, the large number of methods proposed by researchers pose a significant challenge in identifying and selecting suitable validation methods. Furthermore, there is a lack of validation methods tailored for every simulation. In other words, not all validation methods are sufficient for every simulation model. Moreover, the literature lacks a clear classification to describe simulation validation methods. Consequently, this paper aims to propose an integrated architecture for the Simulation Modeling Validation Process (SMVP) and suggest a new classification of validation methods.

In the literature, many suggested approaches interpret validation process stages. Some of these approaches adopt a simple perspective, while others take a more complex view. For example, [41] presented a study incorporating both perspectives and concluded that the simple approach to model validation is clear and unambiguous. Additionally, [42], [43] offered paradigms for each approach. On the other hand, [44] advocated for a complex perspective of the validation process, viewing it not as a distinct stage or set of steps in the model simulation life cycle but as a continuous activity throughout the entire software lifecycle. A diagram of validation activities for the simulation model was also presented. Meanwhile, [45] proposed a different paradigm for simulation model validation through the modelling process, adapting the simple perspective with additional details regarding white-box and black-box validation.

Building on these observations, many researchers have attempted to describe the stages of model simulation validation and how these are performed from multiple perspectives. However, similarities exist in the descriptions of their approaches and architecture. Generally, each proposed architecture identifies a continuous sequence of steps (e.g., phases, activities or processes). It can be concluded that there is a lack of consistency between the diverse validation phases. Therefore, this paper suggests an integrated architecture for SMVP, including a validation process parallel to the simulation model process, as discussed in the next section.

II. PROPOSED METHOD

This paper proposes an SMVP architecture adapted from [42] - [44]. Figure 1 illustrates how the validation process is synchronized with the modeling simulation process. It emphasizes that every stage of the modeling simulation process, including analysis, design, implementation, and testing, must occur in parallel with the validation process. Furthermore, it highlights that various simulation models and validation activities are available.

The first step in initiating a System Problems Definition (SPD) is to understand the real-world system problem that needs to be addressed. Next, a model is developed as a Conceptual Model (CM) through activity analysis and modeling, such as using an activity diagram. Then, a Representation Model (RM) is created based on specification activity, such as formal specification, followed by the computerized model (CoM) through programming. Subsequently, an Experimental Model (EM) is constructed to conduct experiments, such as test-case design. The final step in developing a Simulation Model (SM) involves conducting experiments, such as test-case execution.

The validation process includes Conceptual Model validation, Representation Validation, Experimental Validation, Behavioral Validation, and finally, Data Validation. Without validation data, the simulation model cannot be developed and utilized. As shown in Figure 1, the proposed SMVP architecture used various models and phases.

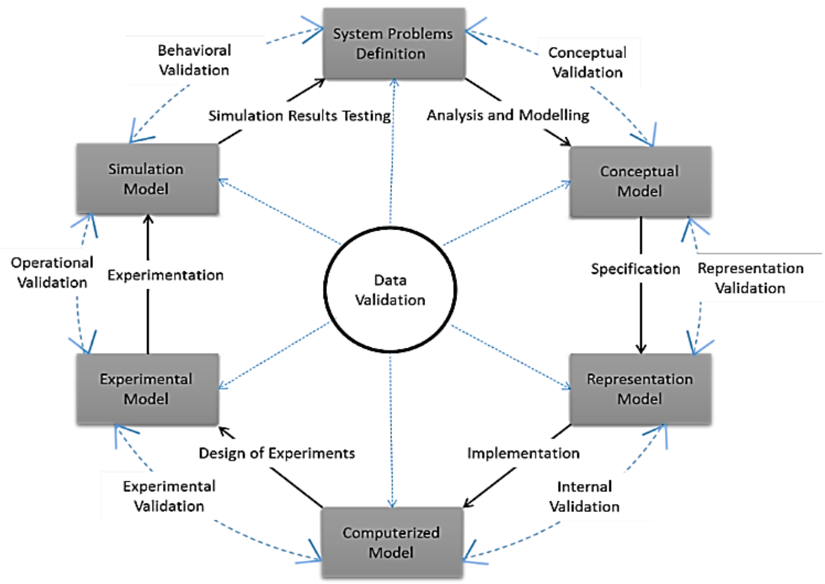

Figure 1. The proposed architecture for SMVP

A. System Problems Definition (SPD)

The SPD refers to a real system or proposed requirements, situations, ideas, or policies for modeling. Problem analysis is based on accurate definition, and the purpose of the model is determined by the objectives of the organization. Furthermore, it describes the ideas and phenomena that are being simulated. The assumptions used in the project have been defined and formalized [42].

