Design and Development of a Power System Digital Twin: A Model-based Approach

Serge P. Kovalyov
V. A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences
Moscow, Russia
ORCID: 0000-0001-5707-5730

***Abstract*—The paper presents an approach to facilitate integration and coherent operation of numerous disparate models that constitute a power system digital twin. We employ contemporary information technologies empowered by model-based systems engineering and advanced algebraic techniques to make the digital twin cost-effective to compose and operate while maintaining sufficient completeness, accuracy, and usability for practical purposes. The model-based digital twin architecture is proposed, which consists of an ontology, digital diagrams and visuals, electronic documentation, master data, real-time data, and mathematical models. The energy domain ontology provides the unified semantic basis for interaction between all the models. The mathematical framework of category theory applies for rigorous formal description and verification of complex digital twin composition procedures.**

Keywords—power system, digital twin, model-based systems engineering, ontology, category theory

I. Introduction

A Digital Twin (DT) is a dynamic digital representation of a physical asset (a system, a facility, a process, or a service) that is linked to the asset via models, information, and data and alters its properties, status, and behavior in real time [1]. DT intensely employs Industry 4.0 data-driven artificial intelligence technologies and simulations for adaptive monitoring, forecasting, optimization, and controlling asset performance [2]. Nowadays, many DT-based applications have been successfully implemented in different industries and DT becomes an emerging market. Gartner has identified DT as one of the Top 10 Strategic Technology Trends.

The DT is most accurate when initiated at the early stages of the asset lifecycle and developed incrementally along with the asset [3]. Even at the design stage, long before the asset comes into existence, the early DT is much useful as a virtual testbed to evaluate and compare alternative design decisions. It facilitates the automatic search for an optimal design using artificial intelligence tools and technologies known as generative design [4]. The key steps of such technologies include deriving a certain multi-objective optimization problem from stakeholder requirements for a product, automatic search for the best solution to this problem, and subsequent physical implementation of the solution through automatic manufacturing. To date, generative design has been successfully applied to dependable parts in the machinery industry, using topological optimization and 3D printing.

For the power industry, DT solutions are available aimed to implement the generic concept shown in Fig. 1 [5]. DT-based power system control applications show promising results in load/generation forecasting, power flow analysis, optimal energy storage scheduling, fault diagnosis, electrical machine health monitoring, and other relevant areas [6]. Yet, these applications are difficult to scale to real-world power grids and cost-effectively adapt to everyday operation and maintenance. A full-fledged grid DT requires high-end IT resources and a lot of skilled labor to set up: it has to promptly follow long-running multi-step processes involving numerous widely distributed disparate facilities and affected by hidden technical, social, and economic factors. For similar reasons, generative design technologies are much less productive in the power industry than in mechanical engineering. The DT is particularly hard to deploy in emerging decentralized systems based on distributed energy resources (DER), such as a microgrid consisting of numerous small low voltage (0.4 kV) prosumers that lack both a significant IT budget and a highly qualified staff [7].