B. Conceptual Model (CM)

The CM is the mathematical, logical and graphic representation of the SPD developed for a particular study. It provides a comprehensive and unambiguous representation of the model. Modelling requirements are transformed into detailed simulation specifications related to the simulation model. Documenting simulation-neutral views of important entities and their key actions and interactions is critical to achieving agreement between the simulation developer and the user. It is a non-technical description of the simulation model to be developed, describing the model's objectives, inputs, outputs, content, assumptions and simplifications. The CM is typically developed during the design and modeling phase. Workflow modeling, workforce modeling, object-role modeling, and the Unified Modeling Language (UML) are common conceptual modeling techniques and methods used [46].

C. Conceptual Validation (CV)

The CV ensures that the theories, hypotheses and scope are adequate for the SPD and the CM, and verifies that all hypotheses are correct. It helps identify whether the model's definition is complete and capable of answering the proposed questions. It also determines whether the CM's assumptions are accurate, along with the SPD representation, model structure, and behavior. If the CM represents logical, mathematical, and causal relationships, then the goals are deemed reasonable.

The formal representation of the model is crucial for validating the Systemic Structural and Simplification assumptions during CM validation. The CM represents the structural relationships between different model elements and the behaviors composing the model. Defined techniques are necessary to ensure that systemic structural assumptions align with the purpose and that simplification hypotheses do not oversimplify the reality they represent. An analysis of the CM's behavior before coding can be performed using several formal languages, such as Specification and Description Language (SDL), Petri Nets, and Discrete Event System Specification (DEVS). Furthermore, this facilitates the integration of different conceptual models that represent the same system or its components.

To determine whether the model's intended purpose is met, each sub-model and the overall model must be evaluated. This evaluation involves determining whether appropriate details and aggregate relationships have been used for the purpose of the model. Face validation involves experts from the SPD evaluating the CM to assess its correctness and reasonableness for its purpose. Experts are required to examine the flowchart or the graphical model and the set of equations of the model to ensure logical accuracy. Traces are used to track entities through each sub-model to maintain overall model accuracy. If errors are identified in the CM, it must be reviewed, and the validation process for the CM repeated [43].

D. Representation Model (RM)

The RM represents the CM and is often similar to an algorithm. It describes the written target phenomenon or theory. To implement this model as a pre-CoM, intermediary models such as text programs written in high-level programming languages are reviewed. At a later stage, one or more CoMs are designated for execution as computerized models based on RM execution. The RM is created during the specification phase.

By integrating the CM and the RM, the gap between informal and formal methods is bridged. The system's requirements are improved during the analysis and design phases through the obtained results, reducing ambiguity and errors, while improving quality and confidentiality [47].

E. Representation Validation (RV)

The RV validates the RM by observing execution behavior and output data, typically using graphing, visualization, and statistical packages. Descriptions that were not included in the representation model are incorporated during this process. Verification of interactions between objects specified at some level of description is conducted by verifying data flows, control flows, and syntax. This ensures that all the objects associated with the CM and the RM are valid. For formal verification, formal languages such as VDM, B-Methods, Larch, and Z notation are used. Z notation is an example of a formal specification language that uses set theory to evaluate the behavior of sequential systems [48].

F. Computerized Model (CoM)

The CoM refers to the computer programming and execution of the RM. RM algorithms are implemented to create the CoM. The simulation that encodes the system model is called the CoM. Using higher-level programming languages, such as FORTRAN, C, or C++, can influence simulation verification. The process is repeated to refine and verify the CoM. A computer programming and implementation phase is used to develop the CoM [49].

G. Internal Validation (IV)

The IV ensures that the computer programming and implementation are accurate and that the model executes correctly. It is important to confirm that the CM has been programmed and implemented correctly and that it executes properly in the computer language used to perform CoM verification. The software must be error-free, properly implemented on the computer, and tested for correctness.