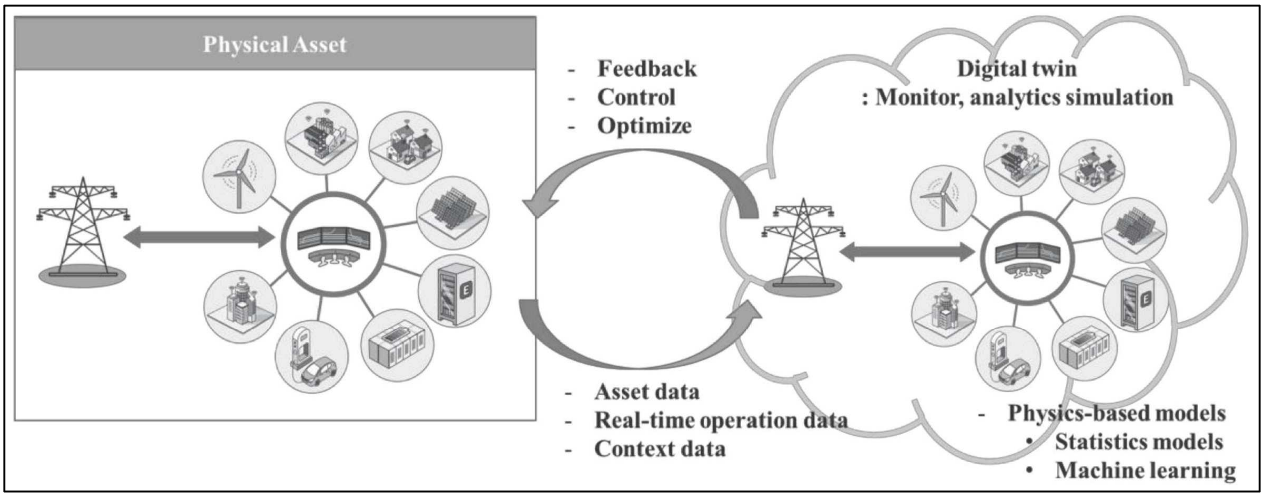

Fig. 1. The power system DT concept [5].

2021 3rd International Conference on Control Systems, Mathematical Modeling, Automation and Energy Efficiency (SUMMA) | 978-1-6654-3981-7/21/$31.00 ©2021 IEEE | DOI: 10.1109/SUMMA53307.2021.9632191

978-1-6654-3981-7/21/$31.00 ©2021 IEEE

Major difficulties stem from the complexity and diversity of models that have to be included in the DT to describe the power system properly [8]. Models differ from each other both horizontally, by parts/units they describe, and vertically, by concerns/viewpoints they represent. The integration and coherent operation of numerous disparate models tend to consume a huge amount of computer and human resources [9]. In essence, the DT composition process has to reproduce the asset construction process digitally. So, DT development activities belong to the problem domain of model-based systems engineering (MBSE) [10].

The paper presents a model-based approach to facilitate DT design and development for power systems. We demonstrate how contemporary information technologies empowered by advanced algebraic techniques make DT cost-effective to compose and operate while maintaining sufficient completeness, accuracy, and usability for practical purposes. In particular, interaction between the models is based on the energy domain ontology to improve interoperability.

II. Digital Twin Architecture

A. Power System Digital Twin Architecture Overview

Different approaches to the DT architectural decomposition exist [11]. For example, a layered architecture similar to cyber-physical systems is proposed, with the physical asset on the lowest layer, intelligent control services on the upper layer, and various mediating components in-between. Alternatively, the functional decomposition applies to break the DT down to data storage, information exchange, calculation, visualization, and other modules. In contrast, the MBSE perspective suggests decomposing by models that differ in kind and viewpoint, to highlight the specifics of DT among other digital systems. The typical power system DT consists of models of the following kinds [3]:

- ontology model;
- digital diagrams and visuals;
- electronic documentation;
- information model;
- real-time data;
- mathematical and simulation models.

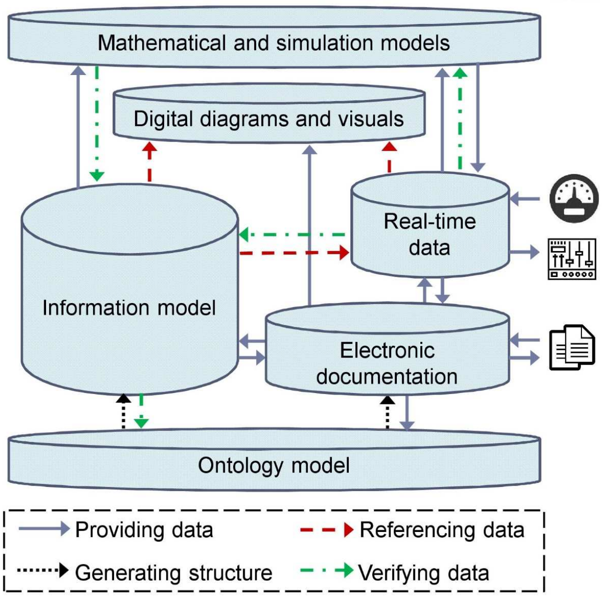

Fig. 2. The power system DT architecture.

To provide convenient access to the models, they are often designed as (micro)services [12]. The service-oriented architecture hides details of the models' implementation, which is particularly useful for controlling the decentralized DER systems. During the power system operation, the models interact intensively with external data sources and recipients, as well as with each other. Model interaction use cases include data exchange, data referencing, cross-model data verification, and data structure generation, as shown in Fig. 2. In particular, different models may contain information about the same physical thing obtained by different means, which is useful for verification.

B. Ontology Model

A domain ontology is a widely recognized tool to improve interoperability between disparate models. An ontology describes the logical structure of the domain concepts, including their definitions, properties, and relationships with each other. The ontology is often represented as a taxonomy tree of concepts that are endowed with properties of various types and associated with various relationships (equivalence, proximity, partonomy, etc.). In a machine-readable form, the ontology is written on the OWL (Ontology Web Language) using various tools such as the interactive editor Protégé.

For power systems, major ontological concepts describe physical parts of the grid: equipment units, line segments, sensors, control and communication devices, etc. Physical and economical parameters constitute other categories of concepts. Auxiliary concepts, relations, and axioms are introduced to group grid parts by different aspects: by type/manufacturer, by location, by ownership/rent, by functional features, by interchangeability/replacements, and so on.

The main use cases for the ontology model as a DT component are:

- information model structure generation;
- reference data maintenance;
- communication protocols design;
- development of service quality models;
- development of document templates and user interface forms;
- axiomatic inference and verification of design decisions;
- learning the subject domain.

C. Digital Diagrams and Visuals

In two dimensions, a diagram is a display that symbolically represents relationships between certain objects: spatial (a map or a layout), causal (a flow chart), partonomical (a product structure graph), and others. In 3D, a geometric model displays a physical asset virtually as if it were real. A diagram in a digital form can be very large yet graspable due to zooming and browsing capabilities. Numerous tools support the creation and displaying diagrams and visuals, from a desktop graphical editor to a CAD system to a planet-scale GIS. To facilitate diagrammatic description of complex systems in every aspect, certain tools are promoted by MBSE, such as SysML. In an information-rich environment, a diagram can be made interactive by associating symbols with data that describe their denotata. Further, augmented reality technologies allow displaying symbols and data right near the relevant parts of a real physical asset viewed through a digital camera.

For power systems, digital diagrams and visuals of several types are in use, such as a single line diagram, an electrical plan of a building (2D or 3D), a transmission lines map, a maintenance workflow model. Their major use cases as DT parts include:

- compact user-friendly visual presentation of the power system structure and state as a symbolical image;
- (soft) real-time indication of places where important events occur;
- search for information about system parts by topological criteria wrt displayed relations.

D. Electronic Documentation

A document is a uniquely identified self-contained unit of semi-structured information readable by a human or a machine. For persistent storage, documents are placed into an archive organized according to a hierarchy of folders that reflect the documents' purposes and contents. A set of metadata is associated with each document. Various standards apply to define the metadata structure depending on the subject domain, with possible references to the domain ontology.

Electronic documents are routinely accepted as input and generated as output by different information processing tools: from desktop editors to CAD and ERP systems. Integrating such tools with a DT allows increasing its accuracy, integrity, and usability. Specifically, the following use cases are relevant for electronic documentation as a part of a power system DT:

- presentation of information about the power system in a self-contained semi-structured human-readable form;
- identification and verification of reasons and permissions to execute control actions;
- extraction of ontological concepts, relationships, and axioms;
- designing digital diagrams and visuals;
- updating and verification of the information model;
- updating and verification of real-time data;
- search for documents by context and metadata;
- collecting the documentation archive.