Software engineering techniques are employed to design, develop, and implement the computer program, whether it uses a higher-level programming language or a simple simulation language. These techniques include object-oriented design, structured programming, and program modularity. Verification is performed to ensure that the simulation model's execution functions correctly. Two basic approaches are employed for verifying simulation software and simulation models: static testing and dynamic testing. Static testing examines the computer program's structure using techniques such as structured walkthroughs and correctness proofs. These techniques evaluate the structural properties of the program to determine its correctness. Dynamic testing, on the other hand, involves executing the computer program under different conditions and analyzing the outcomes. Both the simulation software and the CoM are examined to ensure they operate correctly and meet their intended purposes [27].

H. Experimentation model (EM)

The EM consists of one or more test cases, designed to explore the impact of changes to input variables in the simulation model. To determine whether the requirements are met, experimentation scenarios are conducted using the CoM.

Testing techniques are categorized as black-box or white-box testing. Black-box testing, also known as functional testing, creates test cases based on specification details without requiring access the internal source code. These tests focus solely on the outputs generated in response to specific entries and performance conditions, ignoring the internal mechanisms of a system [50].

I. Experimental Validation (EV)

The EV evaluates whether the test procedures employed to obtain results are adequate. Validating the design and execution of experiments is crucial for conducting the EM based on an experimental framework. To ensure comprehensive coverage of the program units, the selection of test cases for EV is guided by the goal of achieving target adequacy. Determining appropriate validation techniques for a particular validation element is the equivalent of defining suitable test cases. Each element of the simulation can have specific validation techniques applied to it, and a set of techniques can be used for any element in the simulation. The most commonly used validation techniques for simulation models are discussed in the existing literature. These techniques include face validation using subject matter experts, input data validation through goodness-of-fit tests, sensitivity analysis, extreme condition tests, graphical animation, and comparison with other models [10].

J. Simulation Model (SM)

An entity or system can be represented or abstracted by the SM. A model is inherently imperfect because it is an abstraction. The aim is to create a credible simulation model and to obtain reliable simulation results. It is crucial to have reliable simulation results for both modelers and users of simulation models, as they are vital for making informed decisions. The construction of the SM is based on certain assumptions about the system being modeled. After constructing the model, it is subjected to experimentation. Although the programmed model's function may appear normal, if the model's assumptions are violated during experimentation, it may become invalid. The SM is utilized by the computer system as its inference model. The simulation model generates data and the results for the SM. Real-world data or problems serve as the basis for the inference model [51].

K. Operational Validation (OV)

The OV focuses on the outcomes obtained from implementing a model. To validate the entire behavior of the computer program that encodes the model, Black Box validation requires the use of all model assumptions. Operational validation aims to determine whether the simulation model's output behavior meets the specified accuracy requirements. Deficiencies found during operational validation can originate from the simulation model developed at any step. Development simulation models involve either formulating the system's theories or encountering invalid data. Various validation techniques are used for OV. The model development team is responsible for selecting the techniques and deciding their objective or subjective application. Operational validity is influenced by whether the SPD is observable, which means that data on the operational behaviour of the problematic entity can be collected [43].

L. Behavioral Validation (BV)

The behavior of the simulated model is evaluated by the BV based on the system observations. Confirmation is achieved only if the experimental data and model output are in agreement with an acceptable tolerance. BV is performed by comparing the model's output with the observed real dynamic behavior. The model must exhibit reasonable behavior under extreme conditions and be capable of handling uncertainties in initial conditions and parameters [44].

M. Data Validation (DV)

The DV ensures the accuracy of the data used in the model, including probability distributions, data sources, and support structures. To verify the data, systemic data assumptions must be evaluated. Tests focus on two key aspects: ensuring that the data complies with the system process and verifying that the data expiration restrictions are enforced. Validation of the required institutions (such as structures, enterprises, or data warehouses) involved in obtaining and maintaining the data is necessary to ensure the model's validity. The validity of simulation models depends on the definition of the data expiration constraint.