E. Information Model

An information model represents the facility structure and relatively static properties in a machine-readable form of entities, attributes, and relationships. The information model contents are often referred to as master data since it provides context for facility business transactions and operation.

The most convenient storage for the information model is a relational database (RDB). With all its advantages, RDB has a significant drawback: it is not well suited to keep the history of changes in attribute values and relationships. One way to remedy the drawback is to store the full information model changes history in a machine-readable event log and keep only the up-to-date master data in the RDB. This approach fits into an architectural pattern called CQRS (Command Query Responsibility Segregation) [13]. It allows to accurately reproduce the description of any system part at any moment of time, facilitating the DT-based analysis of a large power system on a per-subsystem basis.

The main use cases for the information model as a power system DT component are:

- supplying mathematical and simulation models with master data;
- fine-grained binding of real-time data to the system parts;
- assigning unique identifiers to parts;
- part-wise data access control;
- collecting the system changes history during its lifecycle;
- verifying master data completeness and consistency by comparing with the factual system structure;
- attribute-based search for information about parts;
- generation and verification of the system design and construction documentation;
- visualization of the system structure and parts characteristics as tables and hierarchy trees.

F. Real-time Data

Data from physical or virtual digital sensors arrive at a DT in soft real time, usually through the Internet of Things (IoT). The data include measurement results of physical quantities, event logs, audio and video streams, economic indicators, and so on. Each data value is endowed with the metadata, including the data type, timestamp, and source. Upon arrival at DT, the real-time data is immediately placed to the fast storage, such as a time-series database (TSDB), and then delivered to other components for processing.

For a power system, as well as for any technological asset, the DT employs real-time data according to the following major use cases:

- assessing the state of the system as a whole as well as its parts;
- detecting out-of-range values of the system state variables, especially emergency indicators;
- feeding mathematical and simulation models with input data;
- collecting the observational history of the system operation in different conditions;
- search for patterns in the system's behavior;
- structural and parametric identification and verification of the information model;
- verification of mathematical and simulation models, including retro-forecasting;
- generation and verification of operation reports;
- visualization of state variables as tables, graphs, diagrams, and histograms.

G. Mathematical and Simulation Models

Mathematical (in a broad sense) and simulation models directly implement the DT purpose. Currently, there are two fundamentally different approaches to construct such models: numerical "first-principles" description of real-world (physical, biological, economical, and so on) phenomena and machine learning on historical data. Numerical models based on solving discretized algebraic, differential, and stochastic equations have a long development history as a part of CAD called Computer Aided Engineering (CAE). They implement the concepts invented a few decades ago, long before the advent of DT. Traditionally, CAE tools were intended for use in a design office, processing data manually entered by the engineer under very weak time constraints. Recently, high-performance computers became widely available to execute CAE algorithms in soft real time along with the asset operation, processing data that arrive directly from sensors. This allows using CAE in a DT: for example, there exists a thermoelectric model of a power transformer capable to predict failures [14].

In turn, the machine learning-based models are relevant for processes governed by unknown rules (usually going beyond physics) and/or containing hidden volatile patterns. Such processes are inherent in the DER equipment: for example, deep (multilayer) artificial neural networks are used to predict the productivity of generators on renewables [15]. The machine learning model accuracy crucially depends on the statistical quality of a training sample, viz. a set of asset behavior scenarios observed with different values of all influencing factors (features). In particular, deep learning models are unreliable at predicting the evolution of emergencies and disasters whose observational history is limited. Hence, it is entirely reasonable to equip the DT with several fundamentally different models of the same phenomena to verify each other.

The main use cases for power system mathematical and simulation models, and hence for the DT in general, are:

- evaluation and forecasting the generation, consumption, and storing of electrical energy;
- evaluation and forecasting the grid segments throughput;
- determining, optimizing, testing, and applying operation modes, commands, and control devices settings;
- predictive equipment health monitoring, assessment of failure rate and maintenance needs;
- calibration and verification of models and control algorithms;
- validation and evaluation of design decisions;
- comprehensive virtual training of the energy facilities personnel.

III. Digital Twin Development

The proposed model-based DT architecture suggests starting its development process with the selection/composition of an appropriate ontology. For traditional large energy systems, such as power plants, transmission lines, and substations, relevant ontology terms and relations are available in the well-known common information model (CIM) standardized in IEC 61968, 61970, and 62325. For DER and prosumer devices, auxiliary ontologies exist, such as "SmArt eneRGy dOmain oNtology" (SARGON) [16]. The resulting ontology is (semi)automatically transformed to the information model structure: ontological concepts are turned into tables, properties into attributes, relationships into foreign keys, axioms into content verification rules [17].

Then, given a facility, its information model is populated by the master data along with electronic documentation and digital diagrams. If the facility has already been built and put into operation, the data are collected from the existing documentation and verified by a field survey. Otherwise, CAD and similar tools are used to form the data during the facility design and construction. Moreover, generative design technologies are applicable at this stage to automatically find the best power system configuration. However, mechanical engineering algorithms and tools aimed to generate an optimal product shape are useless to design a complex energy system, for which the computer should generate and evaluate alternative topologies, equipment types, operation modes, and control algorithms. The search for (sub, Pareto)optimal alternatives is performed in the heterogeneous design space that contains not only numerical variables, but also discrete non-numerical variables taking values as grid graphs, database records, control algorithm charts, and so on. Novel intelligent discrete optimization algorithms are needed to efficiently find topological and configuration solutions in such spaces [18]. We propose an algebraic basis for such algorithms in the next Section.

As soon as the information model is ready, mathematical and simulation models are deployed, configured, and tested. Machine learning models are trained on available reference and/or historical datasets. Then (soft) real-time data sources and recipients are connected to the DT, and the latter commences to operate, subject to permanent inter-model verification and enhancement as imposed by the architecture.

The process outlined above has been tested on a small-scale prosumer facility with DERs (a solar panel, a wind turbine, a battery, and a diesel generator) backed up by a connection to a conventional distribution grid [3]. The information model was composed using the special-purpose software tool called Nrjpack that supports the model-based DT architecture implementation. A generative design approach to determine a (sub)optimal DER configuration was tested by integrating Nrjpack with Homer PRO microgrid techno-economic modeling and optimization tool. Soft real-time communication with meters and DER controllers installed at the facility was set up. The comprehensive facility simulation model was created in Matlab Simulink and applied to design and control multi-step processes such as a demand response cycle.

IV. Algebraic Approach to Digital Twin Composition

To automate the DT composition, especially when employing the generative design, a rigorous formal description and verification technique is much desired. A promising approach to develop such a technique is based on the mathematical framework of category theory [19] [20] [21]. Category theory allows to separate compositionality from other aspects of models by shifting the representation focus from their internal contents to behavior wrt each other (a systems engineer's "black box" point of view). Denote by C a category with all possible models of a certain kind as objects. Its morphisms describe, in terms of the models, all actions possibly performed when composing systems from components. Depending on the kind of models, actions may include variable substitution, reference resolution, submodel inclusion, quotienting, and other model construction primitives. It is easy to see that C is indeed a category since it has composition of morphisms (sequential execution of actions) and identity morphisms (idle "doing nothing" action with any model). A complex system configuration involving multiple models is represented as a diagram in C. To describe heterogeneous systems such as a DT that contain models of different kinds, suitable unification functors between model categories are applied to the diagram vertices [22].

For information models, the categorical representation is known [23]. Here, objects are sets of typed data that populate tables in a database, and morphisms describe referential constraints, viz. maps between tables that connect data sets with complex structure. Hence, all possible information models pertaining to a certain domain, such as energy infrastructure, constitute a subcategory of the well-known category **Set** of all sets and maps. A diagram in such a category shows how to compose a database.