To construct a CM, sufficient information about the problem entity must be obtained. The aim is to develop theories that facilitate the construction of the model and to establish mathematical and logical connections that can be utilized in the model. This approach ensures that the model accurately represents the SPD for its intended purpose and verifies the assumptions underlying the model.

To compare the behavior of the problem entity with the model, the operational validity step requires the use of behavioral data related to the problem entity. These data primarily consist of system input/output data. Without sufficient operational validity, obtaining high confidence in the model is typically impossible in the absence of behavior data [43].

III. RESULTS AND DISCUSSION

Based on the discussion in previous sections, it can be concluded that there is not a clear approach in the literature for selecting validation methods. Selecting an appropriate method for SMVP depends on several important attributes. In other words, the critical issue lies in choosing the correct validation method during the correct validation phase. This section presents the proposed classification, which combines three critical attributes: the simulation validation phase, the simulation model, and the purpose of the validation method.

A. Validation Methods Classification

The proposed classification aims to facilitate the selection of a rigorous simulation validation method, as shown in Table 2. On the other hand, the proposed classification can be considered as a standard for determining the purpose of the validation method, whether structural or behavioral, to validate the model. Additionally, based on the suggested architecture (see Figure 1), the proposed classification identifies a clear relationship between the simulation validation phase, the simulation models, and the purpose of validation methods.

As shown in Table 2, the results of the proposed classification indicate that certain methods are more suitable for structural validation, even though most methods align with behavioral validation. It is important to emphasize

Historical Data Validation, Historical Methods, and Internal Validity, as their purpose is structural. The Internal Validity method, in particular, should be highlighted in every validation process since it examines the internal aspects of simulation modeling validation. However, once the structural validation of a model is sufficiently established, behavior validation is assessed to achieve the overall validity of the model or to build confidence in it.

The validation process involves a combination of structural validation and behavior validation. This implies that validation methods for both structural and behavioral types should be used within a single validation process. Furthermore, the comparison of validation methods during the term data validation phase indicates that face validity and predictive validation are excluded. All the methods in Table 2 validate the data validation phase among the most commonly used methods. Moreover, it is noteworthy that no single validation method covers all models and phases in the simulation validation process. Therefore, two or more methods must be combined to achieve effective validation. Additionally, the diverse objectives of structural and behavioral methods must also be considered.

Table 2
The proposed classification of validation methods

Validation Method	Reference	Simulation Modeling Validation Phases							Simulation Models						Validatio n Method Purpose	
		DV	CV	RV	IV	EV	OV	BV	SPD	CM	RM	CoM	EM	SM	Structural	Behaviora
Animation	[27]	✓					✓	✓	✓				✓	✓		✓
Comparison to other Models	[28]	✓				✓	✓	✓	✓			✓	✓	✓		✓
Degenerate Tests	[29]	✓						✓	✓					✓		✓
Event Validity	[30]	✓	✓				✓	✓	✓	✓			✓	✓		✓
Extreme Condition Test	[31]	✓					✓	✓	✓			✓	✓	✓		✓
Face Validity	[32]		✓	✓			✓	✓	✓	✓	✓		✓	✓		✓
Historical Data Validation	[33]	✓	✓			✓			✓	✓		✓	✓		✓	
Historical Methods	[33]	✓	✓	✓					✓	✓	✓				✓	
Internal Validity	[34]	✓		✓	✓					✓	✓	✓			✓	
Multistage Validation	[35]	✓				✓	✓	✓	✓			✓	✓	✓		✓
Operational Graphics	[36]	✓				✓	✓					✓	✓	✓		✓
Parameter Variability - Sensitivity Analysis	[37]	✓				✓	✓	✓	✓			✓	✓	✓		✓
Predictive Validation	[38]							✓	✓					✓		✓
Traces	[39]	✓	✓	✓			✓	✓	✓	✓	✓		✓	✓		✓
Turing Tests	[40]	✓				✓	✓	✓				✓	✓	✓		✓

B. Simulation Modeling Validation Phases

The simulation model validation phases represent the compatibility between the behavior of the real system with the corresponding elements of the simulation model, indicating whether the results are acceptable. If satisfactory agreement is not obtained, the model must be adjusted to bring it closer to the observed behavior of the real system, or any errors must be identified and rectified. Indeed, the validation phases of the simulation modeling process are not clearly specified in the literature. Therefore, this study suggests validation phases (see Figure 1) which start with conceptual validation, followed by representation validation, internal validation, experimental validation, operational validation, behavioral validation, and finally, data validation for all models.

C. Simulation Models

Simulation modeling involves creating and analyzing a model to predict its performance in the real world. It is used to help designers and engineers understand, predict, and apply simulation models effectively. Currently, the models involved in the simulation modeling process are not clearly specified in the literature. As a result, this paper proposed a new architecture (see Figure 1). The proposed architecture begins with an understanding of SPD, CM, RM, CoM, EM, and SM.

D. Validation Method's Purpose

The purpose of the validation method is to provide a stringent attribute to improve confidence in a simulation model. This paper suggests structural and behavioral validation methods, both of which have fundamental importance in the overall validation process. Structural validation confirms whether the model's structure aligns with the relevant descriptive information of the real system. On the other hand, behavioral validation evaluates how well the simulation model's behavior corresponds to the observed behavior of the real system.

Table 2 shows the required methods, which can be selected from the list of validation methods. Consequently, the validation of simulation modeling becomes more grounded, faster, more precise, and cost-effective. This table can be used by individuals or teams to select appropriate validation methods for a simulation model based on the main classification attributes. The first attribute is the simulation modeling validation phases, denoted by symbols (DV, CV, RV, IV, EV, OV, BV), which indicate the accessibility of the validation phase to the validation method. The second attribute is the simulation models, represented by symbols are the second attribute, and the symbols (SPD, CM, RM, CoM, EM, SM), indicating the types of simulation models that the validation method can access. The third attribute relates to the purpose of the validation method, which is either structural or behavioral.

IV. CONCLUSION

This paper proposed an integrated architecture for the Simulation Modeling Validation Process (SMVP). It emphasized the integration the model simulation validation process with the modeling simulation process in an organized, efficient and well-documented manner. Moreover, it proposed a classification that combined three attributes: simulation validation phase, simulation models, and validation method purpose, aimed at selecting the appropriate validation method. Consequently, the accuracy and correctness of the model simulation were obtained. The vagueness surrounding SMVP was resolved, the properties of the simulation modeling validation methods were identified, and the proposed classification of the validation methods was strengthened.

The proposed classification helps researchers perform accurate validation of the model. A model that has not been validated can potentially yield inaccurate results. Therefore, validation of a model holds utmost importance. The validation methods were researched, identified and compared based on the proposed classification within this study, which can guide the validation of models with a similar purposes and phases.

The main results from the proposed classification indicated that most validation methods focused on behavioral validation, while some, such as Historical Data Validation, Historical Methods, and Internal Validity, were suited for structural validation. Additionally, the Internal Validity method should be prioritized in every validation process because it examines the internal structure of the simulation modeling validation. This study also highlighted the importance of considering the diversity between structural and behavioral validation methods within a single validation process. Furthermore, the comparison of methods revealed that, with that exception of face validity and predictive validation, all methods supported data validation phases. More than two methods should be combined to achieve effective validation.

CONFLICT OF INTEREST

Authors declare that there is no conflict of interests regarding the publication of the paper.

AUTHOR CONTRIBUTION

The authors confirm contribution to the paper as follows: lead researcher and author of the manuscript: Zainab Hassan. Data collection and literature review: Dhafer Abdulameer. Study design and data analysis: Zainab Hassan. Methodology Development: Dhafer Abdulameer.

REFERENCES

[1] C. Kuemmel et al., “Consideration of a credibility assessment framework in model-informed drug development: potential application to physiologically-based pharmacokinetic modeling and simulation,” CPT: Pharmacometrics & Systems Pharmacology, vol. 9, no. 1, pp. 21-28, 2022, https://doi.org/10.1002/psp4.12479.

[2] S. E. Olamide, and T. M. Kaba, “Formal verification and validation of DEVS simulation models,” in Proc. of 2013 Africon, 2013, pp. 1-6, https://doi.org/10.1109/AFRCON.2013.6757720.