Yet, mathematical models, being the most important among DT components, are the most technically difficult to represent categorically [24]. In particular, a categorical representation of DERs as finite automata with states labeled by power demand regions is proposed to formalize DER aggregation and solve a power flow problem [25]. For a less involved yet apt example, consider discrete-event simulation. Here, a model specifies an operational scenario, viz. a fragment of the imagined history of the asset behavior described as a sequence of discrete events of various types. Descriptions of actions used to build complex scenarios specify the contribution of the components behavior. Thus, the asset scenario is composed of components scenarios interacting with each other during the asset operation. Algebraically, a scenario is represented by a set of events partially ordered by causal dependencies and labeled by event types; scenarios assembling actions are defined as maps that preserve the order and the labeling [26]. All scenarios and actions constitute a category denoted as **Pomset**, diagrams in which represent complex scenarios composition.

In an arbitrary category C, the universal construction called colimit is rigorously defined. Given a diagram, its colimit, if exists, transforms it into a holistic complex model whose structure the diagram represents. Conceptually, a colimit expresses in algebraic terms the common-sense view of a system as a "container" that includes all parts, respecting their structural interconnections, and nothing else.

Consider the simplest non-trivial colimit, called a pushout, for a span-shaped diagram $f : P \leftarrow G \rightarrow S : g$. This diagram can be interpreted as the structure of a system composed from parts P and S joined by a "glue" G. Various mediators that integrate disparate data volumes, models, or whole systems are representable as glues [27]. In information model design, a "many-to-many" relation between two tables can act as a glue, such as the ownership/rent relation between legal entities and equipment units. In discrete-event simulation, a glue, for

$$\begin{tikzcd}
G \arrow[r, "g"] \arrow[d, "f"'] & S \arrow[d, dashed, "s"] \arrow[ddr] & \\
P \arrow[r, dashed, "p"] \arrow[drr] & R \arrow[dr, dotted] & \\
 & & T
\end{tikzcd}$$

Fig. 3. The pushout.

example, represents a control system that establishes a (sub)optimal coherent operation of two equipment units, e.g. a wind turbine and a storage device. In general, the target integral model constructed via gluing should include both parts and respect the glue in the sense that tracing the glue inclusion through either one of two parts amounts to the same action. Moreover, the target model should contain nothing except two glued parts, i.e. it should be unambiguously identified within any arbitrary model that contains the parts and respects the glue. Fig. 3 shows these criteria as a commutative diagram where the vertex R denotes the target model, the edges $p : P \rightarrow R$ and $s : S \rightarrow R$ denote inclusions of P and S into it, respectively, and T denotes an arbitrary enclosing model. It is easy to verify that the object R, provided that it exists, is defined uniquely up to an isomorphism. Under mild technical conditions, a colimit of any finite diagram can be calculated by a sequence of pushouts.

There exists a natural way to construct morphisms between diagrams in C from morphisms of C and homomorphisms of graphs that specify diagram shapes (an instance of the so-called Grothendieck flattening construction). This construction defines the endofunctor **D** of the "category of all categories" **CAT**, which sends the category C to the category of diagrams $\mathbf{D}(C)$. Moreover, the functor **D** induces a 2-monad in **CAT** known as the metamodel of systems modeling since it "encodes" relevant category-theoretical constructions [28].

When formalizing generative design of power systems, suitable subcategories of $\mathbf{D}(C)$ represent heterogeneous design spaces. Indeed, requirements for a system, translated into the language of models that constitute the category C, allow checking whether a diagram represents an admissible candidate system configuration. For example, consider the simple case where requirements completely specify the target system and prescribe to build it by gluing together the two specified parts, yet with an unspecified glue. Borrowing the notation from the pushout diagram above, assume that the integral model R, the part models P and S, and the inclusion actions $p : P \rightarrow R$ and $s : S \rightarrow R$ are known and fixed. The generative design procedure consists in the search for an optimal (wrt some objective function) glue model G and actions that form a span-shaped diagram whose pushout produces the model R. The general category-theoretical solution of this "inverse pushout problem" is not (yet) known. In many practical cases, where C consists of sets with some structure (including information modeling and simulation), the gluing design space is characterized with the help of the following theorem derived from Proposition 2 of [29].