[3] A. M. Law, “How to build valid and credible simulation models,” in Proc. of 2022 Winter Simulation Conference (WSC), 2022, pp. 1283-1295, https://doi.org/10.1109/WSC40007.2019.9004789.

[4] J. L. Hicks, T. K. Uchida, A. Seth, A. Rajagopal, and S. L. Delp, “Is my model good enough? best practices for verification and validation of musculoskeletal models and simulations of movement,” Journal of Biomechanical Engineering, vol. 137, no. 2, 2015, https://doi.org/10.1115/1.4029304.

[5] D. J. Murray-Smith, “Testing and validation of computer simulation models,” Simulation Foundations, Methods and Applications, pp. 233-343, 2015.

[6] L. A. Franco, and G. Montibeller, “Facilitated modelling in operational research,” European Journal of Operational Research, vol. 205, no. 3, pp. 489–500, 2010, https://doi.org/10.1016/j.ejor.2009.09.030.

[7] H. Mori and K. Kise, "Design and Performance Evaluation of a Manycore Processor for Large FPGA," 2014 IEEE 8th International Symposium on Embedded Multicore/Manycore SoCs, Aizu-Wakamatsu, Japan, 2014, pp. 207-214, doi: 10.1109/MCSoC.2014.37.

[8] C. Cioffi-Revilla, “Simulations II: variable-oriented models,” in Introduction to Computational Social Science, Springer, Cham, 2017, pp. 415–454, https://doi.org/10.1007/978-3-319-50131-4_9.

[9] M. A. A. Faruque, and F. Ahourai, “A model-based design of cyber-physical energy systems,” in Proc. of 19th Asia and South Pacific Design Automation Conference (ASP-DAC), 2014, pp. 97-104, https://doi.org/10.1109/ASPDAC.2014.6742873.

[10] R. G. Sargent, “Verification and validation of simulation model,” in Proc. of the 2010 Winter Simulation Conference, Baltimore, 2010, pp. 166-183, https://doi.org/10.1109/WSC.2010.5679166.

[11] B. Roungas, S. Meijer, and A. Verbraeck, “A framework for simulation validation & verification method selection”, in Proc. of SIMUL 2017: 9th International Conference on Advances in System Simulation, 2017, pp. 35-40.

[12] R. G. Sargent, “Some approaches and paradigms for verifying and validating simulation models,” in Proc. of the 2001 Winter Simulation Conference (Cat. No.01CH37304), 2001, pp. 106-114, https://doi.org/10.1109/WSC.2001.977251.

[13] F. Giannanasi, P. Lovett, and A. N. Godwin, “Enhancing confidence in discrete event simulations,” Computers in Industry, vol. 44, no. 2, pp. 141-157, 2001, https://doi.org/10.1016/S0166-3615(00)00084-1.

[14] O. Balci, R. E. Nance, J. D. Arthur, and W. F. Ormsby, “Expanding our horizons in verification, validation, and accreditation research and practice,” in Proc. of the Winter Simulation Conference, 2002, pp. 653-663, https://doi.org/ 10.1109/WSC.2002.1172944.

[15] R. G. Sargent, “Verification and validation of simulation models”, in Proc. of the 2003 Winter Simulation Conference, 2003, pp. 27-48, https://doi.org/10.1109/WSC.2003.1261406.

[16] D. N. Moriasi et al., “Model evaluation guidelines for systematic quantification of accuracy in watershed simulations,” Transactions of

the ASABE, vol. 50, no. 3, pp. 885-900, 2007, https://doi.org/10.13031/2013.23153.

[17] H. Qudrat-Ullah, “On the validation of system dynamics type simulation models,” Telecommunication Systems, vol. 51, no. 2, pp. 159–166, 2011, https://doi.org/10.1007/s11235-011-9425-4.

[18] S. N. Groesser, and M. Schwaninger, “Contributions to model validation: hierarchy, process, and cessation,” Syst. Dyn. Rev., vol. 28, pp. 157–181, 2012, https://doi.org/10.1002/sdr.1466.

[19] Y. Singh, “Software Testing,” 1st Edition, India: Cambridge University Press, ISBN 978-1-107-01269-7, 2012.

[20] W. L. Lewis, “Software Testing and Continuous Quality Improvement”, 3rd Edition, CRC Press, 2016.

[21] A. M. Law, and D. W. Kelton, “Simulation Modeling and Analysis,” 6th Edition, McGraw-Hill, New York, 2024.

[22] R. Donà and B. Ciuffo, "Virtual Testing of Automated Driving Systems. A Survey on Validation Methods," in IEEE Access, vol. 10, pp. 24349-24367, 2022, doi: 10.1109/access.2022.3153722.

[23] H. Schmidt, K. Büttner and G. Prokop, "Methods for Virtual Validation of Automotive Powertrain Systems in Terms of Vehicle Drivability—A Systematic Literature Review," in IEEE Access, vol. 11, pp. 27043-27065, 2023, doi: 10.1109/access.2023.3257106.

[24] J. Friederich and S. Lazarova-Molnar, "A Framework for Validating Data-Driven Discrete-Event Simulation Models of Cyber-Physical Production Systems," 2023 Winter Simulation Conference (WSC), San Antonio, TX, USA, 2023, pp. 2860-2871, doi: 10.1109/WSC60868.2023.10407382.

[25] Laili, Yuanjun, L. Zhang, and Y. Luo, “A pattern-based validation method for the credibility evaluation of simulation models,” Simulation, vol. 96, no. 2, pp. 151-167, 2020, https://doi.org/10.1177/0037549719856100.

[26] Y. Ma, C. Sun, J. Chen, D. Cao, and L. Xiong, “Verification and Validation Methods for Decision-Making and Planning of Automated Vehicles: A Review,” IEEE Transactions on Intelligent Vehicles, 2020, vol. 7, no. 3, pp. 480-498, https://doi.org/10.1109/TIV.2022.3196396.

[27] R. G. Sargent, and O. Balci, “History of verification and validation of simulation models,” in Proc. of Winter Simulation Conference (WSC), 2017, pp. 292-307, https://doi.org/10.1109/WSC.2017.8247794.

[28] P. J. Durst, D. McInnis, J. Davis, and C. T. Goodin, “A novel framework for verification and validation of simulations of autonomous robots,” Simul. Model. Pract. Theory, vol. 117, 2022, https://doi.org/10.1016/j.simpat.2022.102515.

[29] G. Eremina, and A. Smolin, “Verification and validation of numerical models for the materials of the lumbar spine,” Procedia Structural Integrity, vol. 35, pp. 115–123, 2022, https://doi.org/10.1016/j.prostr.2021.12.055.

[30] R. Kapoor, and B. J. Shah “Simulation model for closed loop repairable parts inventory system with service level performance measures,” International Journal of Services and Operations Management, vol. 23, no. 1, pp. 18, 2016 https://doi.org/10.1504/ijsom.2016.073286.

[31] M. Bastan, R. R. Khorshid-Doust, S. D. Sisi, and A. Ahmadvand, “Sustainable development of agriculture: a system dynamics model,” Kybernetes, vol. 47, no. 1, pp. 142–162, 2018, https://doi.org/10.1108/k-01-2017-0003.

[32] M. M. Elarbi, C. A. Ragle, B. A. Fransson, and K. D. Farnsworth, “Face, construct, and concurrent validity of a simulation model for laparoscopic ovariectomy in standing horses,” Journal of the American Veterinary Medical Association, vol. 253, no. 1, pp. 92–100, 2018, https://doi.org/10.2460/javma.253.1.92.

[33] C. Yin, and A. McKay, “Model Verification and Validation Strategies and Methods: An Application Case Study,” in Proc. of 8th International Symposium on Computational Intelligence and Industrial Applications and the 12th China-Japan International Workshop on Information Technology and Control Applications (ISCIIA 2018), 2018, pp. 1–6.

[34] A. Hung et al., “Validating a budget impact model using payer insight and claims data: a framework and case study,” Journal of Managed Care & Specialty Pharmacy, vol. 25, no. 8, pp. 913–921, 2019, https://doi.org/10.18553/jmcp.2019.25.8.913.

[35] L. Khakimova, T. Bondarenko, A. Cheremisin, A. Myasnikov, and M. Varfolomeev, “High pressure air injection kinetic model for bazhenov shale formation based on a set of oxidation studies,” Journal of Petroleum Science and Engineering, vol. 172, pp. 1120–1132, 2019, https://doi.org/10.1016/j.petrol.2018.09.021.