Theorem 1. Let P, S, R be sets, $p : P \rightarrow R \leftarrow S : s$ be a pair of maps, $[p, s] : P \amalg S \rightarrow R$ be the canonical map that acts as p on P and as s on S. A span-shaped diagram that has a pushout with edges p and s exists if and only if for each $r \in R$, whenever either one of the sets $p^{-1}(r)$ or $s^{-1}(r)$ is empty, there exists a unique $t \in P \amalg S$ such that $[p, s](t) = r$.

Applied to information model design, this theorem paves the way to reconstruct a minimal (in terms of a number of records) "many-to-many" relation between two tables, given a data set where their records are combined; consider for example an archive of documents that establish the equipment ownership/rent relations as mentioned above. As shown in [29], such a relation is produced with the help of the category-theoretical construction called a pullback, which is dual to a pushout. In discrete-event simulation, this construction produces a minimal (in terms of a number of events) scenario of a control system that executes a given process involving coordinated operation of two units. More advanced power system generative design problems may be solved by navigating the design space along diagram morphisms as appropriate, calculating the navigation path by means of computer algebra. There exist performant computer algebra packages for such category theory calculations [30].

Conclusion

The proposed model-based approach to design and develop a power system DT aims to make it cost-effective to compose and operate while maintaining sufficient completeness, accuracy, and usability for practical purposes. The approach has been successfully tested for validity on a small DER prosumer facility. Currently, we scale up the approach to develop a full-fledged digital platform for DT-based smart DER control [31]. The platform aims to control not only prosumers, but also industrial microgrids, smart buildings and cities, electrical vehicle charge stations, distribution grids; so, new versatile requirements for the DT arise, calling for further research.

References

[1] A. Fuller, Z. Fan, C. Day, and C. Barlow, "Digital twin: enabling technologies, challenges and open research," IEEE Access, vol. 8, pp. 108952–108971, 2020.

[2] A. Rasheed, O. San, and T. Kvamsdal, "Digital twin: values, challenges and enablers from a modeling perspective," IEEE Access, vol. 8, pp. 21980–22012, 2020.

[3] S. K. Andryushkevich, S. P. Kovalyov, and E. Nefedov, "Composition and application of power system digital twins based on ontological modeling," in Proc. 17th IEEE Intl. Conf. on Industrial Informatics INDIN'19, pp. 1536–1542, 2019.

[4] H. Sun and L. Ma, "Generative design by using exploration approaches of reinforcement learning in density-based structural topology optimization," Designs, vol. 4(2), pp. 10, 2020.

[5] H. Park et al, "Digital twin for operation of microgrid: Optimal scheduling in virtual space of digital twin," Energies, vol. 13, pp. 5504, 2020.

[6] X. He, Q. Ai, R. C. Qiu, and D. Zhang, "Preliminary exploration on digital twin for power systems: challenges, framework, and applications," arXiv, 2019. https://arxiv.org/abs/1909.06977. Accessed August 15, 2021.

[7] T. Cioara et al, "An overview of digital twins application domains in smart energy grid," arXiv, 2021. https://arxiv.org/abs/2104.07904. Accessed August 15, 2021.

[8] C. Brosinsky, D. Westermann, and R. Krebs, "Recent and prospective developments in power system control centers: Adapting the digital twin technology for application in power system control centers," in Proc. IEEE Intl. Energy Conf. ENERGYCON, pp. 1–6, 2018.

[9] S. P. Kovalev, "Systems analysis of life cycle of large-scale information-control systems," Automation and Remote Control, vol. 74(9), pp. 1510–1524, 2013.

[10] A. M. Madni, C. C. Madni, and S. D. Lucero, "Leveraging digital twin technology in model-based systems engineering," Systems, vol. 7(1), pp. 7, 2019.

[11] G. Steindl et al, "Generic digital twin architecture for industrial energy systems," Applied Sciences, vol. 10, p. 8903, 2020.

[12] Q. Qia, F. Taoa, Y. Zuoa, and D. Zhaob, "Digital twin service towards smart manufacturing," Procedia CIRP, vol. 72, pp. 237–242, 2018.

[13] M. Fowler, CQRS. 2011. https://martinfowler.com/bliki/CQRS.html. Accessed August 15, 2021.

[14] A. I. Tikhonov et al, "Development of technology for creating digital twins of power transformers based on chain and 2D magnetic field models," South-Siberian Scientific Bulletin, vol. 29, pp. 76–82, 2020. [In Russian]

[15] F. Rodríguez, A. Fleetwood, A. Galarza, and L. Fontán, "Predicting solar energy generation through artificial neural networks using weather forecasts for microgrid control," Renewable Energy, vol. 126, pp. 855–864, 2018.

[16] M. Haghgoo, I. Sychev, A. Monti, and F. H. Fitzek, "SARGON – Smart energy domain ontology," IET Smart Cities, vol. 2, pp. 191–198, 2020.

[17] S. P. Kovalyov, "Domain engineering of distributed measurement systems," Optoelectronics, Instrumentation and Data Processing, vol. 44 (2), pp. 125–130, 2008.

[18] S. P. Kovalyov, "An approach to develop a generative design technology for power systems," in Proc. VIth Intl. Workshop IWCI'2019, Advances in Intelligent Systems Research ser, vol. 169, pp. 79–82, 2019.

[19] M. A. Mabrok and M. J. Ryan, "Category theory as a formal mathematical foundation for model-based systems engineering," Applied Mathematics and Information Sciences, vol. 11(1), pp. 43–51, 2017.

[20] J. C. Baez and J. Erbele, "Categories in control," Theory and Applications of Categories, vol. 30(24), pp. 836–881, 2015.

[21] S. P. Kovalyov, "Methods of category theory in model-based systems engineering," Informatics and Applications, vol. 11(3), pp. 42–50, 2017.

[22] S. P. Kovalyov, "Methods of the category theory in digital design of heterogeneous cyber-physical systems," Informatics and Applications, vol. 15(1), pp. 23–29, 2021.

[23] D. Spivak and R. Kent, "Ologs: a categorical framework for knowledge representation," PloS one, vol. 7(1), p. e24274, 2012.

[24] K. Gürlebeck, D. Hofmann, and D. Legatiuk, "Categorical approach to modelling and to coupling of models," Mathematical Methods in the Applied Sciences, vol. 40(3), pp. 523–534, 2017.

[25] J. S. Nolan et al, "Compositional models for power systems," in Proc. Applied Category Theory 2019, EPTCS, vol. 323, pp. 149–160, 2020.

[26] V. R. Pratt, "Modeling concurrency with partial orders," International Journal of Parallel Programming, vol. 15(1), pp. 33–71, 1986.

[27] L. Garcés, F. Oquendo, and E. Nakagawa, "Software mediators as first-class entities of systems-of-systems software architectures," Journal of the Brazilian Computer Society, vol. 25, art. No. 8, 2019.

[28] S. P. Kovalyov, "Category theory as a mathematical pragmatics of model-based systems engineering," Informatics and Applications, vol. 12(1), pp. 95–104, 2018.

[29] S. P. Kovalyov, "Leveraging category theory in model based enterprise," Advances in Systems Science and Applications, vol. 20(1), pp. 50–65, 2020.

[30] J. Gross, A. Chlipala, and D. I. Spivak, "Experience implementing a performant category-theory library in Coq," in Proc. 5th Intl. Conf. Interactive Theorem Proving, Lecture Notes in Computer Science ser, vol. 8558, pp. 275–291, 2014.

[31] S. P. Kovalyov and A. A. Nebera, "A platform-based approach to implementation of future smart distributed energy control systems," in Proc. 2nd Intl. Conf. Control Systems, Mathematical Modeling, Automation and Energy Efficiency (SUMMA'2020), pp. 608–613, 2020.