[36] W. P. Ferreira, F. Armellini, L. A. Santa-Eulalia, and V. Thomasset-Laperrière, “Extending the lean value stream mapping to the context of industry 4.0: an agent-based technology approach,” Journal of Manufacturing Systems, vol. 63, pp. 1–14, 2022, https://doi.org/doi:10.1016/j.jmsy.2022.

[37] L. Jia, B. Zhang, and Y. Song, “Evaluation of process simulation model based on a multi-level test case method,” in Proc. of Theory, Methodology, Tools and Applications for Modeling and Simulation of Complex Systems, 2016, pp. 420-431, https://doi.org/10.1007/978-981-10-2672-0_43.

[38] T. V. Porgo et al., “The use of mathematical modeling studies for evidence synthesis and guideline development: a glossary,” Research Synthesis Methods, vol. 10, no. 1, pp. 125–133, 2021, https://doi.org/ doi:10.1002/jrsm.1333.

[39] R. Gore, C. J. Lynch, and H. Kavak, “Applying Statistical Debugging for Enhanced Trace Validation of Agent-Based Models,” SIMULATION, vol. 93, no. 4, pp. 273–284, 2016, https://doi.org/10.1177/0037549716659707.

[40] A. L. Nsakanda, G. Grant, M. Vafei, and M. Leafloor, “A simulation modeling approach to understanding workflow changes in healthcare: the case of CPOE deployment at the Ottawa hospital,” in Proc. of 48th Hawaii International Conference on System Sciences, pp. 2933-2941, 2015, https://doi.org/10.1109/HICSS.2015.355.

[41] J. Banks, D. Gerstein, and S. P. Searles, “Modeling processes, validation, and verification of complex simulations: a survey,” Methodology and Validation, Simulation Series, vol. 19, no. 1, pp. 13-18, 1988.

[42] R. G. Sargent, “An assessment procedure and a set of criteria for use in the evaluation of computerized models and computer-based modeling tools,” Final Technical Report RADC-TR-80–409, 1981.

[43] R. G. Sargent, “Verification and validation of simulation models,” in Proc. of Winter Simulation Conference, 2005, pp. 43–130.

[44] O. Balci, “Verification, validation, and accreditation,” in Proc. of 1998 Winter Simulation Conference (Cat. No.98CH36274), 1998, pp. 41-48. https://doi.org/10.1109/WSC.1998.744897.

[45] S. Robinson, “Simulation model verification and validation: increasing the users’ confidence,” in Proc. of the 29th conference on Winter simulation, 1997, pp. 53–59. https://doi.org/10.1109/WSC.1997.640377.

[46] I. Davies, P. Green, M. Rosemann, M. Indulska, and S. Gallo, “How do practitioners use conceptual modeling in practice?,” Data & Knowledge Engineering, vol. 58, no. 3, pp. 358-380, 2006.

[47] Z. H. Muhamad, D. A. Abdulmonim, and B. Alathari, “An integration of UML use case diagram and activity diagram with Z language for formalization of library management system,” International Journal of Electrical & Computer Engineering, vol. 9, no. 4, pp. 8708, 2019.

[48] N. David, “Validation and verification in social simulation: Patterns and clarification of terminology,” in Proc. of International Workshop on Epistemological Aspects of Computer Simulation in the Social Sciences, 2006, pp. 117-129.

[49] P. W. Kenneally, S. Piggott, and H. Schaub, “Basilisk: a flexible, scalable and modular astrodynamics simulation framework,” Journal of Aerospace Information Systems, vol. 17, no. 9, pp. 496-507, 2020.

[50] A. Z. Javed, P. A. Strooper, and G. N. Watson, “Automated generation of test cases using model-driven architecture” in Proc. of 2nd International Workshop on Automation of Software Test (AST'07), 2007.

[51] Y. Wang, M. K. Traoré, G. Zacharewicz, and D. Chen, “Verification and validation of D2FD method,” in Proc. of NICST'2017: New Information Communication Sciences and Technology for Sustainable Development, 2017.